Wireless Communications and Mobile Computing

Service-Oriented Management and Computing in Edge-Cloud IoT

Lead Guest Editor: Yingjie Wang
Guest Editors: Yueshen Xu and Yulong Pei

Service-Oriented Management and Computing in Edge-Cloud IoT

Wireless Communications and Mobile Computing

Service-Oriented Management and Computing in Edge-Cloud IoT

Lead Guest Editor: Yingjie Wang
Guest Editors: Yueshen Xu and Yulong Pei

Chief Editor

Associate Editors

Academic Editors

Contents

Contents

Hindawi
Wireless Communications and Mobile Computing
Volume 2023, Article ID 9832673, 1 page
https://doi.org/10.1155/2023/9832673

Retraction

Retracted: An Empirical Study on the Artificial Intelligence-Aided Quantitative Design of Art Images

Wireless Communications and Mobile Computing

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] W. Zhang and S. Tsai, "An Empirical Study on the Artificial Intelligence-Aided Quantitative Design of Art Images," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 8036323, 11 pages, 2021.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 2308615, 13 pages
https://doi.org/10.1155/2022/2308615

Research Article

Low-Delay Opportunistic Routing with Reducing Overhead in Asynchronous Duty-Cycled Wireless Sensor Networks

Fang Liu, Zheng Zhang, and Yuanan Liu

School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

Correspondence should be addressed to Fang Liu; lf@bupt.edu.cn

Received 28 February 2022; Revised 2 June 2022; Accepted 26 July 2022; Published 17 August 2022

Academic Editor: Yingjie Wang

The Internet of Things (IoT) and cloud computing are well developed and applied in different services. Among these services, some of them take latency and overhead as the metrics to evaluate the quality of services. This paper introduces how an opportunistic routing (OR) protocol reduces latency and overhead when it is applied in wireless sensor network (WSN) application among lightweight devices. OR mitigates the delay problem and balances the energy consumption and delay of the nodes in WSNs. However, OR, with the nature of broadcasting, can easily cause heavy overhead such as redundant data and control packets in the forwarding process. In the current solution, the number of candidates (potential forwarders) is not limited by the dynamics of forwarding process. Also, the overhead of establishing the candidate sets is high due to global information calculation. Therefore, a low-delay OR protocol with reducing overhead which designs the dynamic candidate area (DCA) to establish candidate sets is first proposed in this paper. The main work is as follows. (a) The duty cycle of a node is adjusted according to the distance between the node and sink node in the initialized network. (b) A method to establish dynamic candidate sets is proposed based on the adaptive duty cycle. Before and during the forwarding process, the dynamic candidate area is adjusted in time. (c) According to the feature of the candidate area, the corresponding routing metric is proposed to complete the cooperative communication among candidates. Through further theoretical analysis and simulations, the results indicate that this protocol achieves better performance in terms of transmission delay, data and control overhead, and network lifetime compared to the state-of-the-art solutions.

1. Introduction

In smart city, smart industry, smart healthcare, environmental protection, and military fields, the Internet of Things (IoT) and cloud computing have been widely used. Wireless sensor networks (WSNs) use easy-to-deploy and low-cost sensor nodes to meet communication requirements and collect and upload data. It is one of the most powerful ways for data acquisition in the era of cloud computing. In WSNs, due to the limited battery, storage, and computing capacity of sensor nodes, the focus of related research is often how to extend the lifetime of network. From the perspective of energy harvesting, some studies have proposed the methods to collect energy from the environment where the network is located. Additionally, there are also some ways to supplement energy for nodes through wireless charging. However, in some scenarios the requirements for replenishing energy cannot be met, because the environment in which the sensor network is located can be diversified. Duty cycling [1] can greatly reduce the energy consumption of sensor node idle listening, which has become a primary research direction.

Under the duty cycling mechanism, sensor nodes will periodically perform active/sleep scheduling and nodes will turn off their radio for sleep mode when there is no data transmission. Obviously, while the duty cycling reduces energy consumption, it also introduces a waiting delay for the receiving nodes to wake up during the data forwarding process. In order to reduce the sender's waiting time, the literatures [2–5] consider applying the synchronous duty-cycled MAC protocols at the MAC layer. This type of protocols, with the communication among nodes and base station system scheduling, controls nodes to wake up synchronously or according to schedule during the forwarding process. However, this kind of node active/sleep scheduling costs a

huge amount of control overhead due to time synchronization and reduces the network scalability. In asynchronous duty-cycled MAC (ADC), each sensor node has an independent active/sleep schedule. During the operation of the network, each node has a same or different duty cycle and wakes up at different time. When deterministic routing is used, it will be impossible to avoid the waiting time delay caused by sleeping nodes.

Opportunistic routing (OR) was first used in wireless mesh network [6]. Unlike traditional single routing, there is no fixed forwarder set for each node in opportunistic routing. Each node has a set of candidate nodes as potential forwarders. When a node is forwarding data packets, it can select the node with better real-time communication quality as its forwarding node. This feature in WSNs can not only reduce the delay caused by waiting for a single forwarder to wake up but also select the nodes with higher battery power or better link quality in communication. Therefore, it is promising to combine opportunistic routing and duty cycling in WSNs. Opportunistic routing protocols have already had specific research practices in duty-cycled WSNs [7–9]. Although opportunistic routing provides a set of candidates to solve the delay problem of waiting for a single forwarder to wake up, it also brings many new challenges. First of all, the size of candidate set is an important factor, affecting network delay and energy consumption performance. If there are more nodes in the candidate set, there will be more potential forwarders that can be selected by the sender. However, as the candidate set expands, the number of nodes that wake up at the same time will also increase, thereby increasing the data and control packet overhead. The generation of redundant data packets will greatly increase the energy consumption of network and affect the transmission quality of network. Conversely, the smaller the number of nodes in the candidate set is, the smaller the expected number of nodes in the active mode during forwarding process will be. Therefore, the waiting delay will increase and the performance of opportunistic routing will degrade to a deterministic routing. Actually, sender's waiting delay and overhead of data and control packets are the most important issues in OR. Therefore, a reasonable size of candidate set has become a key in the success of opportunistic routing. Secondly, starting from ORW [10], the establishment of candidate sets in various opportunistic routing mainly includes global calculation and local calculation. Many studies have deduced the maximum number of candidate sets in different scenarios, but they depend on specific network conditions. The computational overhead of the network is high and the network scalability is weak. Therefore, it is also an important challenge to design an opportunistic routing protocol that reduces computing power and control overhead for future application in WSNs. This paper proposes a low-delay opportunistic routing protocol based on dynamic candidate area design, which is aimed at dynamically partitioning candidate area to establish candidate sets and update them in real time for reducing overhead. In this protocol, we take advantage of the energy distribution of sensor nodes in WSNs to reduce the transmission delay of the network through the adaptive duty cycle, without affecting the lifetime of the network. Also, adaptive duty cycle establishes the foundation of dynamic candidate area design.

2. Related Work

Many papers focus on effectively combing the OR and duty cycling to achieve better network performance [10–19]. In these papers, the researchers systematically proposed the OR protocol design based on the specific asynchronous duty-cycled MAC protocol. According to the actual routing process, OR protocol mainly includes two steps: designing a candidate set of forwarding nodes and selecting a unique forwarder. Among them, how to establish the candidate set for each node is the key point that every protocol designer needs to consider. The basic idea of establishing candidate sets is to use the global information of the network and the local information of each node to select an appropriate subset from the set of neighbor nodes. It should be noted here that candidate and forwarder sets are the same concept in most papers [7–12].

In the previous research, most protocols mainly design candidate sets based on global information. In ExOR [7], expected transmission distance is used to measure the distance between each node and the sink, which in turn serves as the metric rank of neighbor nodes. In 2012, ORW [10] first proposes OR combined with asynchronous duty-cycled on the basis of ExOR. ORW combines duty cycling on this basis and proposes the Expected Duty Cycle (EDC) as route metric. By calculating the EDC of each neighbor node, ORW adds the nodes that meet the requirements of the candidate set and selects the candidate node that wakes up firstly in the active mode to forward data packets. ORW firstly examines the improvement of network performance which is brought by the combination of OR and duty-cycled WSNs. However, calculating the EDC of each node requires a global, recursive calculation, and it does not consider the energy balance of the network which leads to calculation overhead and a decrease in the lifetime of the network. Similarly, on the basis of ORW, Meng successively proposes ORD [20] and ORR [21] protocols. ORD, adopting data aggregation, allows nodes to wait for a period of time before forwarding so as to receive data packets from multiple nodes, which achieves a trade-off between network energy consumption and delay. ORR, based on local information introduced in ORW, is combined with the remaining energy of the node to propose Forward Score (FS). In 2018, on the basis of ORR, Khan proposes MORR by taking the number of neighbor nodes and channel interference to select candidates on the basis of ORR [15]. In 2021, Weiqi proposed a piggybacking-based opportunistic routing protocol (PORA) [22] to improve the performance of the network. In PORA, R value distribution and H value distribution are proposed to evaluate the weight of each edge in the network. Base on the weight calculation, the Dijkstra shortest path algorithm is used to find a set of prioritized forwarders from a source node to sink node. And one of the forwarders is selected by forwarder coordination. With piggybacking mechanism, PORA achieves better performance in terms of energy consumption and packet delivery ratio. However, calculation

for global information to establish weighted graph limits the scalability of the network and increases the overheads caused by global information communication.

In the process of establishing candidate sets, some need to use global information through complex calculations. Some do not explicitly limit the size of the candidate set, which increases the possibility of packet redundancy overhead. And they all ignore the impact of duty cycling. With continuous evolution of the OR protocol in ADC-WSNs, it is gradually mature to use both global and local information to design candidate areas, in order to determine candidate sets. In 2017, Chen et al. proposes a lightweight OR protocol LWOF [8]. LWOF uses the conclusion in [23] to establish the candidate set by dividing a 60-degree sector-shaped candidate area, and it optimizes the length of the preamble in the MAC protocol according to the duty length. Although the LWOF protocol is only suitable for WSNs with high network density, how it designs areas is another effective solution to establish candidate sets without global computation. Therefore, the idea of candidate area is adopted in recent opportunistic routing protocols. For example, in 2019, Hawbani et al. propose the LORA protocol based on the Candidate Zone (CZ) [12]. LORA first calculates the width of the CZ according to the network density, and it uses the global information of the network to establish candidate sets. Then, it calculates the distribution of multiple dimensions as a metric to measure the forwarding priority of the candidates. Through the combination of global information and local information, LORA limits the size of the candidate set and achieves good performance in terms of transmission delay and energy consumption. But its multidimensional computation limits the scalability of the network and requires higher computational and control overhead, and there is the possibility of no subregions. In addition, Xiang et al. propose the ADCCOR [24] protocol in 2019. The ADCCOR combines the relationship between link quality and transmission radius to establish a candidate area that reaches a certain forwarding success probability. Because it adjusts the duty length of different nodes, ADCCOR requires additional communication overhead. In 2020, LEOR [3] proposed by Omid Abedi1 et al. constrains the range of the candidate area according to the distance progress [25] and dynamically adjusts the range during the forwarding process according to the number of the wake-up candidates. Receiver node's duty cycle became longer because of adaptivity, so the nodes far from sender and near to receiver should be selected, which leads to high power transmission. How it restricts the scope of the forwarding area will lead to unbalanced network energy consumption in the case of constant data at a fixed node.

In addition to local geographic information and link quality information, Qaisar introduced a trust-based load-balanced OR (TORP) [26] for security information. The candidates in TORP are prioritized on the basis of a trusted OR metric based on the average of two probability distributions: the direct trust distribution and the recommended trust distribution. With local security information, TORP increase the throughput and packet delivery ratio. In 2021, Shen proposed LDC-COR [27] considering link correlation. LDC-COR firstly assigns nodes with low correlation to a common group and schedules the nodes within this group to wake up simultaneously for forwarding packets in a common cycle. Then, LDC-COR takes account of both link correlation and link quality to improve the expected transmission count (ETX). Also, LDC-COR only requires the information of one-hop neighboring nodes which introduces minimal communication overhead. As a result, LDC-COR reduces the energy consumption with a slight increase of end-to-end delay. Besides, it makes the networks hard to respond dynamically if we consider the traditional approaches. Machine learning (ML) can be applied to solve the routing issues in WSNs [28]. For example, Donta et al. propose a delay-aware data fusion (DADF) [29] approach to achieve the trade-off between the delay and energy while this approach performs the data fusion in 2022. They conduct the simulation tests in various scenarios to evaluate the performance of their work.

In general, the existing schemes do not comprehensively consider the dynamics of the forwarding process and the duty length of nodes to limit the size of the candidate set. In addition, global metrics are centrally computed in the sink node [12] and the overhead of establishing candidate sets based on global information is not considered in most literatures. Therefore, this paper proposes a low-delay OR protocol with reducing overhead. The protocol efficiently establishes candidate sets by dynamically partitioning the candidate area and designing the corresponding routing metric according to the characteristics of the division of the candidate area.

3. System Model and Problem Formulation

The network is deployed in a circular area with a radius of R. We assume that the sink node is located at the center of the circle, and the rest of the sensor nodes are randomly and statically distributed in this area. They make up the entire network set N, $N = \{n_0, n_1, n_2 \cdots\}$. The communication radius of sensor nodes (including sink node) is r, and all sensor nodes are homogeneous. Except for the sink node, its energy is not limited, and the battery capacity of the remaining nodes is a fixed value EJ. A node $n_i \in N$ can obtain its own location information (x_i, y_i) through GPS or other methods, and the Euclidean distance between node n_i and node n_j is $d_{i,j}$. All nodes whose distance to node n_i is less than the communication radius r constitute the neighbor node set N_i, $N_i = \{n_j | n_j \in N \& d_{i,j} \le r\}$. Table 1 lists the notations used in this article.

In the proposed protocol, data forwarding between nodes is based on the BoX-MAC protocol. As an asynchronous duty-cycled MAC protocol, the active/sleep scheduling of each node is independent. We use duty length λ to represent the number of slots contained in each node's cycle. Assuming that the duty length is λ, then the duty cycle of each node is $1/\lambda$, which means that every λ time slots, the node wakes up randomly in one of the time slots. When a node is forwarding packets, it will continue to send data packets until a neighbor node in the candidate set wakes

Table 1: Notation.

Symbol	Description
N	The network $N = \{n_0, n_1, n_2 \cdots\}$; $n_i \in N$ is sensor node
N_i	The neighbor node set of n_i
$N_{c,i}$	The candidate set of n_i
$d_{i,j}$	The Euclidean distance between n_i and n_j
λ	The duty length λ
λ_0	The initial duty length broadcasted by sink
R	The radius of circular area
r	The communication radius of sensor nodes
S_i	The candidate area of n_i
T_i	The cycle time of n_i
d	The network density
θ_0	The initial angle of candidate area which is broadcasted
θ_i	The angle of candidate area for n_i
δ	The packet transmission delay
$l_{\text{ring},i}$	The ring where n_i is located

up or the forwarding times out. The node without data to forward will wake up periodically according to its own duty cycle and check whether there is data packet transmission in the channel when waking up. During the active mode, if no data packet transmission is detected, the node will enter the sleep mode. If a data forwarding request is detected, the node will send an ACK. The sender analyzes the received ACKs to determine whether there is a neighbor node belonging to its candidate set to wake up.

The proposed protocol is mainly to lower the end-to-end delay and overhead from redundant data packets. In duty-cycled WSNs, delay mainly includes two parts: transmission delay δ_{tran} and sender waiting delay δ_{wait}. According to previous literature, the influence of the distance between the forwarding nodes of the same hop node on the transmission delay is often ignored, so the transmission delay mainly depends on the number of multiple hop transmissions. Different from single routing protocol, opportunistic routing protocol also introduces sender waiting delay in the forwarding process. Sender waiting delay refers to the time sender that waits for a node in the candidate set to wake up. The end-to-end multihop delay $\delta_{i,s}$ from the source node n_i to the sink node equals to the accumulation of single-hop delay.

$$\delta_{i,s} = \sum_{m=1}^{k} (\delta_{\text{tran},m} + \delta_{\text{wait},m}). \tag{1}$$

In the forwarding process of opportunistic routing, when multiple candidates are awake at the same time, they will all receive data packets from the sending node. Although only one forwarding node will be selected from the wake-up nodes after the *candidate coordination*, the overhead caused by redundancy of the data packets will inevitably occur in this process, because the remaining nodes will discard the data packets they receive. Based on the above analysis, the expected number of redundant data packets $M_{\text{RP,single}}$ during a single-hop forwarding process is calculated by

$$M_{\text{RP,single}} = \sum_{0}^{k} (k-1) P_{\Delta t}(i=k)\, k \in [0, N_c]. \tag{2}$$

Among them, $P_{\Delta t}(i=k)$ is the probability that k nodes wake up at the same time. And $k-1$ means that the data packets received by the remaining $k-1$ nodes are all redundant data packets when k nodes wake up at the same time. Then, $M_{\text{RP,sum}}$, stands for the sum of the expected redundant data packets generated by the final single data packet sent from the source node to the sink node.

$$M_{\text{RP,single}} = \sum_{0}^{k} (k-1) P_{\Delta t}(i=k) k \in [0, N_c]. \tag{3}$$

4. Low-Delay Opportunistic Routing with Reducing Overhead (LDORRO) Protocol

Before we introduce each part of the protocol, we take a source node to send a data packet as an illustrative example to show the process of the entire protocol.

A source node n_s calculates its own duty length according to the duty length broadcast by the sink node and its own geographic location information at the adaptive duty cycle stage. According to the calculated duty length, the node can further calculate the angle value of its own candidate area, which corresponds to the angle of the fan-shaped area shown in Figure 1. That is the angle value of the initial candidate area. Then, the node will enter periodic active/sleep scheduling until it collects data packets from the environment. The node holding the data packets will continue to send the preamble until it receives ACKs from the nodes in the current candidate area. Within a certain time, if the corresponding ACK is not received, n_s will increase the angle of the candidate area and continue to send the preamble. If it receives ACKs from nodes in the candidate set, it will select a single forwarder from multiple awakened candidates according to the predesigned metric as the final forwarding one. The above process will be repeated until the data packet is forwarded to the sink node.

4.1. Network Initialization. In the proposed opportunistic routing protocol, first, we need to adjust the duty length of each node according to the distance between the node and the sink node. The farther the node is from the sink node, the longer the active time of the node in each cycle.

Since we take the sink node as the center of the network, we divide the entire network into a number of rings with r as the diameter of ring. As stated in [24], it may not be possible to divide the network evenly into multiple rings. We call the ring closest to the sink node as ring 0, like [24] do, from inside to outside, in order of ring $0, 1, 2, \cdots, n-1$. After

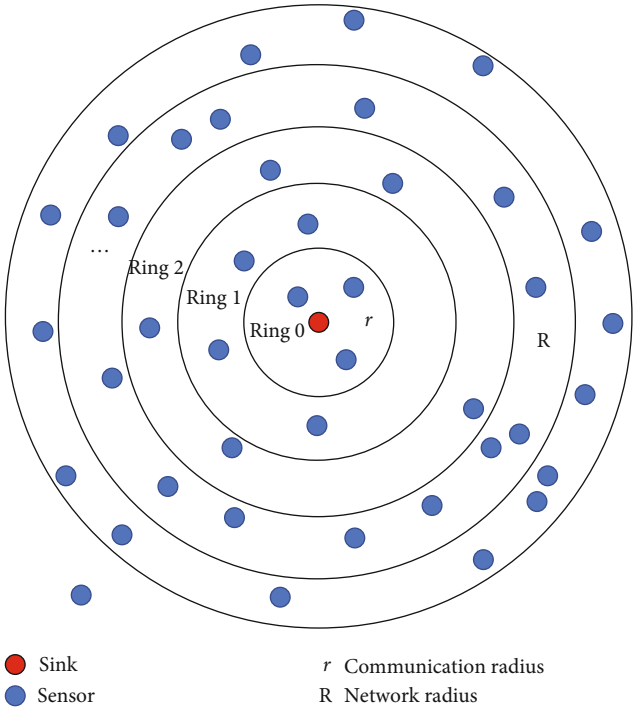


FIGURE 1: The wireless sensor network is divided into different rings according to the communication radius r. The sink node is located at center of the circular area with radius R.

dividing the networks, we can calculate the ring in which each node is located and update the duty length of each node. What needs to be pointed out is we only update the duty length of different nodes during the network initialization, which is different from the traditional adaptive duty cycle approaches [30]. It reduces the additional overhead caused by multiple updates and we can make up for the shortcomings of this simplified operation through proposed dynamic candidate area design.

After the network is deployed, node i determines the ring $l_{\text{ring},i}$, where it is located according to the distance $d_{i,s}$ between it and the sink node by (1). For nodes located at ring 0, that is, the nodes located within the distance between themselves and the sink node which is less than the communication radius r, to be specific, the following equation is for reference.

$$l_{\text{ring},i} = \left\lfloor \frac{d_{i,s}}{r} \right\rfloor. \tag{4}$$

Then, the sink node first broadcasts a duty length λ_0 as the initial value of each node. After receiving the broadcast duty length, n_i updates its duty length λ_i according to

$$\lambda_i = \begin{cases} \lambda_0 \, d_{i,s} \le r, \\ \lambda_0 \left(1 - \dfrac{E_{r,0} + E_{t,0} - E_{t,i} - E_{r,i}}{E_{lpl,i}}\right) d_{i,s} > r. \end{cases} \tag{5}$$

In equation (2), the nodes in ring 0, their duty length is consistent with the received value λ_0. The node located in the ith ring calculates the difference between ring i and ring 0 in terms of energy consumption, and then, it adjusts its own duty length according to the difference. To simply evaluate the difference, we calculate the total transmit energy consumption $E_{t,0}$ and the received energy consumption $E_{r,0}$ for nodes on ring 0. Then, we can calculate the proportion of low power consumption that the excess energy consumption is sufficient to support. Finally, we get the decreased duty length without affecting the lifetime of the network since we take advantage of the energy distribution. After all nodes have calculated their own duty length, the node duty length initialization is completed.

4.2. Candidate Selection. The design of the dynamic candidate area is mainly embodied at two stages. And the basic idea of this dynamic design is simplified as follows. (1) Before sender forwarding, the size of the candidate area is designed according to the duty length of the node. (2) When forwarding, the size of the forwarding area is dynamically adjusted according to the waiting time of the node.

After the node initialization is completed, the duty length of nodes on different rings is different. Thanks to this operation, we make full use of the characteristics of energy distribution in the WSNs to reduce the transmission delay without changing the whole lifetime of the network. To further reduce the transmission delay near the sink node, we adjusted the initial candidate area of the nodes on different rings according to the duty length of the node.

As discussed in [1], in our protocol, we adopt a fan-shaped candidate area design for each node based on the research in [8]. Before sender forwarding, the fan-shaped candidate area is calculated, for each node, according to the number of rings where it is located. The design of the candidate area is mainly derived from reducing the sender waiting time. And that is to ensure that the time of waiting for the first node to wake up when the nodes on different rings are forwarded is consistent. Consequently, the angle θ_i of each node's candidate area is calculated by equation (6) before it starts to forward packets.

$$\theta_i = \frac{\theta_0}{\lambda_0} \lambda_i, \tag{6}$$

where θ_0 is the initial angle; here, we choose the angle of the candidate area of the nodes on the outermost ring as the initial angle. From (3), it can be found that θ_i of the fan-shaped candidate area is proportional to the duty length λ_i of the node. Since the nodes in the outermost ring have the biggest duty length, the angle of the candidate area is the smallest, which means the candidate area is the smallest. This design not only considers the duty length of the outermost node but also constrains the range of the outermost node forwarding node selection. Just as the purpose of calculating the direction distribution and perpendicular distribution of each node in [12], we give a higher priority to the nodes that are closer to the sink and central line. And the nodes in the ring that is closest to the sink node have the largest forwarding angle, which is also conducive to reducing

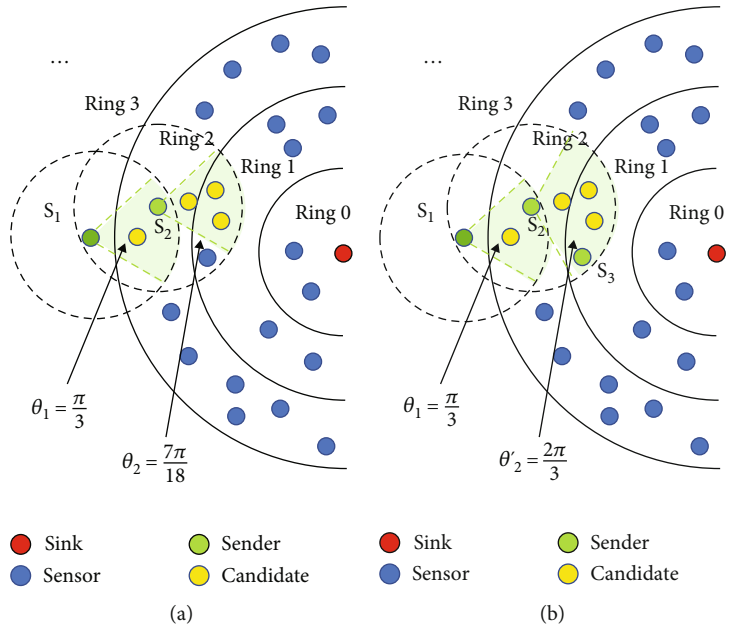


FIGURE 2: (a) Sensor nodes in different rings get different candidate areas which are determined by the angle calculated according to adaptive duty cycle. Since the duty cycle of s_2 is smaller than s_1, the calculated angle θ_1 is smaller than θ_2. (b) When s_2 is forwarding packets, if all the candidates of s_2 showed in (a) are not awake, it will change θ_2 into $\theta_2{}'$ to get more candidates until θ_2 equals to π or a candidate wakes up.

the delay problem caused by the small duty length of the nodes in the inner rings.

When a node needs to forward data, the node will select a fan-shaped area corresponding to the angle as the candidate area according to its own initial angle, as shown in Figure 2. Specifically, when a node is forwarding, it will first continue to send the preamble. If its neighboring node receives the preamble signal when it wakes up, it will calculate whether it is located in the forwarding area according to the forwarding angle value carried in the preamble. This calculation process relies on the geographic location information and the forwarding angle value of the sink node, the sending node, and the neighbor node that received the preamble. The geographic location of the sink node can be obtained through the initial network broadcast, and the geographic location information of the sending node and the current forwarding angle range can be obtained from the preamble.

When a neighboring node judges that it meets the receiving conditions, the node will send an ACK to inform the sending node. At the same time, it will remain awake until the sending node successfully forwards or the waiting time expires.

At this time, according to the duration of sending the preamble by the sending node and the number of ACK received during this period, it is further judged whether it is necessary to dynamically adjust the size of the forwarding area. If there is a wake-up node in its candidate area, the sending node can select one of the nodes as the final forwarding node to complete the forwarding task.

4.3. Candidate Coordination. Candidate coordination is to select the final and only forwarder from multiple candidates. Reasonable candidate coordination should consider the energy consumption difference and transmission delay among nodes to ensure that only one node is responsible for the forwarding task. Since we have restricted the selection of candidate nodes from the perspective of forwarding when designing the candidate area, we consider more energy consumption and forwarding distance factors. According to (7), we calculate the metric ϕ of each candidate j and select the node with the largest metric among the wake-up nodes as the forwarding node.

$$\phi = \alpha \frac{E_i}{\bar{E_w}} + \beta \frac{\bar{d_{i,w}}}{d_{i,j}}. \tag{7}$$

E_i is the residual energy of node i, and $\bar{E_w}$ is the average energy level of wake nodes in candidate area. The distance between sender i and candidate j is $d_{i,j}$, while $\bar{d_{i,w}}$ describes the average forwarding progress of wake nodes in the candidate area.

To summarize, the above process can be illustrated by Algorithm 1. From the pseudocode of LDORRO, we can see that the time complexity and space complexity of the algorithm are both O(N). LDORRO's time complexity depends on the number of candidates and the routing metric calculation for each candidate. Because each candidate can locally calculate the routing metric, the time complexity is only O(N). In LWOF, it selects candidates with fixed candidate area, and the time complexity is O(1) [8]. It does not consider the residual energy and duty length of nodes. Compared with LEOR [3], LEOR also constructs the dynamic candidate sets with O(N) time complexity. In terms of space complexity, each node only needs to store the residual energy value, position, and the number of its neighbors' packets. The space complexity is also O(N). It is a lightweight opportunistic routing

When network N is deployed **do**
1: Sink node broadcasts θ_0, λ_0
2: **For** each node such as $n_i \in N$**do**
3: Get $d_{i,s}$ and calculate θ_i, λ_i by (5) and (6)
4: Establish candidate set $N_{c,i}$ by θ_i
5: **End for**
6: **End**
7: **For** each node such as $n_i \in N$ **do**
8: **If** n_i gets packet in buffer (generated or received) **do**
9: **If** n_i is a neighbor node of the sink node **do**
10: Send data packets to sink node
11: **Else do**
12: Send preamble to data packet to $N_{c,i}$
13: **If** n_i received ACKs from $N_{c,i}$ **do**
14: Calculate the metrics of nodes which send ACKs
15: Select one node such as n_j as forwarder
16: **Else do**
17: n_i recalculates θ_i and rebuilds $N_{c,i}$
18: **End if**
19: **End for**

Algorithm 1: LDORRO scheme.

algorithm. Therefore, it can solve the issue which is raised by the limited computing and memory resources of sensor nodes.

4.4. Correctness. In this section, it proves that the protocol we proposed can ensure the stability of the transmission delay and rationally reveal our dynamic candidate area design ideas. Also, we analyze the performance of LDORRO in terms of delay and overhead.

Theorem 1. *Given network density d, the candidate area S_i, and transmission delay Δt, if the active/sleep scheduling between different nodes meets the exponential distribution, the probability that the time waiting for the first node in the candidate area to wake up is less than or equal to Δt can be formulated as follows.*

$$P_{\Delta t}(i \geq 1) = 1 - e^{-(d \times S_i / T_i)\Delta t}. \tag{8}$$

Proof. When the wake-up time difference between nodes meets exponential distribution, we assume that the wake-up time difference between any two nodes is an exponentially distributed random variable, and its average value is the T/N_i, where T is the duty-cycled period and N_C is the number of nodes in the candidate set of sender n_i. Then, we can view the wake-up sequence of a group of nodes n_i as the Poisson process. For example, the k candidates are $n_0, n_1, n_2 \cdots n_{k-1}$, and the corresponding wake-up process $W\{w_0, w_1, w_2, \cdots w_{k-1}\}$ is a Poisson process.

Since the network density is d, the number of candidates N_C in the candidate area can be calculated according to (9). Therefore, according to the exponential distribution and the nature of the Poisson process, within the time Δt, the probability of at least 1 of the N_c candidates being awakened is shown in (10). From (9) and (10), we can prove Theorem 1 obviously.

$$N_c = d \times S_i, \tag{9}$$

$$P_{\Delta t}(i \geq 1) = 1 - P_{\Delta t}(i = 0) = 1 - e^{-(N_i/T_i)\Delta t}. \tag{10}$$

□

Theorem 2. *Given the network density d, the transmission delay Δt, and the probability $P_{\Delta t}$ that the time waiting for the first node in the forwarding area to wake up is less than or equal to Δt, the duty length λ of the candidate node is proportional to the fan-shaped candidate area angle θ.*

Proof. According to Theorem 1, when we fix some variables in (10), we can deduce the relationship of several other variables. Here, we derive the relationship between the size of the candidate set and the duty length of a given delay and network density, while maintaining the same probability $P_{\Delta t}$. The reason why the above variables are fixed is consistent with the commonly used network scenarios in WSNs. For example, the application of WSNs often needs to meet the given delay requirements. And the nodes in WSNs are static and randomly distributed as assumed in this article. □

For nodes n_i and n_j located in different rings, (11) and (12) describe the probability $P_{\Delta t}(i > 1)$ and $P_{\Delta t}(j > 1)$ that at least one node wakes up within the same given time Δt when they start opportunistic forwarding.

$$P_{\Delta t}(i \geq 1) = 1 - P_{\Delta t}(i = 0) = 1 - e^{-(N_i/T_i)\Delta t}, \tag{11}$$

$$P_{\Delta t}(j \geq 1) = 1 - P_{\Delta t}(j = 0) = 1 - e^{-\left(N_j/T_j\right)\Delta t}. \tag{12}$$

In our network, in order to keep the same in sender waiting time of nodes in different ring, we set $P_{\Delta t}(i > 1) = P_{\Delta t}(j > 1)$ to obtain new equation (13). In (13), we will

further observe the relationship between the candidate area angle and the duty length that satisfies equation (11).

$$e^{-(N_i/T_i)\Delta t} = e^{-\left(N_j/T_j\right)\Delta t}. \tag{13}$$

For the sector-shaped candidate area, the relationship between S_i and the candidate area angle θ_i is described by (14). And according to node's duty length and time slot, the period T_i can be calculated by (15).

$$S_i = \frac{\theta_i \times r^2}{2}, \tag{14}$$

$$T_i = \lambda_i \times \Delta t_{\text{slot}}. \tag{15}$$

Bringing (9), (14), and (15) into (13), we get the relationship between the angle of the fan-shaped candidate area and the duty length, that is, it satisfies the proportional relationship in equation (17).

$$e^{-\left(\theta_i r^2 d/2\lambda_i \Delta t_{\text{slot}}\right)\Delta t} = e^{-\left(\theta_j r^2 d/2\lambda_j \Delta t_{\text{slot}}\right)\Delta t}, \tag{16}$$

$$\frac{\theta_i}{\lambda_i} = \frac{\theta_j}{\lambda_j}. \tag{17}$$

5. Results and Discussion

In the simulation, nodes are randomly distributed in a circular area with the radius of 100 m, and the sink node is located in the center of the circular area. Here, we implement the BoX-MAC protocol at the simulation level: each node randomly selects a slot to wake up within the duty length, and the size of each slot is 10 ms. The initial battery power of each node is 50 mJ, and its transmit power, low-power listening, and sleep mode power are shown in Table 2. A data packet is generated from a random node every 0.1 s in the network, and the size of the data packet is 36 bytes. Table 2 shows the main relevant parameters used in the simulation.

In order to analyze the performance of the routing protocol, we mainly simulate the following metrics.

Average delay: after simulating a specified number of times, we calculate the average transmission time required for a single randomly generated data packet to be sent from the source node to the sink node.

Average number of redundant data packets: these metrics reflect the overhead of redundant data packets. During the transmission of a single data packet, all awoken candidates will hear and receive it due to the nature of broadcast. The average number of redundant data packets increases per hop. Similarly, we conduct multiple simulations for the generation of a single random data packet.

Energy consumption: the overall energy consumed by the network when a specified number of data packets are randomly generated and forwarded to the sink node during a simulation period.

According to the application scenarios of our protocol and the idea of dynamically adjusting the forwarding area, we mainly compared with the two existing protocols: LWOF and LEOR.

Table 2: Settings of simulation parameters.

Simulation parameter	Setting value
Simulation area	A circular area with a 100-meter radius
Number of nodes	100~300
Transmission range	20~40 m
Initial energy	50 mJ
Transmission power	38.4 mW
Reception power	38.4 mW
Packet size	36 bytes
Data rate	512 kbps
Time slot	10 ms

LWOF: LWOF is a lightweight opportunistic routing protocol. It uses the design forwarding area to establish a candidate set and then selects the first awakened node in the set as the final only forwarding node. The defect of LWOF is that the size of the forwarding area is fixed, and the forwarding area of each node is a fan-shaped area with an angle of 60 degrees. It can only be used in extremely dense WSNs. In addition, LWOF does not use metric to prioritize candidates, resulting in an unbalanced energy consumption of the network.

LEOR: LEOR establishes a set of candidates in the design forwarding area and also designs the selection of candidate nodes for metric optimization. LEOR's forwarding area design is mainly to limit the size of the forwarding set and ensure a certain forwarding process. The initial forwarding area of each node is an annular area with an inner radius of $R/2$ and an outer radius of R. And LEOR also considers the dynamics of the forwarding area. When no candidates wake up within the waiting time, LEOR will adjust the size of the forwarding area. The main drawback is that the design of the forwarding area does not incorporate the duty length of the node, and the initial forwarding area size of different nodes is the same, which leads to a decrease in network performance.

The simulation results show that, compared with LEOR and LWOF, our proposed routing protocol has achieved better results in reducing delay and reducing overhead. Under different network densities, our proposed protocol reduces the delay of 10 percent on average and reduces the number of redundant data packets of 15 percent. In terms of energy consumption, we have achieved a better energy balance above the similar energy consumption level, as well as the other performance advantages mentioned above. In the simulation, we studied the influence of different node numbers and different node communication distances on each simulation metric.

5.1. Average Delay. In order to evaluate the delay under different opportunistic routing protocols in different scenarios, we separately studied the influence of the number of nodes in the network and the communication distance of nodes on the delay in Figures 3 and 4. Here, we mainly compare the average delay of sending a single data packet from the source node to the sink node multiple times in different

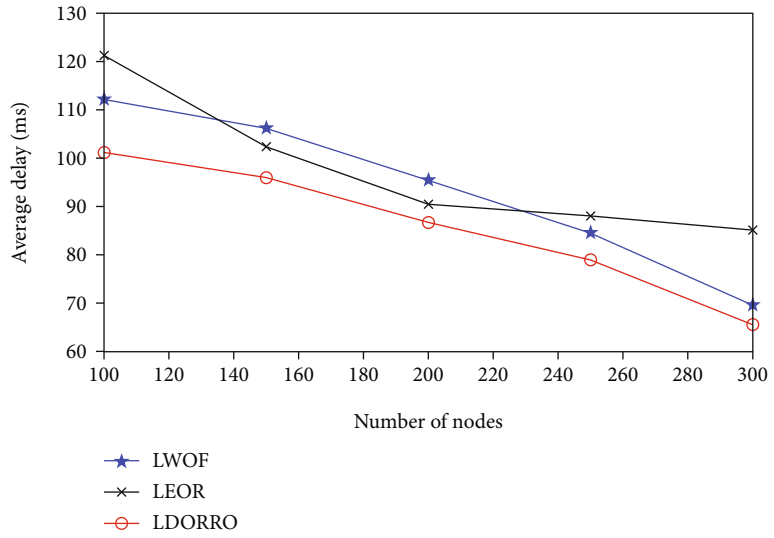


Figure 3: Average delay of different OR schemes for networks with different number of nodes.

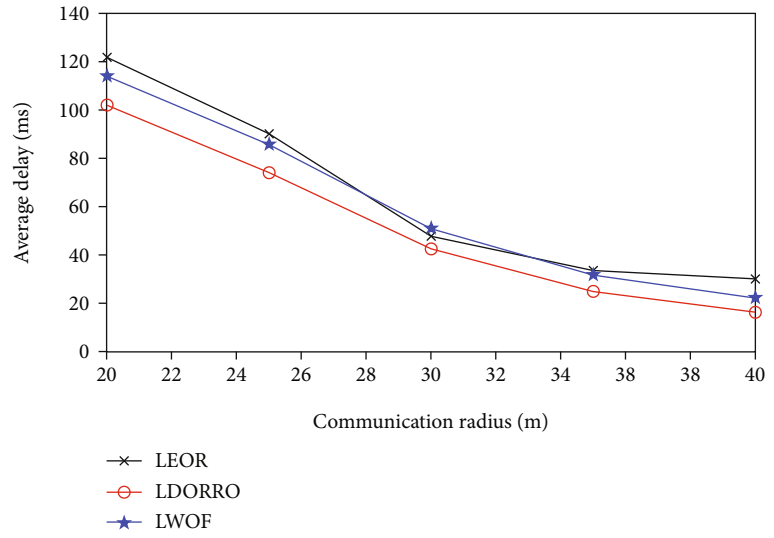


Figure 4: Average delay of different OR schemes for networks with different communication radius.

scenarios. First, when the communication distance of the node remains unchanged (20 m), the relationship between the delay and the number of nodes in the network is shown in Figure 3. It can be seen in the figure that as the number of nodes continues to increase, the time delay shows a downward trend. This is because although different routing protocols restrict the size of the candidate set, the size of the candidate set will still increase in varying degrees as the network density increases. The increase of the candidate set increases the probability of awaken nodes in the same time, thereby reducing the waiting delay in the opportunistic routing process. When the number of nodes continues to increase, the performance of LWOF gradually approaches LDORRO. This is because when the network density is large, the probability that LDORRO needs to dynamically adjust the forwarding area during the forwarding process is reduced, and the forwarding area sizes of LWOF and LDORRO will be more consistent. A closer effect has been achieved in terms of time delay. Compared with LEOR and LDORRO, when the number of nodes is small, the effect of the forwarding area established by LEOR is significantly worse, and the waiting time delay is longer. This is because when the network density is low, LEOR screens forwarding nodes according to the threshold radius, which causes its forwarding set to drop extremely, and the threshold radius needs to be changed many times, resulting in too long waiting time for the sender. In Figure 4, when we keep the number of nodes (200) constant, as the communication radius of nodes increases, the end-to-end delay also shows a decreasing trend due to the decrease of hops from source node to sink node. As the increase of the communication radius, the candidate set keeps increasing, so the gap between the

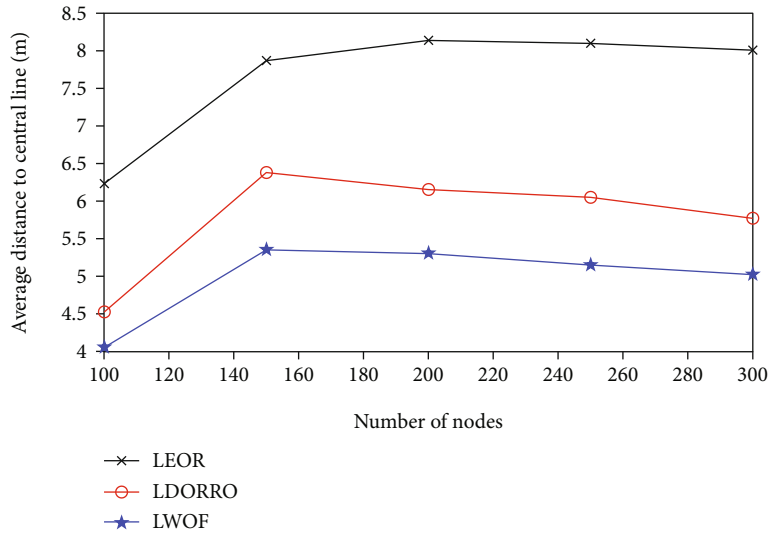


Figure 5: Average distance to central line for candidates of different OR schemes with different number of nodes.

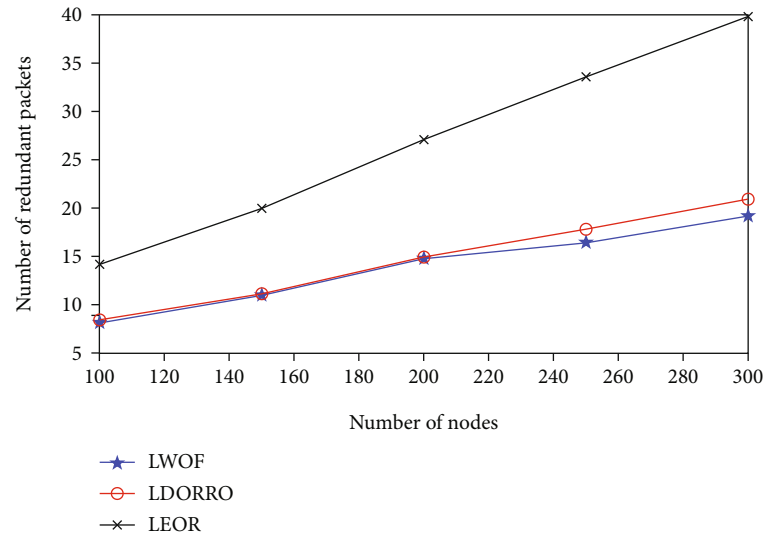


Figure 6: Number of redundant packets of different OR schemes for networks with different number of nodes.

three protocols also decreases. However, LDORRO still achieves the best time delay effect by adjusting the duty cycle and designing the dynamic candidate area.

At the same time, in order to prove the superiority of LDORRO candidate node selection, we also calculated the distance from the candidate nodes in different candidate sets to the central line (the connection between the sink node and the sender node) in the simulation. In Figure 5, it can be seen that since LWOF establishes the smallest and densest candidate regions, its value is the smallest, while LDORRO also achieves a smaller average value by virtue of the design of dynamic candidate regions. Compared with LWOF, LDORRO can dynamically adjust the size of the candidate region when the number of nodes is small so that LDORRO can obtain better end-to-end latency performance when the number of nodes is small.

5.2. The Number of Redundant Packets. We compare the number of redundant data packets generated by the three routing protocols in the forwarding process under different network densities. In Figure 6, as the number of nodes in the network increases, the number of redundant data packets will inevitably increase. However, thanks to the design of the dynamic candidate area, LDORRO has achieved a better effect of limiting the size of the candidate set and reducing the number of redundant data packets. When the number of nodes increases, the probability that the sender will wake up the node during the forwarding task also increases, so the probability of dynamically increasing the size of the candidate area decreases. Therefore, when the number of nodes continues to increase and the network density continues to rise, LDORRO can effectively slow down the surge of redundant data packets through this

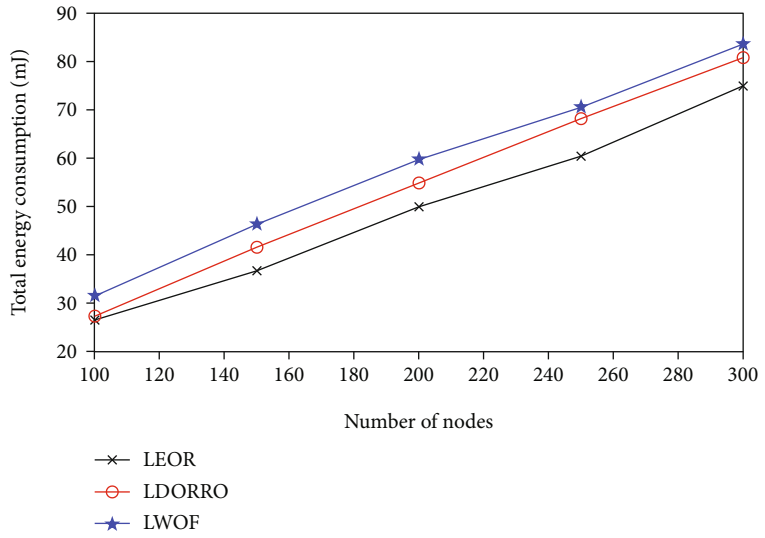


Figure 7: Total energy consumption of different OR schemes for networks with different number of nodes.

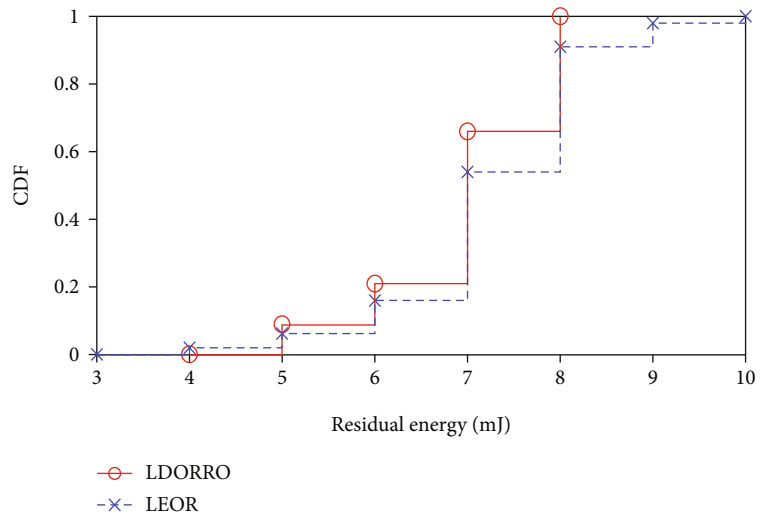


Figure 8: The CDF function of the residual energy of different OR schemes after generating 1000 data packets.

lightweight design method. In contrast, although LEOR also uses a threshold radius to limit the size of the candidate set, it does not use the local information of the node, which results in poor results. As far as LWOF is concerned, due to its fixed and small candidate area, it always maintains a small number of redundant data packets, but it brings an increase in delay. However, LDORRO can maintain a similar level of data packet redundancy while reducing time delay. This is due to the idea of dynamically increasing the candidate area. When there are fewer wake-up nodes, appropriately increasing the set of candidates will not increase the number of redundant data packets to a large extent, but it can well reduce the network transmission delay.

5.3. Energy Consumption. Finally, we analyze the performance of LDORRO in terms of energy consumption from two aspects: the overall energy consumption level of the network and the energy consumption difference between nodes in Figures 7 and 8. As the number of nodes increases, the total energy consumption of nodes increases. The total energy consumption for forwarding a certain number of data packets shows an upward trend in Figure 7. This is mainly due to the increase in the total number of nodes, which does not mean that the energy consumption level of each node has increased. For LDORRO, although the waiting delay of forwarding is reduced, it does not gain an advantage in total energy consumption. This is because LDORRO dynamically increases the size of the candidate set during the forwarding process, resulting in more nodes participating in the forwarding process. Therefore, more nodes will stay awake and receive related data packets. Although the energy consumption of sending the preamble of the sender is reduced, the energy consumption of the all the candidates is increased. Compared with LEOR, when the number of nodes increases, the benefits of the waiting

delay of LDORRO gradually lose the advantage of energy consumption, so the energy consumption is higher than that of LEOR.

In terms of energy consumption balance, we calculate the CDF function of the node's residual energy level after transmitting a certain number of data packets, as shown in Figure 8. It can be seen from the figure that LDORRO achieves better energy balance compared with LEOR. After forwarding 1000 data packets, the residual energy of nodes with LDORRO is kept between 4 mJ and 8 mJ. However, the residual energy levels of the nodes are different and the residual energy of the nodes is too high or too low with LEOR.

6. Conclusions

We propose a low-delay opportunistic routing with reducing overhead (LDORRO) in WSNs. First, we initialize the network by dividing it into different rings and adopt adaptive duty cycle to reduce transmission delay in this protocol. Then, we construct dynamic candidate sets for each node to reduce the overhead of data and control packets. Finally, we design the routing metric for the candidate coordination which is used to select the final forwarding node among candidates. The simulation results prove our progress in these areas. In this protocol, we only calculate the routing metric by residual energy and distance of nodes in a traditional way. In future work, we will consider combining artificial intelligence to optimize the design of routing metric.

Data Availability

The simulation data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 61821001 and 62090010.

References

[1] T. Dinh, Y. Kim, T. Gu, and A. V. Vasilakos, "An adaptive low-power listening protocol for wireless sensor networks in noisy environments," *IEEE Systems Journal*, vol. 12, no. 3, pp. 2162–2173, 2018.

[2] Z. Chen, A. Liu, Z. Li, Y.-j. Choi, and J. Li, "Distributed duty cycle control for delay improvement in wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 10, no. 3, pp. 559–578, 2017.

[3] O. Abedi and S. R. Kariznoi, "Load-balanced and energy-aware opportunistic routing with adaptive duty cycling for multi-channel Wsns," *The Journal of Supercomputing*, vol. 77, no. 2, pp. 1038–1058, 2021.

[4] I. Amdouni, C. Adjih, N. AitSaadi, and P. Muhlethaler, "Extensive experimentations on opportunistic routing in wireless sensor networks," *Sensors*, vol. 18, no. 9, p. 3031, 2018.

[5] A. Castagnetti, A. Pegatoquet, Trong Nhan le, and M. Auguin, "A joint duty-cycle and transmission power management for energy harvesting Wsn," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 928–936, 2014.

[6] N. Chakchouk, "A survey on opportunistic routing in wireless communication networks," *IEEE Communication Surveys and Tutorials*, vol. 17, no. 4, pp. 2214–2241, 2015.

[7] D. Chen, J. Deng, and P. K. Varshney, "On the forwarding area of contention-based geographic forwarding for ad hoc and sensor networks," in *Sensor and Ad Hoc Communications and Networks, 2005. IEEE SECON 2005. 2005 Second Annual IEEE Communications Society Conference on*, Santa Clara, CA, USA, 2005.

[8] H.-M. Chen, L. Cui, and G. Zhou, "A light-weight opportunistic forwarding protocol with optimized preamble length for low-duty-cycle wireless sensor networks," *Journal of Computer Science and Technology*, vol. 32, no. 1, pp. 168–180, 2017.

[9] R. W. L. Coutinho and A. Boukerche, "Pcon: a novel opportunistic routing protocol for duty-cycled Internet of underwater things," in *2019 IEEE Symposium on Computers and Communications (ISCC)*, Barcelona, Spain, 2019.

[10] E. Ghadimi, O. Landsiedel, P. Soldati, S. Duquennoy, and M. Johansson, "Opportunistic routing in low duty-cycle wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 10, no. 4, pp. 1–39, 2014.

[11] R. W. L. Coutinho, A. Boukerche, L. F. M. Vieira, and A. A. F. Loureiro, "A joint anypath routing and duty-cycling model for sustainable underwater sensor networks," *Sustainable Computing*, vol. 4, no. 4, pp. 314–325, 2019.

[12] A. Hawbani, X. Wang, Y. Sharabi, A. Ghannami, H. Kuhlani, and S. Karmoshi, "Lora: load-balanced opportunistic routing for asynchronous duty-cycled Wsn," *IEEE Transactions on Mobile Computing*, vol. 18, no. 7, pp. 1601–1615, 2019.

[13] T. Heimfarth, J. C. Giacomin, E. P. de Freitas, G. F. Araujo, and J. P. de Araujo, "Pax-Mac: a low latency anycast protocol with advanced preamble," *Sensors*, vol. 20, no. 1, p. 250, 2020.

[14] J. Niu, L. Cheng, Y. Gu, J. Jun, and Q. Zhang, "Minimum-delay and energy-efficient flooding tree in asynchronous low-duty-cycle wireless sensor networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 7–10, Shanghai, April 2013.

[15] A. Khan, N. Javaid, A. Sher, R. A. Abbasi, Z. Ahmad, and W. Ahmed, "Load balancing and collision avoidance using opportunistic routing in wireless sensor networks," in *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, Krakow, Poland, 2018.

[16] A. A. Lata and M. Kang, "A review on broadcasting protocols for duty-cycled wireless sensor networks," in *2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN)*, p. 2019, Zagreb, Croatia, 2019.

[17] A. A. Lata and M. Kang, "A survey on the evolution of opportunistic routing with asynchronous duty-cycled Mac in wireless sensor networks," *Sensors*, vol. 20, no. 15, p. 4112, 2020.

[18] G. Li, F. Li, T. Wang, J. Gui, and S. Zhang, "Bi-adjusting duty cycle for green communications in wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, 2020.

[19] D. Liu, Z. Cao, Y. He, X. Ji, M. Hou, and H. Jiang, "Exploiting concurrency for opportunistic forwarding in duty-cycled Iot networks," *ACM Transactions on Sensor Networks*, vol. 15, no. 3, pp. 1–33, 2019.

[20] R. S. Rathore, S. Sangwan, K. Adhikari, and R. Kharel, "Modified echo state network enabled dynamic duty cycle for optimal opportunistic routing in Eh-Wsns," *Electronics*, vol. 9, no. 1, p. 98, 2020.

[21] J. So and H. Byun, "Load-balanced opportunistic routing for duty-cycled wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 1940–1955, 2017.

[22] W. Wu, X. Wang, A. Hawbani, and T. Qureshi, "PORA: piggybacking-based opportunistic routing for asynchronous duty-cycled WSNs," in *2020 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*, Exeter, United Kingdom, 2020.

[23] D. Xu, W. Jiao, Z. Yin et al., "Maximizing throughput for low duty-cycled sensor networks," *Computer Networks*, vol. 139, pp. 48–59, 2018.

[24] X. Xiang, W. Liu, A. Liu, N. N. Xiong, Z. Zeng, and Z. Cai, "Adaptive duty cycle control–based opportunistic routing scheme to reduce delay in cyber physical systems," *International Journal of Distributed Sensor Networks*, vol. 15, no. 4, Article ID 155014771984187, 2019.

[25] H. Yoo, M. Shim, and D. Kim, "Dynamic duty-cycle scheduling schemes for energy-harvesting wireless sensor networks," *IEEE Communications Letters*, vol. 16, no. 2, pp. 202–204, 2012.

[26] M. U. F. Qaisar, X. Wang, A. Hawbani, A. Khan, A. Ahmed, and F. T. Wedaj, "TORP: load balanced reliable opportunistic routing for asynchronous wireless sensor networks," in *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Guangzhou, China, 2020.

[27] X. Shen, L. Liu, Z. Ni, M. Liu, B. Zhao, and Y. Shang, "Link-correlation-aware opportunistic routing in low-duty-cycle wireless networks," *Sensors*, vol. 21, no. 11, p. 3840, 2021.

[28] D. P. Kumar, T. Amgoth, and C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: a survey," *Information Fusion*, vol. 49, pp. 1–25, 2019.

[29] P. K. Donta, T. Amgoth, and C. S. R. Annavarapu, "Delay-aware data fusion in duty-cycled wireless sensor networks: a *Q*-learning approach," *Sustainable Computing: Informatics and Systems*, vol. 33, article 100642, 2022.

[30] X. Zhang, C. Wang, and L. Tao, "An opportunistic packet forwarding for energy-harvesting wireless sensor networks with dynamic and heterogeneous duty cycle," *IEEE Sensors Letters*, vol. 2, no. 3, pp. 1–4, 2018.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7579515, 11 pages
https://doi.org/10.1155/2022/7579515

Research Article

Recovering Latent Data Flow from Business Process Model Automatically

Sheng Ye[1,2] **Jing Wang**[1,2] **Sikandar Ali**[3] **Hasan Ali Khattak**[4] **Chenhong Guo**[1,2] **and Zhongguo Yang**[1,2]

[1]*School of Information Science and Technology, North China University of Technology, Beijing 100144, China*
[2]*Beijing Key Laboratory on Integration and Analysis of Large-Scale Stream Data, Beijing 100144, China*
[3]*Department of Information Technology, The University of Haripur, Haripur 22620, Khyber Pakhtunkhwa, Pakistan*
[4]*School of Electrical Engineering and Computer Science, National University of Sciences and Technology (NUST), Islamabad 44000, Pakistan*

Correspondence should be addressed to Sikandar Ali; sikandar@uoh.edu.pk
and Hasan Ali Khattak; hasan.alikhattak@seecs.edu.pk

Received 18 February 2022; Accepted 4 June 2022; Published 20 June 2022

Academic Editor: Yingjie Wang

Process-driven applications evolve rapidly through the interaction between executable BPMN (Business Process Modeling and Notation) models, business tasks, and external services. Given these components operate on some shared process data, it is imperative to recover the latent data by visiting relation, which is known as data flow among these tasks. Data flow will benefit some typical applications including data flow anomaly checking and data privacy protection. However, in most cases, the complete data flow in a business process is not explicitly defined but hidden in model elements such as form declarations, variable declarations, and program code. Some methods to recovering data flow based on process model analysis of source code have some drawbacks; i.e., for security reasons, users do not want to provide source code but only encapsulated methods; therefore, data flows are difficult to analyze. We propose a method to generate running logs that are used to produce a complete data flow picture combined with the static code analysis method. This method combines the simple and easy-to-use characteristics of static code analysis methods and makes up for the shortcomings of static code analysis methods that cannot adapt to complex business processes, and as a result, the analyzed data flow is inaccurate. Moreover, a holistic framework is proposed to generate the data flow graph. The prototype system designed on Camunda and Flowable BPM (business process management) engine proves the applicability of the solution. The effectiveness of our method is validated on the prototype system.

1. Introduction

In the discipline of Business Process Management and Automation, BPMN, based on ISO standards, is widely adopted as a modeling language for workflow and executable business process models [1]. When used as a common business process modeling language for domain experts and others in the industry, BPMN helps to better business-IT alignment.

The business process model explicitly describes the control flow that consists of events (depicted by circles), tasks (by rectangles), and gateways (by diamond shapes). The data flow that is represented by data objects (depicted by parallelogram) and is associated with tasks as their input or output, respectively, is often overlooked. Recently, many researchers are starting to take the data flow seriously and use it to create greater value especially, in the data privacy protection and data anomaly detection tasks [2]. Referring to Figure 1, the black part is the business process diagram, which illustrates the execution sequence of the business process. The data flow is an ordered sequence of bytes with a start and an endpoint, including an input and output flow. The data involved in a business process may be read and written sequentially by multiple tasks. In Figure 1, the entire data flow is represented by blue. The blue part denotes the data flow involved in the business process, which can be variables, forms, or even other data structures permitted by the BPMN2.0 standard.

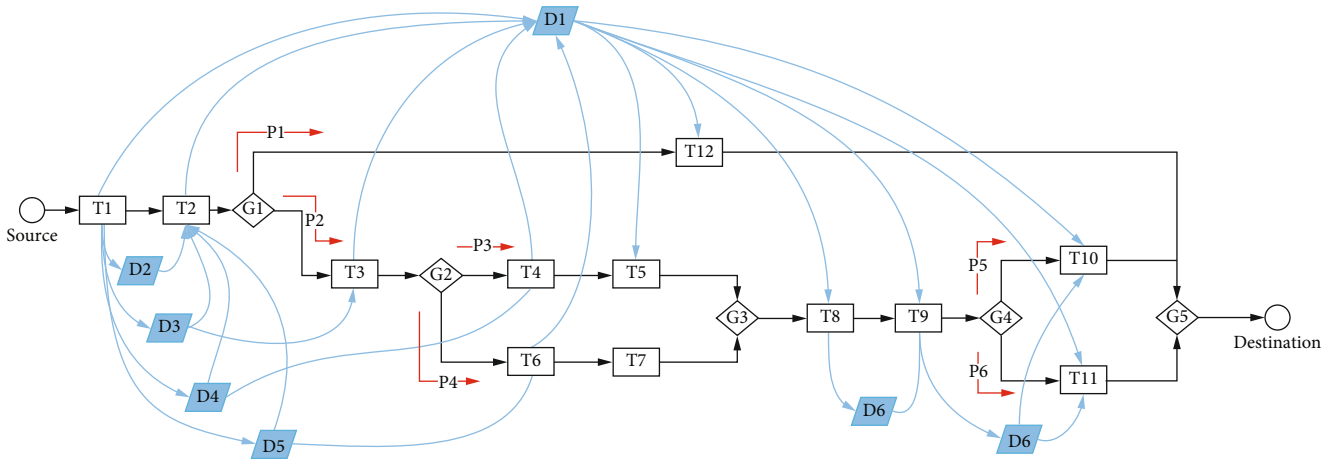


Figure 1: Sample diagram of complete data flow.

It is difficult to find all data flows in the business process. In related literature to BPM, there are some works [3] that are all about anomaly detection and processing of data flow in BPM. But unfortunately, most of them ignore the process of getting data flow from BPM. Schneid et al. [4] propose a method of using static code analysis tools to detect data flow in a simple business process without branches. The approach that writes rules through code analysis tools instead of human actions to find the data flow is partial, incoherent, cumbersome, and inaccurate when dealing with particularly complex BPM files [5]. We propose a new method to analyze data flow based on analyzing the reading and writing of data from logs of running processes. Static code analysis methods have some shortcomings such as inaccurate data flow detection, inability to adapt to complex business processes, and time-consuming. The method proposed in this paper not only solves these problems but also is efficient and convenient. Complete data flow requires complete business process logs. Some activities or tasks can generate data flow. Therefore, the challenge of this article is to ensure that every element of the business process is run once (or traveled along every path). As shown in Figure 1, two branch instances (P1, P2) are generated when the process engine passes through the G1 gateway, G2 gateway (P3, P4), and G4 gateway (P5, P6). Compared with previous work, this paper has the following three contributions:

(i) Suit for complex BPM: it is suitable for a variety of complex business process files, rather than a single branchless path as in previous experiments

(ii) A business process automation tool: automated deployment runs business processes faster and more accurately than manual judgment and simple static source code analysis

(iii) Complete data flow: it can analyze the complete-data flow by providing the executable file of the external agent task

We developed a tool (named BP-Dataflow) that consists of a front-end user interaction page, a business process automation running program, and a data flow drawing framework to generate a complete data flow diagram according to the necessary information submitted by users. Firstly, users need to submit corresponding BPM files, initial parameters, and some similar Jar packages. Moreover, the proposed program parses these files and parameters to drive the engine program that can be parsed by the process engine to run and generate log files. Finally, the drawing program will analyze the read/write records of data in the log to automatically draw a complete data flow graph.

The rest of this paper is structured as follows: Section 2 discusses the related work. Section 3 introduces the proposed method to implement the system. Section 4 evaluates the proposed method's effectiveness and applicability. Finally, Section 5 concludes the paper and gives a brief outlook on future work.

2. Related Work

2.1. Recovery of Data Flow. Most of the existing research on business process management focuses on the perspective of workflows and ignores the importance of the data flow perspective. However, various constraints in business processes and workflows often depend on the correctness of data. If the control flow is correct, then the data flow may not be correct; therefore, it is very important to analyze and restore the data flow in business processes and workflows [6]. Moreover, there is less research on data flow recovery. Recently, some studies [4] have been conducted to analyze the problem from the perspective of data flow recovery. By considering an integrated view, data flow recovery has been studied by Schneid et al. [4]. The main idea is to restore the data flow in the business process by merging the calling diagram of external service and the resulting diagram of associated data operation into the control flow of the process model. Chaim et al. [5] model data flow tests as a data flow analysis framework to quickly discover data flow relationships using efficient algorithms. Guo et al. [7] proposed a new data perspective of workflow management and a mathematical technique to solve the problem of data exchange in business process centralization in a dynamic environment to better recover data flow. In another study, Schneid et al. [8]

presented an integrated DFA diagram based on process models and artifacts (such as source code or user forms) to indicate operational relationships between data flows and nodes. Ji et al. [9] proposed a method to analyze data flow in the BPEL specification business process and ensure the correctness of data flow based on XCFG (eXtended Control Flow Graph). Amme et al. [10] proposed a CSSA method to extract data flow information from WS-BPEL, the Web Services Business Process Execution Language.

However, the above-mentioned methods own some obvious disadvantages. First of all, they research the anomaly detection and processing of data flow, but they do not elaborate on the method of data flow acquisition. Secondly, majority of these methods of obtaining data flow are through static code analysis, which requires a lot of time to fill in matching rules and is difficult to verify the accuracy systematically. Therefore, the integrity and accuracy of the data flow obtained through these methods are questionable.

2.2. Application of Data Flow. Modeling and validation of data flow are very important for anomaly detection; thus, Chadli et al. [2] explain three approaches that are used for detecting data flow anomalies and its proper method and tools. Tao and Fang [11] opt for workflow nets with tables (WFT-nets) to model workflow systems and detect inconsistent data. Liu et al. [12] have proposed a Petri net-based approach to model and analyze data flows. Ramon-Cortes et al. [6] build a Distributed Stream Library supporting the integration of workflow and data flow to meet the needs of new Data Science workflows. Xiang et al. [13] have proposed a PN-DOS model to reduce the accessibility of graphs to quickly detect data flow errors and ensure the correctness of business processes. Zhai et al. [14] have proposed a novel approach of data flow optimization to determine the optimal partition of data flow in BEPL processes, which is complex and accurate. Kabbaj et al. [15] have introduced an approach to detect data flow errors in business process models by validating and correcting fragments of the model as the model is modeled.

In a nutshell, there are a lot of literature on data flow application, such as data flow anomaly and privacy protection, so data flow is very important for some applications. However, the recovery of data flow is described in insufficient detail in the previous work. Therefore, this article targets those gaps in the recovery of data flow. The framework proposed in this article is described in detail in the next section.

3. Method

3.1. Framework. Figure 2 shows the overall operating architecture. This paper designs the architecture from the point of view of full automation. Moreover, a method is proposed to automatically generate the executable code depending on parsing the BPM file. The framework consists of three main steps: entering related files and parameters, algorithms for generating executable code, and generating a complete data flow diagram. In the first step, the user submits a standard BPM file, initial variable, and external agent files. Finally, the Pydotplus-Python library is utilized to generate a complete data flow diagram by analyzing process running logs.

3.2. Basic Concept. The business process engine used for the experiment in this paper is Flowable, and the tools presented in this article are capable to work with the most popular Camunda engine. BPM defines multiple data types, such as variables, forms, DMN, and so on. These can be resolved using specific matching rules. It is important to note that different activities in the business process have unique ways of reading and writing data.

(i) User task: as the name implies, the user assigned to the task must view the corresponding data view before deciding whether to complete the task. In some cases, completing the task requires writing data such as variables and forms. These can be explicitly parsed in a file or analyzed from the running log

(ii) Service task: the service task that is independent of the engine is customized by the user, which requires the user to provide the executable program that can be called so that the algorithm proposed in this paper can run smoothly and accurately find the corresponding data flow

(iii) Script task: it is written in scripting languages like JavaScript and Python. The solution is similar to a service task and requires the user to provide the corresponding execution file

(iv) DMN task (business rule task): as with the above activities, which are independent of the engine, the user is required to provide the corresponding DMN file. It should be noted that all data involved in this task will be read once and then written again after processing with corresponding rules. Therefore, it is better to analyze the data flow of this task

3.3. Work Details. This paper presents an algorithm, i.e., Algorithm 1, for combining static code analysis and dynamic log analysis. The static approach uses the regular expressions to match the explicit data flow as shown in Figure 3 while the dynamic approach addresses the implicit data flow. Data flow read/write relationships in logs can also be extracted using regular expressions. In order to better adapt to each BPM file, there are many points to consider. The key point is how to ensure that the complex path is covered. In fact, almost all the papers on recovering data flow do not consider multipath, since the BPM file used for the experiment is relatively simple. In this paper, we use a graph structure to temporarily store information in business processes and data flow information. Pydotplus, Python's powerful drawing library, outputs a complete data flow graph with the graph as input. In the following sections, I describe the three steps of the framework.

First step: the front-end interface: this paper uses the React front-end framework to write an interface to interact with users. The React framework is extremely powerful, with many ready-made component libraries that are closely aligned with background operations. The display interface is shown in Figure 4. Users just need to submit standard BPM files and process execution variables and independent external proxy class files.

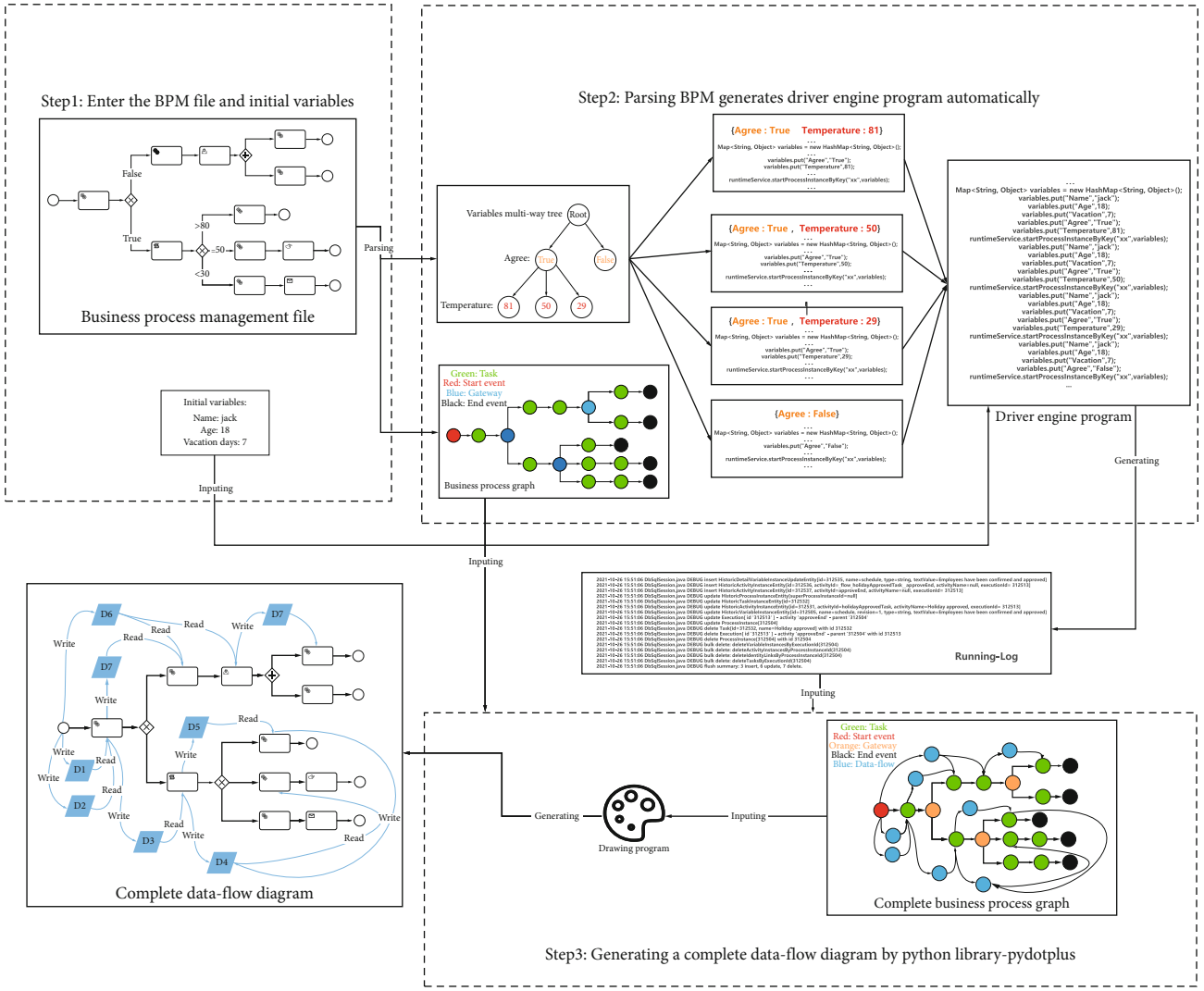


Figure 2: An automatic generating BPM data flow tool: BP-Dataflow.

Second step: parsing BPM to generate driver-engine program automatically: we need to parse the ID of process, task, and gateway in the business process file to facilitate the program to drive the engine.

$$\begin{aligned} &<\text{process id} = ''\text{holidayRequest}''\cdots> \\ &\text{runtimeService.startProcessInstanceBykey}\left(''\text{holidayRequest}'', \text{variables}\right); \end{aligned} \tag{1}$$

One graph can represent only one business process in a swimming pool. We use the API (Algorithm 1 line 1) provided by the engine to store the business process sequence in the graph structure as shown in Figure 5.

When a business process is running in the engine, only one instance is generated at a time. It is important to generate multiple instances to traverse all paths of the BPM. The type of gateway determines how many instances are launched when you code. The following describes two common gateways.

Exclusive gateways control different branches by depending on the value of a variable (Algorithm 1 line 7). This paper uses the multifork tree to store these variables. Later sections explain why this data structure is used, which is defined here as the multifork tree of variables. Boolean has only two branches, with true and false as the two child nodes of the variable multifork tree, as shown in Figure 6. It needs to find the corresponding condition and value in the file when exclusive gateways are of numeric types. As shown in Figure 7, when the condition of branching is greater than 80, the program automatically adds 1 to this value to get 81. If the branching condition is less than 40, it subtracts one from the value to get 39. If it is a range condition, the

Input: BPMN *bp*, Hashmap *iv*,Graph *cg*,Muti-Tree *mt*,JavaFile *jf*
Output: RunningLog *lg*
1: *cg* = *bp*.Deployment().FindSequence()//Deploy the BPMN file to get the process order
2: **while***temp* = *bp*.ReadLine() **do**
3: *cg*.AddInformation(temp.FindID)
4: if *temp*.ContainStaticData() **then**
5: *cg*.AddDataflow(*temp*.EtxractDataflow())
6: **end if**
7: **if***temp*.JudgeGateway() **then**
8: *mt*.Add(*temp*.GatewayInformation)
9: **end if**
10: **end while**
11: **for***m* = *mt*.TraverseAll() **do**
12: *iv*.Add(*m*)
13: WriteFile(Engine.RunAPI(*iv*), *jf*)
14: **end for**
15: **if** Engine.HasUserTask() **then**
16: WriteFile(Engine.CompleteAPI(), *jf*)//Complete human tasks
17: **end if**
18: WriteFile(FixedCode, *jf*)//Write the fixed template code to the Java file
19: RUN(*jf*)//Run the Java file

Algorithm 1: Generating driver-engine program automatically.

```
<flowable:executionListener expression=" ${execution.getVariable( 'employee' )}" event=" start" />
<flowable:executionListener expression=" ${execution.getVariable( 'nr0fHolidays' )}" event=" start" />
<flowable:executionListener expression=" ${execution.getVariable( 'description' )}" event=" start" />
<flowable:executionListener expression=" ${execution.getVariable( 'schedule' ,' Manager approval has been completed' )}" event=" end" />
```

Figure 3: Explicit data streams in BPMN files.

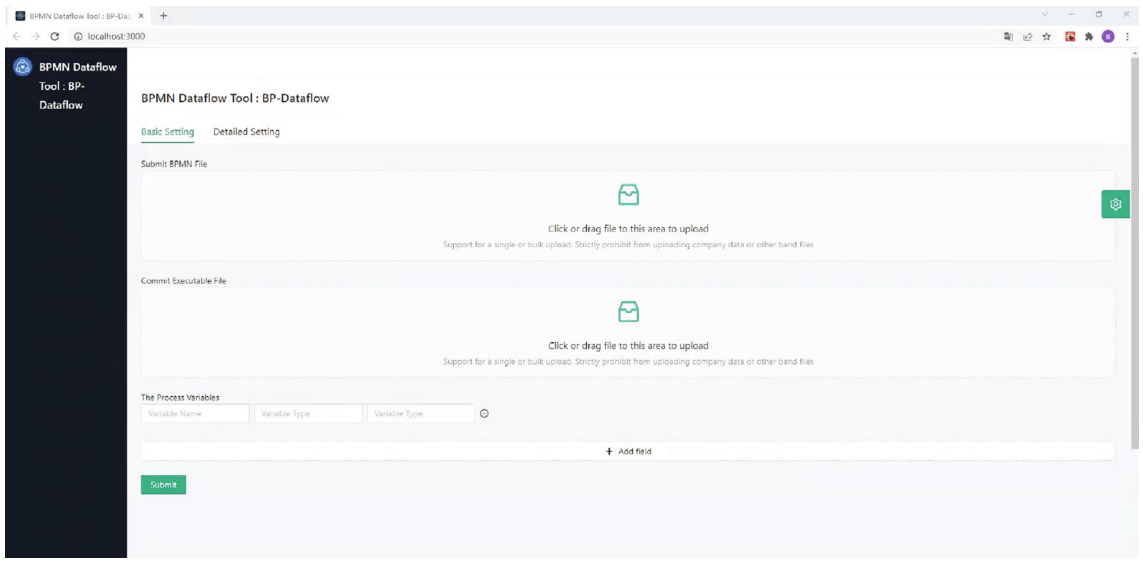


Figure 4: The front-end interface.

program will automatically take the value that meets the condition. The condition for branching is equal to 50, so it will take 50, eventually storing each of these variables on the children of the variable multifork tree.

Parallel gateways, as the name implies, allow the engine to continue execution separately along each branch. The engine automatically generates the number of instances of branches; therefore, there is no need for the variable multifork trees to assist storage. One of the most important purposes of this paper is to convert variables involved in the original business process into the multifork tree of variables as shown in Figure 8.

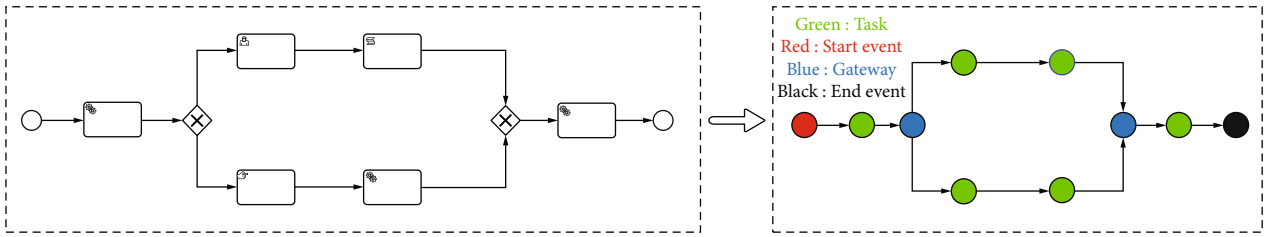


Figure 5: Store business process sequence by using a graph.

```
<exclusiveGateway id=" decision" />
<sequenceFlow id=" s2" sourceRef=" decision"
targetRef=" externalSystemCall" >
    <conditionExpression xsi:type=" tFormalExpression" >
        <![CDATA[
        ${approved}
        ]]>
    </conditionExpression>
</sequenceFlow>
<sequenceFlow id=" s3" sourceRef=" decision" targetRef=" sendRejectionMail" >
    <conditionExpression xsi:type=" tFormalExpression" >
        <![CDATA[
        ${!approved}
        ]]>
    </conditionExpression>
</sequenceFlow>
```

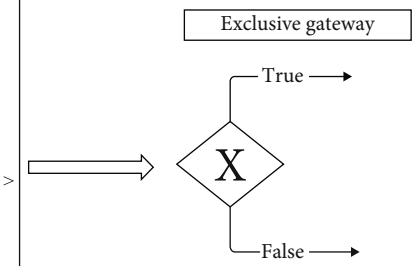


Figure 6: Parse a gateway with a Boolean branching condition.

```
<sequenceFlow id=" d1" sourceRef=" decision"
targetRef=" externalSystemCall" >
    <conditionExpression xsi:type=" tFormalExpression" >
        <![CDATA[
        ${temperature>80}
        ]]>
    </conditionExpression>
</sequenceFlow>
<sequenceFlowid=" d2" sourceRef=" decision" targetRef=" sendRejectionMail" >
    <conditionExpression xsi:type=" tFormalExpression" >
        <![CDATA[
        ${temperature<40}
        ]]>
    </conditionExpression>
</sequenceFlow>
<sequenceFlowid=" d3" sourceRef=" decision" targetRef=" sendRejectionMail" >
    <conditionExpression xsi:type=" tFormalExpression" >
        <![CDATA[
        ${temperature=50}
        ]]>
    </conditionExpression>
</sequenceFlow>
```

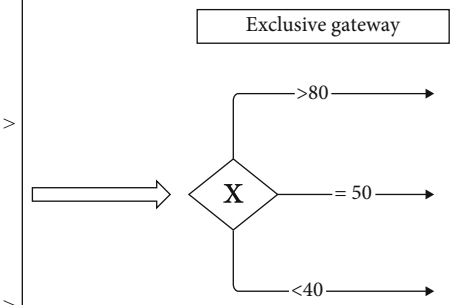


Figure 7: Parse a gateway with a numerical branching condition.

Finally, information about each path from the root node to the child node in the variable multifork tree, respectively (line 11), is {initial variables, approved: true, temperature: 81}, {initial variables, approved: true, temperature: 29}, {the initial variable, approved: true, temperature: 50}, and {initial variables, approved: false}. Call engine API (line 13) to launch 4 instances that, respectively, carry four variables' sequence to go through all the paths of the business process; other running code is fixed and can be automatically generated by using templates (line 18). By analyzing the log files generated after the process runs (line 19), we can easily find which activities read and write data that can be variables, forms, etc. As shown in Figure 9, the activityId is the unique identification number of the activity, Revision is the number of times the variable was written to clarify how many times it was written, while the textValue is what was written. Finally, the read and write of data are stored in the data flow HashMap for the drawing program to generate a data flow graph.

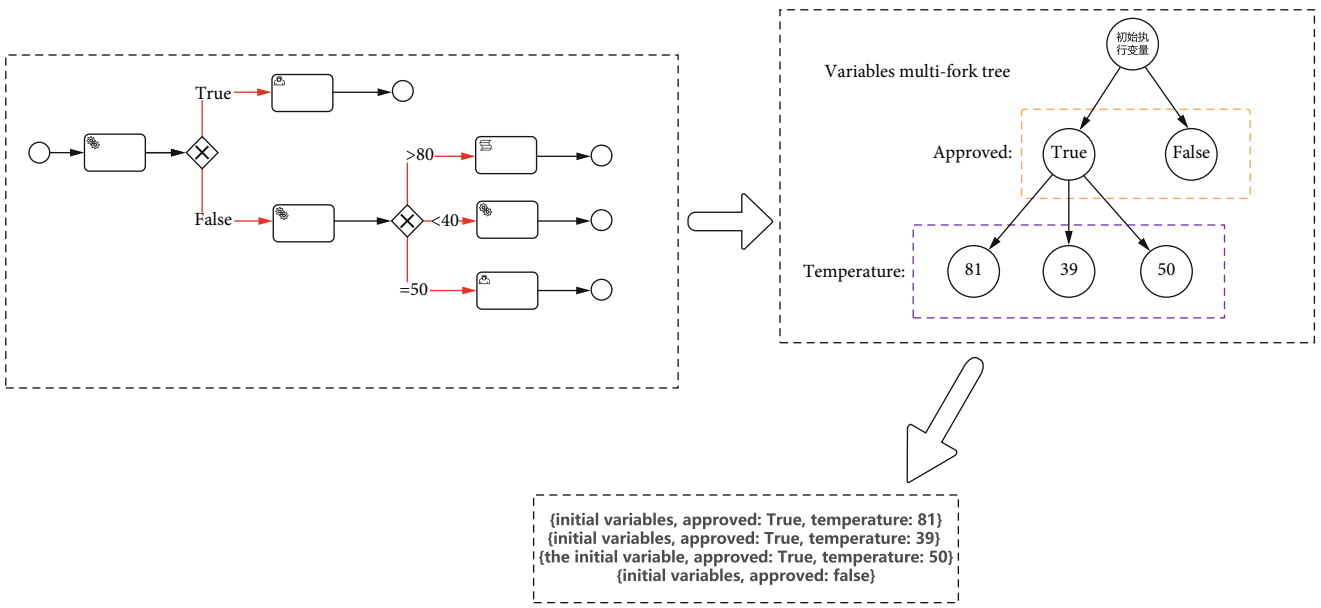


Figure 8: Traverse the path information from the root node to each leaf node of a variable multifork tree.

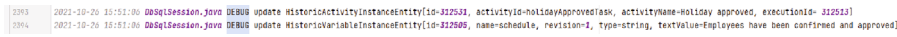


Figure 9: The data flow in running logs.

Third step: automatic plotting program: this drawing program uses Pydotplus—a Python drawing library. Pydotplus accepts the business process sequence graph and data flow HashMap mentioned in the preceding section as input and then automatically generates a complete business process model data flow diagram using the front-end interface.

4. Evaluation

To assess the effectiveness of our system in business processes, we test 5 real-world business process cases in which two cases are presented in detail. We evaluate the system in terms of two sets of measures:

(i) Accuracy. What is the fraction of correctly analyzed data flow in the complete data flow (precision rate)?

(ii) Performance. Can we cope with the real-world business process? How long does it take when submitting a BPM File to generate a data flow diagram need?

4.1. Evaluation Process Cases. As shown in Table 1, we have carefully listed the amount of data flow of five real-world business processes for subsequent analysis and evaluation.

The first case to verify the accuracy of the algorithm is the example mentioned in the official Flowable manual (business process for employees to apply for leave) as shown in Figure 10. As the example is too simple to have an obvious data flow, we add some read/write operations of data flow without changing the real business logic. The user needs to submit three initial variables including the employee's name, number of days leave requested, and reason for requesting leave, when the process runs. The manager needs to look at the information submitted by the employee to decide whether to agree or disagree. Eventually, the data flow diagram is shown in Figure 11.

The second case is the insurance business process as shown in Figure 12. The process is complex, with multiple swim lanes, script tasks, business rules tasks, and subprocesses. These elements do not affect the use of the algorithm proposed in this paper. Finally, the generated data flow diagram is shown in Figure 13.

4.2. Accuracy

4.2.1. Experimental Setup. To evaluate the precision of the proposed system, we compare the attributes of all aspects of the data flow that are automatically produced by the system with the values judged by the human in the above table. Then, we calculate five formulas to assess the accuracy of the system.

$$\text{PathNumberPrecisionRate} = \frac{\text{SystemToCalculatePath}}{\text{ActualPath}}, \tag{2}$$

$$\text{DataNumberPrecisionRate} = \frac{\text{SystemToCalculateData}}{\text{ActualData}}, \tag{3}$$

$$\text{Data} - \text{read} - \text{flowPrecisionRate} = \frac{\text{SystemToCalculateDataRead}}{\text{ActualDataRead}}, \tag{4}$$

Table 1: The evaluation index of 5 real business process.

Business process name	Path number	Data number	Data-read-flow number	Data-write-flow number	Data flow number
Employees apply for vacation	2	5	7	7	14
Customer insurance	5	9	9	12	21
The ship trajectory	8	11	12	13	25
Personalized intelligent medical care	12	8	8	11	19
Transportation of dangerous chemicals	13	16	14	18	32

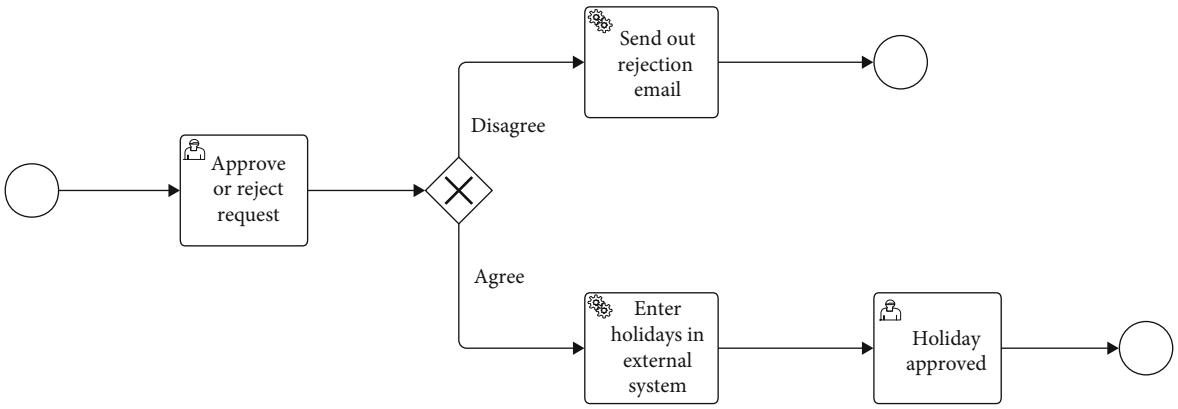


Figure 10: Employee leave process.

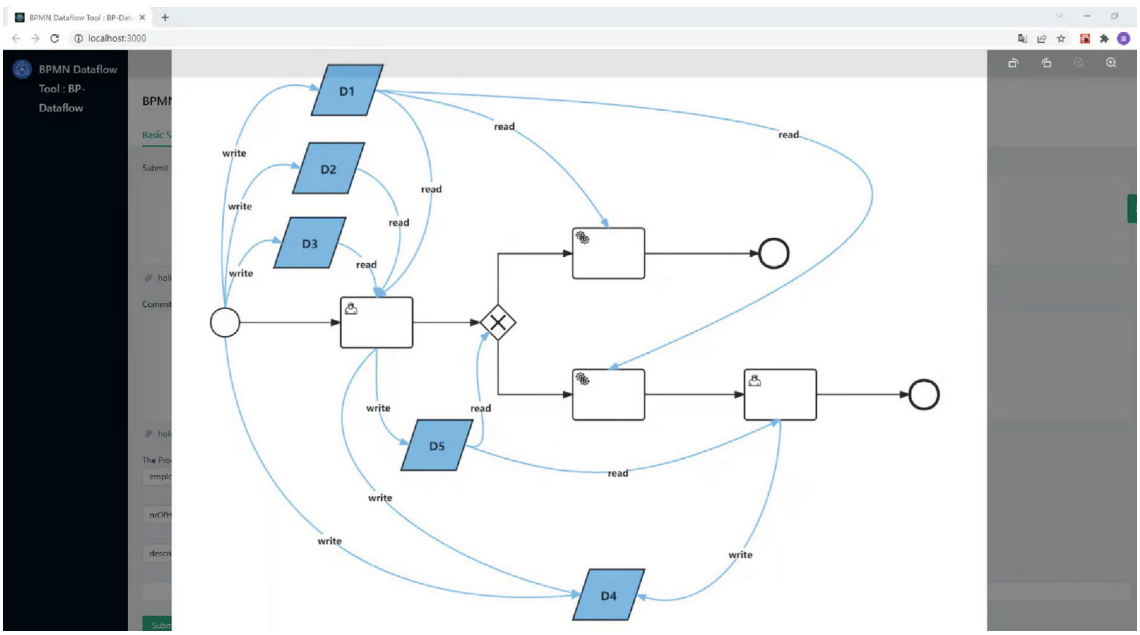


Figure 11: Data flow diagram of the employee leave process.

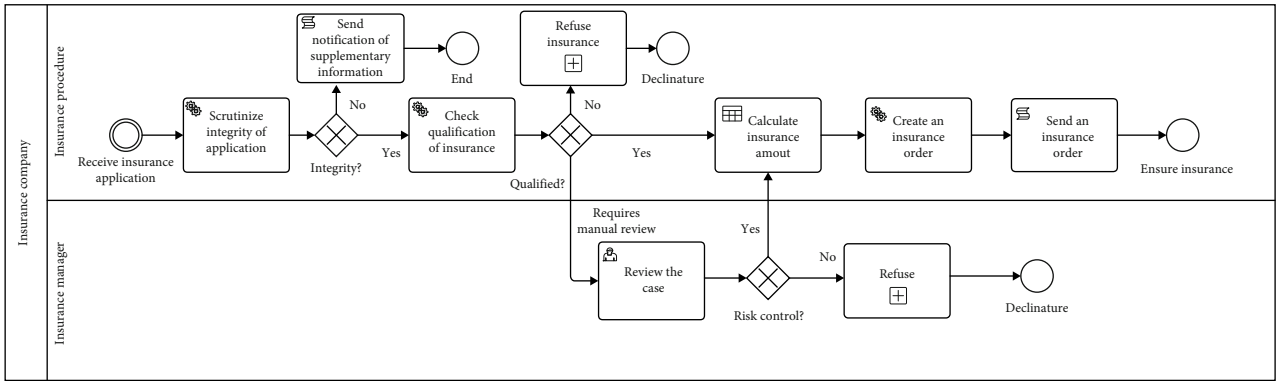


Figure 12: Insurance business process.

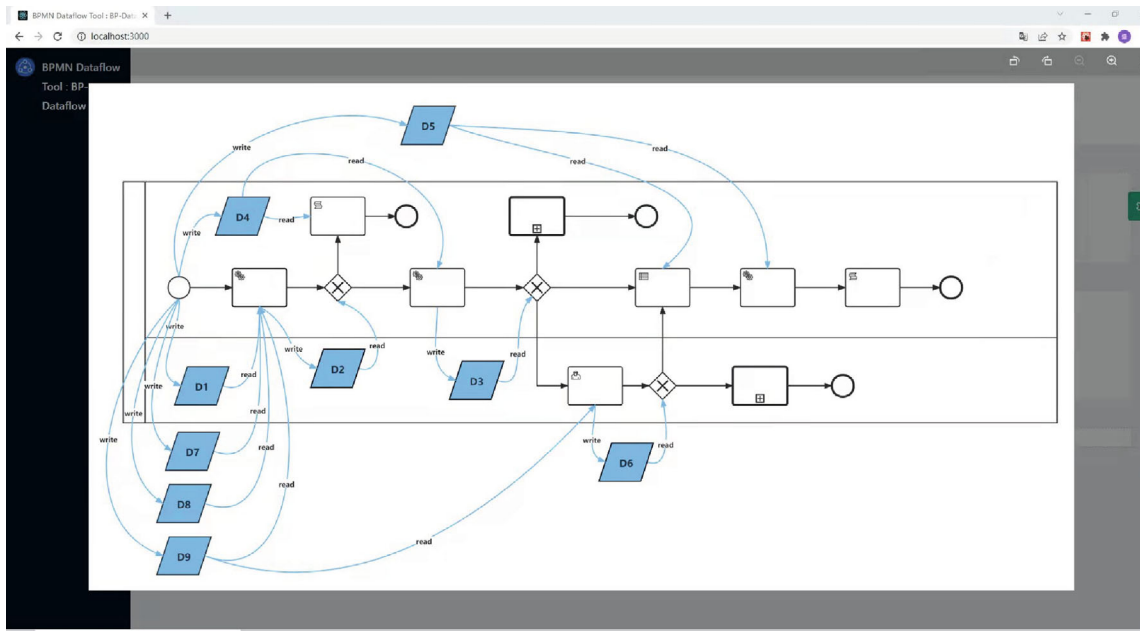


Figure 13: Data flow diagram of insurance business process.

$$\text{Data} - \text{write} - \text{flowPrecisionRate} = \frac{\text{SystemToCalculateDataWrite}}{\text{ActualDataWrite}}, \tag{5}$$

$$\text{DataflowPrecisionRate} = \frac{\text{SystemToCalculateDataflow}}{\text{ActualDataflow}}. \tag{6}$$

The algorithm proposed in this paper guarantees full path coverage, which is a necessary condition for data-flow integrity. As shown in Equation (2), it is used to measure the accuracy of path coverage. The other four equations are designed to evaluate the accuracy of the data flow in more detail.

4.2.2. Findings. As shown in Table 2, the test results for each of the five business process examples are the same as the real value with 100 percent accuracy. Based on these numbers, we can conclude that the number of paths and data are correct, which makes the precision of the system 100%.

4.3. Performance

4.3.1. Experimental Setup. We intend to evaluate the performance of the proposed system through total runtime that includes generating a data flow diagram from user submission to the system.

4.3.2. Findings. We first focus on the time spent on running the business process. It is important to understand that the complexity of business processes is a combination of the multipath and data flow complexity of the business processes. Furthermore, running is directly proportional to the complexity of the business process. From the results, we can see that as business processes become more complex, they take longer to run.

Table 2: Statistics of various indicators and running time of the data flow.

Business process name	Calculate path number	Actual path number	Calculate data number	Actual data number	Calculate data-read-flow number	Actual data-read-flow number	Calculate data-write-flow number	Actual data-write-flow number	Calculate data flow number	Actual data flow number	Run times (s)
Employees apply for vacation	2	2	5	5	7	7	7	7	14	14	4.37
Customer insurance	5	5	9	9	9	9	12	21	21	21	5.21
The ship trajectory	8	8	11	11	12	12	13	13	25	25	6.22
Personalized intelligent medical care	12	12	8	8	8	8	11	11	19	19	5.77
Transportation of dangerous chemicals	13	13	16	16	14	14	18	18	32	32	8.32

Based on the analysis, we can conclude that by comparing with piecing together a complete data flow diagram from the data flow analyzed by each independent task, it is more accurate and efficient to execute the business process by automatically writing the executable program to get the data flow diagram.

5. Conclusion and Future Work

Previous work on the data flow is limited to static code analysis, which is not accurate and time-consuming. Based on previous work, this paper proposes a method to recover the data flow in the business process by combining static code analysis technology and dynamic running log analysis method, which not only has high accuracy and short time, but also, the tool developed according to this method is simple and efficient. This hybrid architectural approach is based on the Flowable engine and platform. Of course, with some modifications, we can run BPM files that support the Camunda engine. Essentially, their underlying API is the same, but the namespace in the XML is different. In terms of evaluation, the paper uses 5 real-world business processes, and the results are promising.

In summary, BPM files are composed of basic and common BPM elements that are analyzed in this paper. However, some tasks and events defined by the entire BPMN2.0 specification have not been analyzed here, such as intermediate events, boundary events, and tasks such as Web tasks, shell tasks, and listeners. This is also the direction of our future work, as we continue to refine the architecture to fully adapt to all specifications defined by BPMN2.0.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Key R&D Plan (No. 2018YFB1402500), the Key Program of the National Natural Science Foundation of China (No. 61832004), and the International Cooperation and Exchange Program of the National Natural Science Foundation of China (No. 62061136006).

References

[1] X. Kechagioglou, R. Lemmens, and V. Retsios, "Sharing geoprocessing workflows with Business Process Model and Notation (BPMN)," in *Proceedings of the 2019 2nd International Conference on Geoinformatics and Data Analysis*pp. 56–60, Prague, Czech Republic, 2019.

[2] N. Chadli, M. I. Kabbaj, and Z. Bakkoury, "Detection of data-flow anomalies in business process an overview of modeling approaches," in *Proceedings of the 12th International Conference on Intelligent Systems: Theories and Applications*pp. 1–6, Rabat, Morocco, 2018.

[3] N. Trčka, W. M. Van der Aalst, and N. Sidorova, "Data-flow anti-patterns: discovering data-flow errors in workflows," in *International Conference on Advanced Information Systems Engineering*, pp. 425–439, Springer, Berlin, Heidelberg, 2009.

[4] K. Schneid, H. Kuchen, S. Thöne, and S. Di Bernardo, "Uncovering data-flow anomalies in bpmn-based process-driven applications," in *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, pp. 1504–1512, New York, United State, 2021.

[5] M. L. Chaim, K. Baral, J. Offutt, M. Concilio, and R. P. Araujo, "Efficiently finding data flow subsumptions," in *2021 14th IEEE Conference on Software Testing, Verification and Validation (ICST)*, pp. 94–104, Porto de Galinhas, Brazil, 2021.

[6] C. Ramon-Cortes, F. Lordan, J. Ejarque, and R. M. Badia, "A programming model for hybrid workflows: combining task-based workflows and dataflows all-in-one," *Future Generation Computer Systems*, vol. 113, no. 7, pp. 281–297, 2020.

[7] X. Guo, S. X. Sun, and D. Vogel, "A dataflow perspective for business process integration," *ACM Transactions on Management Information Systems (TMIS)*, vol. 5, no. 4, pp. 1–33, 2015.

[8] K. Schneid, S. Di Bernardo, H. Kuchen, and S. Thöne, "Data-Flow analysis of BPMN-based process-driven applications: detecting anomalies across model and code," *ERCIS Working Paper*, vol. 2021, no. 38, 2021.

[9] S. Ji, B. Li, and P. Zhang, "XCFG based data flow analysis of business processes," in *2019 5th International Conference on Information Management (ICIM)*, pp. 71–76, Cambridge, UK, 2019.

[10] W. Amme, A. Martens, and S. Moser, "Advanced verification of distributed WS-BPEL business processes incorporating CSSA-based data flow analysis," *International Journal of Business Process Integration and Management*, vol. 4, no. 1, pp. 47–59, 2009.

[11] X. Tao and X. Fang, "Detecting data inconsistency based on workflow nets with tables," *IEEE Access*, vol. 9, pp. 81740–81749, 2021.

[12] C. Liu, Q. Zeng, H. Duan et al., "Petri net based data-flow error detection and correction strategy for business processes," *IEEE Access*, vol. 8, pp. 43265–43276, 2020.

[13] D. Xiang, G. Liu, C. Yan, and C. Jiang, "Detecting data-flow errors based on Petri nets with data operations," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 1, pp. 251–260, 2018.

[14] Y. Zhai, H. Su, and S. Zhan, "A data flow optimization based approach for BPEL processes partition," in *IEEE International Conference on e-Business Engineering*pp. 410–413, Hong Kong, China, 2007.

[15] M. I. Kabbaj, A. Bétari, Z. Bakkoury, and A. Rharbi, "Towards an active help on detecting data flow errors in business process models," *International Journal of Computer Science and Applications*, vol. 12, no. 1, pp. 16–25, 2015.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 5152598, 15 pages
https://doi.org/10.1155/2022/5152598

Research Article

An Efficient Computing Offloading Scheme Based on Privacy-Preserving in Mobile Edge Computing Networks

Shanchen Pang[1], **Huanhuan Sun**[1], **Min Wang**[2], **Shuyu Wang**[1], **Sibo Qiao**[1] and **Neal N. Xiong**[3]

[1]*Department of Computer Science and Technology, China University of Petroleum, Qingdao 266580, China*
[2]*Department of Control Science and Engineering, China University of Petroleum, Qingdao 266580, China*
[3]*Department of Mathematics and Computer Science, Northeastern State University, Tahlequah, OK, USA*

Correspondence should be addressed to Shanchen Pang; pangsc@upc.edu.cn

Received 29 December 2021; Accepted 18 May 2022; Published 14 June 2022

Academic Editor: Amrit Mukherjee

Computation offloading is an important technology to achieve lower delay communication and improve the experience of service (EoS) in mobile edge computing (MEC). Due to the openness of wireless links and the limitation of computing resources in mobile computing process, the privacy of users is easy to leak, and the completion time of tasks is difficult to guarantee. In this paper, we propose an efficient computing offloading algorithm based on privacy-preserving (ECOAP), which solves the privacy problem of offloading users through the encryption technology. To avoid the algorithm falling into local optimum and reduce the offloading user energy consumption and task completion delay in the case of encryption, we use the improved fast nondominated sorting genetic algorithm (INSGA-II) to obtain the optimal offloading strategy set. We obtain the optimal offloading strategy by using the methods of min-max normalization and simple additive weighting based on the optimal offloading strategy set. The ECOAP algorithm can preserve user privacy and reduce task completion time and user energy consumption effectively by comparing with other algorithms.

1. Introduction

The rapid development of the Internet of things leads to the increasing number of mobile devices and the explosive growth of various new mobile applications. These new types of applications (such as driverless cars, virtual reality, and face recognition) usually require intensive computing with high energy consumption [1–3]. However, the limited computing resources of mobile user equipment (UE) have brought challenges to the operation of new types of applications.

In order to solve the above challenges, mobile edge computing [4, 5] provides cloud computing capabilities for UEs on the network edge. Edge cloud is a cloud computing platform built on edge infrastructure. Edge cloud, central cloud, and IoT terminal form an end-to-end technical framework of cloud edge end three-body cooperation. Because edge cloud computing provides computing and network coverage nearby, the generation, processing, and use of data occur within a very close range from the data source, so receiving and responding to terminal requests have a low delay. For example, edge cloud computing applications in interactive live broadcasting. The media stream of the anchor is pushed to the nearest edge node, transcoded directly at the edge node, and then the transcoded media stream is distributed to the CDN edge node. When there is user access, the content is returned nearby. The services based on edge nodes, the upstream and downstream content push of live streams, and transcoding processing do not need to return to the cloud center, which greatly reduces the service delay and improves the interactive experience. At the same time, the edge processing architecture also saves the bandwidth cost. Resource-constrained UEs can offload tasks to edge servers, so MEC can achieve low latency and

high bandwidth to improve the quality of service and user experience [6].

However, computing tasks need to be offloaded to edge server (ES) through wireless link, which causes additional delay and energy consumption. In addition, ESs have limited resources different from traditional cloud computing centers [7]. Therefore, the offloading decision of computing tasks has become a key issue to achieve efficient offloading [8]. Chen et al. [9] studied the multiuser MEC system under in wireless interference environment, and a distributed efficient computing offloading algorithm has been proposed to achieve the Nash equilibrium. In [10], a new method of user collaborative computing offloading has been proposed to minimize energy consumption under the constraint on computing delay. Compared with [9, 10], the offloading problem was formalized as a multiobjective optimization problem in [11], and they try to find a compromise between delay and energy consumption. However, when UEs offload too many tasks to the same edge server, Chen et al. [9–11] neglected that the edge server may be overloaded, while other servers are in a light load state.

To solve the problem of load imbalance, Wei et al. [12] configured a data buffer for the MEC server to store data that cannot be executed immediately. Similar to [12], the problem of MEC server overload in [13] was solved by setting buffer queues on the mobile device side and the edge server side, respectively. In addition to queuing mobile user requests, we can also choose to reject and postpone user requests to decrease the load of MEC [14]. However, service interruption increases task waiting time and execution time, which reduce the quality of service for users. Therefore, it is critical to maximize system performance for task offloading in ultradense networks and balance the load of MEC servers.

On the other hand, due to the openness of wireless links, the task is prone to be exposed to external threats in the offloading process, which leads to the problem of privacy disclosure. For example, malicious eavesdroppers can eavesdrop on computing data offloaded by IoT devices. Therefore, the confidentiality of privacy is another key issue we need to consider [15]. There are currently two technologies: (1) one is physical layer security technology, which uses the status information about the wireless channel to effectively distinguish between legitimate users and eavesdroppers, to achieve information encryption; (2) the other is data encryption technology, which uses encryption algorithms, and the encryption key turns the plaintext into ciphertext. In this article, considering that the eavesdropper's eavesdropping ability and instantaneous channel state information are difficult to obtain, we use data encryption technology. In [16], a broadcast encryption based on anonymous attributes has been proposed to achieve an efficient and secure data sharing system. In [17], in order to ensure the security requirements of workflow intermediate data, the encryption algorithm and the hash function were sequentially applied to the output data of the task, which enables the implementation of the confidentiality service and integrity service. Xiong et al. and Chen et al. [16, 17] both researched on the security of cloud computing. In [18], the security of mobile edge computing has been considered, and the transmitted data were encrypted to prevent data from being threatened by the external world, but the impact of data size on encryption and decryption time was not considered. For the single server scenario, Wu et al. [19] proposed a joint optimization scheme of data confidentiality and computing offload to minimize the total delay of completing user computing requirements. Different from the above scenario, for the multiuser multicell MEC scenario in this paper, we need to use security services to protect the privacy of users.

To solve the above problems, we propose an efficient offloading method based on privacy-preserving. The main contributions of this paper can be summarized as follows.

(i) We introduce hybrid encryption technology to encrypt the offloaded data to protect user privacy and ensure the confidentiality of transmitted data. This encryption technology combines the encryption advantages of AES and RSA to improve the security and the speed of encryption

(ii) The load mean variance is proposed to evaluate the current load situation of edge servers and avoid overload or light load of some servers

(iii) We propose an improved NSGA-II algorithm (INSGA-II) to reduce the UE energy consumption and task completion delay and improve the system performance by introducing logistic chaotic sequence

The rest of the paper is organized as follows. Firstly, we review the relevant researches in the Section 2. In Section 3, we present the system model and the formation of the problem. In Section 4, we propose an efficient computing offloading method based on privacy-preserving. In Section 5, we present the simulation results. Finally, Section 6 summarizes the paper.

2. Related Work

In recent years, MEC as an emerging technology has attracted more and more attention [20–22], especially the problem of computing offloading of MEC. Most of the existing researches take the delay, energy consumption, weighted sum of energy consumption, and delay as the performance index of computing offloading. For delay-based computational offloading, to obtain the optimal task scheduling strategy, Liu et al. [23] proposed an efficient one-dimensional search algorithm to solve the problem of power constrained delay minimization. Considering the collaborative of MEC and cloud computing, Ning et al. [24] proposed an iterative heuristic resource allocation algorithm for dynamic offloading decisions. To minimize the total completing time of all mobile terminal tasks, Wu et al. [25] designed a computing offloading scheme based on nonorthogonal multiple access (NOMA) technology. Zhang et al. [26] integrated computing offloading, content caching, and resource allocation into one model and designed an asymmetric search tree to minimize the total delay consumption of computing tasks.

Under the constraint on computational delay, there are some researches on the problem of minimizing the total

mobile energy consumption. Chen et al. [27] designed a new communication and computing resource allocation method by clarifying the inherent characteristics of AR mobile applications. Al-Shuwaili and Simeone [28] proposed a joint optimization problem to optimize the total energy consumption of the entire system under delay constraint. Combined with the multiaccess characteristics of 5G, Yang et al. [29] considered the small-cell network architecture for task offloading and modeled the energy consumption of offloading from two aspects of task computing and communication. Wang et al. [30] designed an innovative framework to improve the performance of MEC, based on this framework, an optimal resource allocation scheme has been proposed to optimize total energy consumption of wireless access points.

Computation offloading based on delay and energy consumption is another important research problem [31–33]. To meet the task processing delay and energy consumption constraints on mobile devices, Mashhadi et al. [31] proposed an auction in which edge servers were assigned to mobile devices executed by a pair of neural networks. In [32], to minimize the total overhead of MEC system, an improved genetic algorithm was used to solve the joint optimization problem of computing offload decision and channel resource allocation. Guo et al. [33] solved the problem of MEC offloading in ultradense networks and designed a two-layer game greedy offloading scheme to minimize the total computational overhead of processing time and energy consumption.

It is important to balance the system load to improve system performance. Fakhri et al. [34] proposed a discrete particle swarm optimization algorithm to solve the load balancing optimization problem. In order to balance the load of virtual machines, Tong et al. [35] proposed a new dynamic load balancing task scheduling algorithm based on reinforcement learning and service protocol. In [36], a load balancing algorithm based on autonomous agent has been designed, which presaves the information of candidate virtual machines to improve dynamic load balancing and reduce service time for the cloud environment.

Privacy-preserving is an important issue to be considered for wireless transmission. Aiming at the privacy problem when processing max/min queries in two-layer sensor networks, Yao et al. [37] proposed a privacy protection scheme for max/min queries. The scheme adopts the prefix member authentication method to ensure the privacy of sensitive data stored in nodes. To reduce the risk of user privacy material exposure, Wan et al. [38] proposed an optimized cloud computing security deployment structure and a security mechanism for material protection. In [39], an encrypted data processing and retrieval security solutions were designed to resolve the data security problem in cloud computing.

However, the aforementioned privacy protection and system load issues were designed for cloud computing or mobile cloud computing environments. This limitation has prompted our research to solve the problem of efficient offloading based on privacy protection. In this paper, we consider the privacy protection of data transmission and system load in edge computing environment, which can realize the confidentiality of user privacy, effectively reduce task completion time, and UE energy consumption.

Table 1: Key symbol definition.

Notation	Description
N	Set of N users
M	Set of M MEC servers
Q_m	Set of virtual machines in server m
T_n	Computation task of user n
β_n	Workload for computation tasks T_n
α_n	Data size for computation tasks T_n
$f_{s,m}$	Computing power of virtual machine in server m
$f_{l,n}$	Local computing capability of user n
p_n^{up}	Transmission power of user n
$p_{l,n}$	Local computing power of user n
$g_{n,m}$	Channel gain from user n to edge server m
B	Uplink bandwidth
$t_{l,n}$	Delay of task T_n execution locally
$t_{n,m}^{up}$	Delay for task T_n uploaded to edge server m
$t_{n,m}^{com}$	Delay of task T_n execution on edge server m
t_n^{en}	Encryption time of task T_n
$t_{n,m}^{de}$	Decryption time of task T_n at the server m
$o_{n,m}$	Task offloading strategy, $\forall n \in N, m \in M$
$r_{n,m}$	Rate of from task T_n uploaded to edge server m
rur_m	Resource utilization rate of serve m
VAR	Load balance mean variance rate
q_m	Number of virtual machines in server m

3. System Model and Problem Formation

In this section, we introduce the system model and expounds on the researched problems. Table 1 summarizes the key symbols used in this article.

3.1. System Model. As shown in Figure 1, we consider a MEC system with multicells and servers. $N = \{1, 2, 3, \cdots, N\}$ and $M = \{1, 2, 3, \cdots, M\}$ are used to represent UEs (such as iPad and smart phones) and ESs set in the system, respectively. Each UE has a computing task to complete, and each task is atomic and indivisible. And the capacity of each ES is equal to the number of virtual machines in the ES, and $Q_m = \{1, 2, 3, \cdots, q_m\}$ represents a collection of virtual machines in ES m. Each virtual machine performs only one task at a time, and the computing capability of virtual machines on the same server is the same. The system model is established from four aspects of local computation, MEC offloading computation, system load, and security transmission mode.

3.1.1. The Model of Local Computation. The two-tuple $\{\alpha_n, \beta_n\}$ is used to represent the task T_n of the user n, where

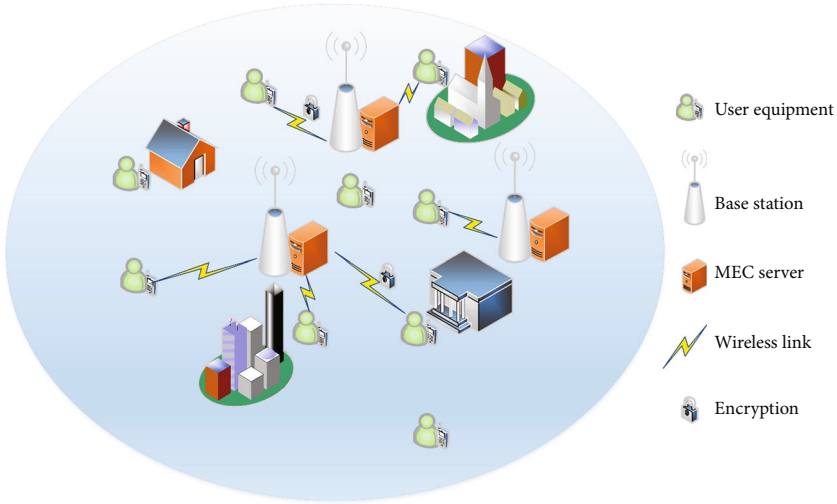


Figure 1: A cellular system with multiple MEC servers.

α_n (bits) represents the data size required to complete task T_n (including input parameters and program codes), and β_n represents the number of cycles required to complete task T_n. The values of α_n and β_n can be obtained by the program analyzer. The binary variable $o_{n,m} \in \{0, 1\}, \forall n \in N, m \in M$, is defined to represent the offloading decision of the task. $o_{n,m} = 1$ indicates that the task T_n is offloaded to the edge server m for execution; otherwise, the task T_n is executed locally. Each computing task can be offloaded to ES execution or executed locally. Therefore, a reasonable offloading strategy needs to meet the limitation:

$$\sum_{m=1}^{M} o_{n,m} \leq 1, \forall n \in N. \tag{1}$$

If $o_{n,m} = 0$, we perform task T_n locally. $f_{l,n}$ represents the local computing capability of user n. The total time to perform task T_n can be calculated as

$$t_{l,n} = \frac{\beta_n}{f_{l,n}}. \tag{2}$$

In order to calculate the energy consumption when the task T_n is executed locally, the energy consumption model $\varsigma(f_{l,n})^2$ in [40] is used, which represents the energy consumption of a calculation cycle, where ς is energy coefficient that depends on the chip structure. Therefore, the energy consumption when the task is executed locally can be calculated as

$$e_{l,n} = \beta_n \varsigma (f_{l,n})^2. \tag{3}$$

3.1.2. The Model of MEC Offloading Computation. When $o_{n,m} = 1$, the task $T_n, \forall n \in N$ is offloaded to the ES $m, \forall m \in M$ to execute. The task T_n offloading calculation includes three steps: (1) task T_n is uploaded to ES m, (2) ES m executes task T_n, and (3) the calculation results is returned to user n. We consider that the downlink transmission rate is much larger than the uplink transmission rate [41], and the amount of data for the calculation result is much smaller than that of the input task, so we ignore the delay of transmitting the calculation results from ES to UE n. Next, the two steps of task upload and task execution will be introduced in detail.

(1) Task Upload. In this paper, we use OFDMA as a multiaccess scheme for uplink. When a subband is occupied by multiple users, it will cause additional interference. Therefore, the signal-to-noise ratio (SNR) from UE n to ES m can be calculated as

$$\chi_{n,m} = \frac{p_n^{up} g_{n,m}}{\sum_{i=1,i\neq n}^{N} \sum_{j=1,j\neq m}^{M} o_{i,j} p_i^{up} g_{i,m} + \sigma^2}, \tag{4}$$

where $g_{n,m}$ represents the uplink channel gain between UE n and ES m, p_n^{up} represents the upload power of user n. The first term of the denominator represents the interference generated by other users on the same subband. The second term σ^2 of the denominator represents the background noise power. Therefore, the upload rate of user n to server m can be calculated as

$$r_{n,m} = B \log_2 (1 + \chi_{n,m}), \forall n \in N, m \in M, \tag{5}$$

where B represents the bandwidth of the uplink. According to (5), the time to upload to ES m can be calculated as follows

$$t_{n,m}^{up} = \frac{\alpha_n}{r_{n,m}}. \tag{6}$$

The energy consumption for user n to upload task T_n to ES m can be calculated as follows

$$e_{n,m}^{up} = p_n^{up} t_{n,m}^{up} = p_n^{up} \frac{\alpha_n}{r_{n,m}}. \tag{7}$$

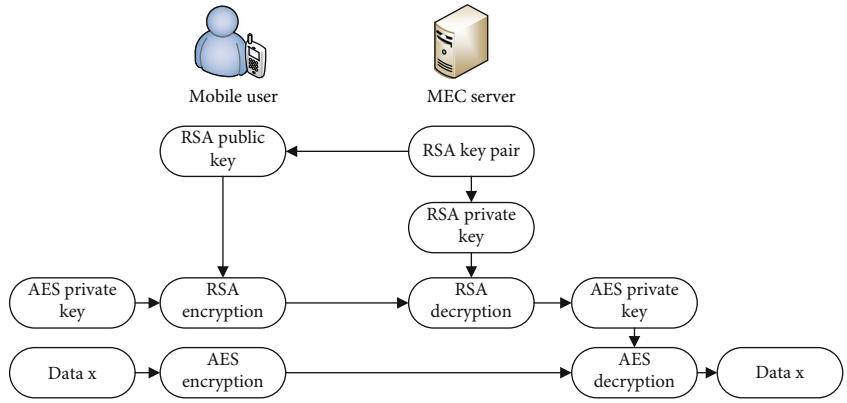


Figure 2: Flow chart of hybrid encrypted data transmission.

(2) Task Execution. The time to perform task is calculated as follows

$$t_{n,m}^{com} = \frac{\beta_n}{f_{s,m}}, \tag{8}$$

where $f_{s,m}$ represents the computing capability of virtual machine in ES m.

3.1.3. Load Model. In this paper, we propose the load mean variance to evaluate the current system load state. According to the occupancy of virtual machines in ES, the resource utilization rate of each ES is calculated as

$$rur_m = \frac{1}{q_m} \sum_{n=1}^{N} \sum_{k=1}^{q_m} o_{n,m} Q_{k,n}^m, \forall m \in M, \tag{9}$$

where $Q_{k,n}^m$ represents the occupancy of virtual machine in ES m and $Q_{k,n}^m = 1$ represents that the kth virtual machine of ES m is occupied by user n; otherwise, $Q_{k,n}^m = 0$. According to (9), the average load of all ESs is calculated as

$$AVG_{rur} = \frac{1}{M} \sum_{m=1}^{M} rur_m. \tag{10}$$

Thus, the load mean variance of the system is calculated as

$$VAR = \sqrt{\frac{1}{M} \sum_{j=1}^{M} \left(rur_j - AVG_{rur}\right)^2}. \tag{11}$$

3.1.4. The Security Transmission Mode. When UE offloads the task to be executed, it is easy to cause privacy leakage in the transmission process. In order to prevent the privacy leakage of the transmitted data, as shown in Figure 2, a hybrid encryption technology based on AES and RSA is used to protect the transmitted data. AES needs to transmit the key from the user to the server, if the key is not encrypted, it will lead to the problem of key leakage. Therefore, RSA is used to encrypt AES key to improve the security of encryption. Since the encryption speed of AES is faster than RSA, we use RSA to encrypt the key with a small amount of data, and AES encrypts the offloading data with a large amount of data, thereby improving the encryption speed.

Each offloading user determines whether to encrypt the transmitted data according to their own security requirements. We use $c_n, n \in N$ to represents UE n's security decision; if $c_n = 1$, it means that UE n encrypts the transmitted data; otherwise, $c_n = 0$. According to [42], the time for encrypting data is calculated as

$$t_n^{en} = c_n \left(\frac{k \times \alpha_n}{f_{l,n} \times v(aes)} \right), \tag{12}$$

where $v(aes)$ is the speed of AES encryption and $f_{l,n}$ is the computing capability of user n. The time to decrypt the offloaded data of MEC server is calculated as

$$t_{n,m}^{de} = c_n \left(\frac{f_{l,n} \times t_n^{en}}{f_{s,m}} \right), \forall n \in N, m \in M. \tag{13}$$

Then the total time of encryption and decryption can be expressed as

$$t_{n,m}^{enc} = t_n^{en} + t_{n,m}^{de}, \forall n \in N, \forall m \in M. \tag{14}$$

The energy consumed by UE to encrypt the offloaded data can be calculated as

$$e_n^{enc} = t_n^{en} \times p_{l,n} = c_n \left(\frac{k \times \alpha_n}{f_{l,n} \times v(aes)} \right) \times p_{l,n}. \tag{15}$$

3.2. Problem Formation. In this section, we will formulate the problem of computing offloading and task completion delay and maximize system performance.

According to (2), (6), (8), and (14), the total delay required to complete all tasks can be calculated as

$$f_T(o_1, o_2, \cdots, o_N) = \sum_{n=1}^{N} \left[\left(1 - \sum_{m=1}^{M} o_{n,m} \right) t_{l,n} + \sum_{m=1}^{M} o_{n,m} \left(t_{n,m}^{up} + t_{n,m}^{com} + t_{n,m}^{enc} \right) \right]. \tag{16}$$

According to (3), (7), and (15), the total energy consumption of UEs to complete all tasks can be calculated as

$$f_E(o_1, o_2, \cdots, o_N) = \sum_{n=1}^{N} \left[\left(1 - \sum_{m=1}^{M} o_{n,m} \right) e_{l,n} + \sum_{m=1}^{M} o_{n,m} \left(e_{n,m}^{up} + e_{n,m}^{enc} \right) \right]. \tag{17}$$

According to (11), the load mean variance of the system can be calculated as

$$f_V(o_1, o_2, \cdots, o_N) = \sqrt{\frac{1}{M} \sum_{m=1}^{M} (rur_m - AVG_{rur})^2}. \tag{18}$$

Therefore, the efficient computing offload problem based on privacy protection is described as a multiobjective optimization problem.

$$\min_{O} [f_T(O), f_E(O), f_V(O)], \tag{19}$$

$$s.t. o_{n,m} \in \{0, 1\}, \forall n \in N, m \in M, \tag{20}$$

$$\sum_{m=1}^{M} o_{n,m} \leq 1, \forall n \in N, \tag{21}$$

$$\sum_{n=1}^{N} o_{n,m} \leq q_j, \forall m \in M. \tag{22}$$

The constraints on the above problem can be interpreted as follows: constraint (20) implies that each computing task can be offloaded to ES or local to execute; constraint (21) states that each task only can be offloaded to one ES; and constraint (22) that the number of tasks performed on each ES cannot exceed the total number of virtual machines on the ES.

4. Our Proposed Efficient Computing Offloading Scheme Based on Privacy-Preserving

In this section, the improved NSGA-II algorithm (INSGA-II) is used to solve the efficient computing offloading problem based on privacy protection, and the optimal offloading strategy set is obtained. Finally, the optimal offloading strategy is obtained by min-max normalization and weighted accumulation.

4.1. Optimize the Efficient Computing Offloading Model Based on Privacy Protection by INSGA-II. In this section, we mainly solve the multiobjective optimization problem (19). We can solve the problem by transforming the multiobjective optimization problem into a single objective problem and set weights for different objectives according to user needs. However, when the user's demands changes, we need to reset the weights and rerun the algorithm. Therefore, we can use the multiobjective optimization algorithm NSGA-II to solve the problem (19). Even if the user's demands changes, there is no need to rerun the algorithm. First, we encode the strategy of task offloading and give the fitness function. Then, we propose an improved NSGA-II algorithm to solve the problem (19). As shown in Figure 3, the basic idea of NSGA-II can be described as follows:

(1) First, initialize a population of size P. And the first-generation population is obtained by selection, cross-over, and mutation of the initial population

(2) Then, from the second generation, $2P$ individuals are obtained by combining the parent population with the offspring population. P individuals are selected from the combined population to form a new parent population by crowding degree calculation and fast nondominated sorting

(3) Finally, a new offspring population is generated through selection, cross-over, and mutation of genetic algorithm

(4) Iteration will stop until the maximum number of iterations are reached, so as to obtain the optimal population

Next, the preparation work and implementation steps of INSGA-II are introduced in detail.

4.1.1. Encoding. Encoding is the first problem solved by NSGA-II algorithm. To solve problem (19), we transform the solution into chromosome embodied in the code. As shown in Figure 4, a solution is designed as a two-tuple, which includes execution Location = $\{1, 0, 1, 1, 0\}$ and Server = $\{4, 0, 7, 2, 0\}$, where the task is offloaded. For Location, if task is executed in ES, the value is 1; otherwise, the value is 0. For Server, its value represents the server number to which task is offloaded, and the value is 0 if task is executed locally.

4.1.2. Fitness Function. In the process of finding the best individual, the fitness function is used to evaluate the quality of the individual. We use Equations (16)–(18) as fitness functions to express task completion time, total energy consumption, and load mean variance, respectively. Our goal is to find an offloading strategy to make the values of the three fitness functions relatively good.

4.1.3. Initialize the Population. Under the constraint of decision space, the initial population with size P is randomly generated. Based on the ergodic characteristics of chaotic sequences [43], we introduce the chaotic sequences to initialize the population to improve the global optimization

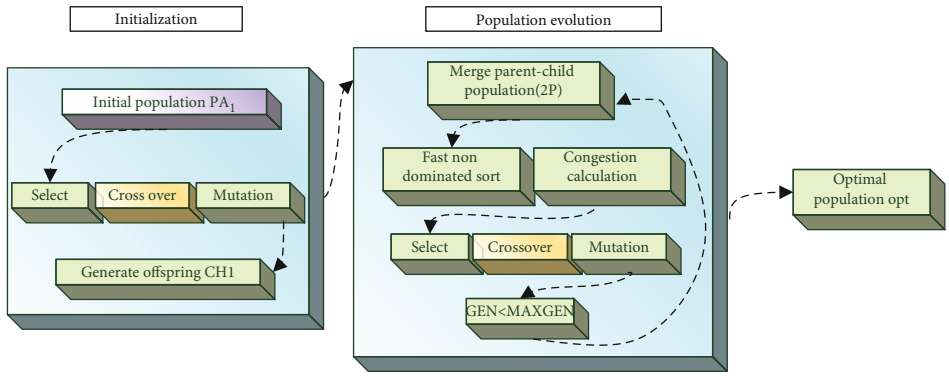


Figure 3: The INSGA-II algorithm flow chart (Algorithm 1 improves the initialization population operation. Algorithm 2 improves the cross-over operation).

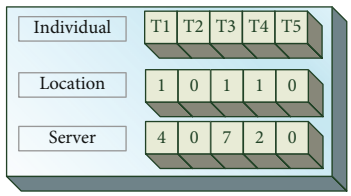


Figure 4: Encoding scheme.

ability and avoid the search process falling into local optimization. Algorithm 1 gives the specific steps of population initialization. Iterate for P individuals in the population (lines 1-2). Firstly, for each individual, the logistic chaotic map $x(n+1) = 4x(n)(1-x(n))$ is iterated N times to generate chaotic sequence Y_i (lines 3-6). Then, the initial value of the current individual i is obtained according to Y_i (lines 7-14). Incorporate the initialized individuals into the initial population set (lines 15-17). Finally, the above process is iterated P times to generate an initial population PA_0 with a size of P individuals.

4.1.4. Fast Nondominated Sorting and Crowding Calculation. In order to retain the best offloading strategy, we merge the offspring and the parent with population size of P and select the best P individuals as the new parent population by fast nondominated sorting and crowding calculation. The specific steps are described as follows: (1) Firstly, a new population PC_i with population size of $2P$ is obtained by combining parent PA_i with offspring CH_i, where $i \in (0, Gen)$, and Gen is the total number of evolutions. (2) According to the fitness functions (16)–(18), the individuals in population PC_i is arranged in fast nondominated sorting. (3) To ensure the diversity of individuals, we calculate the crowding degree of individuals in same dominant layer according to formula (23), where f_j is the j-th fitness function and i_d is crowding degree. (4) P individuals are selected to form a new parent population PA_{i+1} by crowding degree calculation and fast nondominated sorting.

$$i_d = \sum_{j=1}^{F} \left(f_j(i+1) - f_j(i-1) \right) / \left(f_j^{\max} - f_j^{\min} \right). \quad (23)$$

4.1.5. Selection. The tournament selection algorithm is used for selection operation. Firstly, k $(k < p)$ individuals are randomly selected from the P individuals in parent population. Then, the individuals with best fitness value are selected to enter next generation population. The above process is repeated until new P individuals are obtained.

4.1.6. Cross-Over and Mutation. Cross-over operation refers to the operation of replacing and reorganizing some structures of two parent individuals according to the cross-over probability to generate new individuals. Cross-over operation is the main operator to generate new individuals. As an auxiliary operator, mutation operation is to generate new patterns. Assuming that there is only cross-operation, the new solution generated in the iterative process can always only be the combination of existing patterns in the initial population. If the key modes of constructing the optimal solution are missing in the initial population, the optimal solution cannot be obtained only through cross-operation, and we also need to use the local random search ability of mutation operator to accelerate the convergence to the optimal solution. Therefore, both cross-over and mutation operations are indispensable. Next, we will describe these two operations, respectively.

(1) Cross-Over. Cross-over can retain the excellent genes left by each evolution. However, if the two crossed individuals are very similar, it will be difficult to produce new individuals, thus reducing the diversity of the population.

In order to solve this problem, the individual similarity judgment is introduced. In Algorithm 2, we give the specific process of cross-over operation based on similarity judgment. First, traverse P individuals in the population (lines 1). Generate a random number P_{cri} (lines 2) for individual i in the current iteration. If P_{cri} is less than the cross-over probability P_{cr}, add the current individual to the cross-over individual set (lines 3-7). Then, traverse the cross individual set (lines 8) and calculate the similarity between the two crossed individuals according to Equation (24) (lines 9). If the similarity is less than the similarity threshold P_θ, perform the cross-over operation (lines 10-13). The new population is obtained by the above cross-over operation. An

Input: Initial values of logistic chaotic map y_1; Number of iterations N; Population size P.
Output: first-generation population PA_0.
1: **for** $i = 1$ to P **do**
2: $Y_i \longleftarrow y_1 \cup Y_i$
3: **for** $j = 2$ to N **do**
4: $y_j = 4 \times y_{j-1} \times (1 - y_{j-1})$
5: $Y_i \longleftarrow y_j \cup Y_i$
6: **end for**
7: **for** $j = 1$ to N **do**
8: **if** $Y_i(j) \geq 0.5$ **then**
9: T_j is offloaded to MEC server, $o_j = 1$
10: **else**
11: T_j is executed locally, $o_j = 0$
12: **end if**
13: $O_i \longleftarrow o_j \cup O_i$
14: **end for**
15: $y_1 = Y_i(N)$
16: $PA_0 \longleftarrow O_i \cup PA_0$
17: **end for**

Algorithm 1: Population initialization.

Input: cross-over probability $p_{cr} = 0.8$; similarity threshold p_θ.
Output: Individuals after crossing *newCR*.
1: **for** $i = 1$ to P **do**
2: P_{cri}= Math. random
3: **if** $P_{cri} < P_{cr}$ **then**
4: $CR = CR \cup O_i$
5: $count++$
6: **end if**
7: **end for**
8: **for** $j = 1; j \leq count; j += 2$ **do**
9: Calculate $Sim(CR_j, CR_{j+1})$using (24)
10: **if** $Sim(CR_j, CR_{j+1}) < P_\theta$ **then**
11: $newCR$ =cross-over (CR_j, CR_{j+1})
12: **end if**
13: **end for**

Algorithm 2: Cross-overs based on similarity.

example of cross-over operation is given in Figure 5, which performs a single-point cross-over on two individuals.

$$Sim\,(Y_i, Z_i) = \frac{N\text{-}\sum_{i=1}^{N}(Y_i \oplus Z_i)}{N}. \tag{24}$$

4.1.7. Mutation. Mutation breaks through the limitations of the current search and is more conducive to the algorithm to find global optimal solution. Individuals whose mutation probability is less than $p_{mu} = 0.1$ will randomly select a gene for mutation operation. An example of a mutation operation is shown in Figure 6.

We get the offspring CH_g of evolution. Combine offspring CH_g with parent PA_g to form a new parent PA_{g+1}, and continue the evolution of next generation until the maximum evolutionary generation is reached. Solutions with good fitness will spread in the solution set, and solutions with poor performance will be slowly eliminated. Finally, an optimal set OPT of offloading strategies is obtained.

4.2. Get the Optimal Offloading Strategy. In this section, we select an optimal individual from the solution set OPT. Min-max normalization is used to normalize the fitness values to ensure the reliability of results. The total time delay $T = t_l + t_s + t_{enc}$ of individual opt_i to complete all tasks is normalized as

$$T'_{opti} = \frac{T_{opti} - T_{\min}}{T_{\max} - T_{\min}}, \tag{25}$$

where $T_{\max}$, $T_{\min}$, and T_{opti} represent the maximum task completion time, the minimum task completion time, and

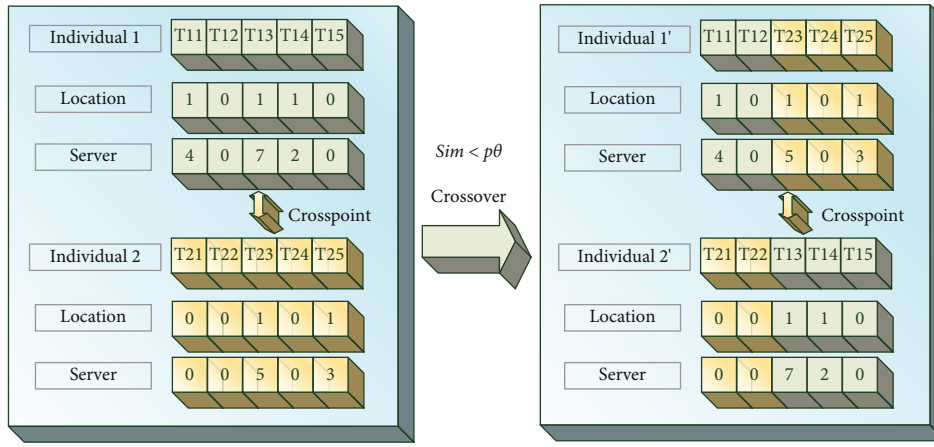


Figure 5: Cross-over operation.

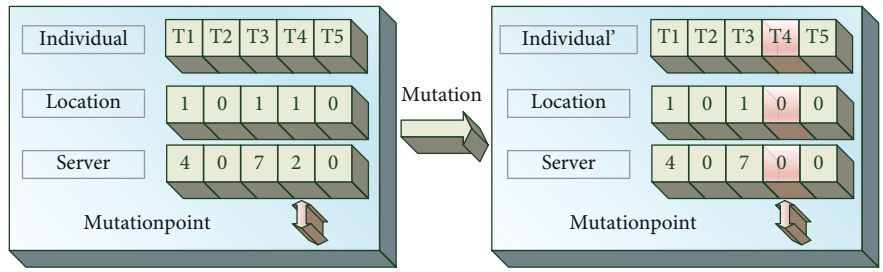


Figure 6: Mutation operation.

the delay for i-th individual to complete task, respectively. The total energy consumption $E = e_l + e_s + e_{enc}$ of UEs is normalized as

$$E'_{opti} = \frac{E_{opti} - E_{\min}}{E_{\max} - E_{\min}}, \tag{26}$$

where $E_{\max}$, $E_{\min}$, and E_{opti} represent the maximum energy consumption of UEs, the minimum energy consumption of UEs, and the UE energy consumption of i-th individual, respectively. Finally, the load mean variance of system is normalized as

$$VAR'_{opti} = \frac{VAR_{opti} - VAR_{\min}}{VAR_{\max} - VAR_{\min}}, \tag{27}$$

where $VAR_{\max}$, $VAR_{\min}$, and VAR_{opti} represent the maximum load mean variance, the minimum load mean variance, and the load mean variance of i-th individual, respectively.

Next, we use simple additive weighting for normalizing fitness values to measure the quality of individuals in population OPT.

$$F(opt_i) = \phi_1 T'_{opti} + \phi_2 E'_{opti} + \phi_3 VAR'_{opti}. \tag{28}$$

where ϕ_1, ϕ_2, and ϕ_3 represent the weight of delay, energy consumption, and load mean variance, respectively. And ϕ_1, ϕ_2, and ϕ_3 satisfies $\phi_1 + \phi_2 + \phi_3 = 1$.

In Algorithm 3, the main process of obtaining offloading strategy based on INSGA-II is introduced. Firstly, the initial population PA1 is generated by Algorithm 1 (line 1). Then, the optimal offloading policy set OPT is obtained by the INSGA-II algorithm (lines 2-10). Finally, the optimal offloading strategy is obtained by the min-max normalization and weighted accumulation (lines 11-18).

5. Performance Analysis

In this section, we evaluate the performance of our proposed ECOAP algorithm through simulation results. The simulation is performed on MATLAB based on simulator 2018. We consider a multiuser multicell scenario, and each cell has a base station. We assume that a single antenna is used for communication between the user and the base station. The parameters used in the simulation are given in Table 2.

To evaluate the performance of our proposed algorithm, we compare it with the following five basic offloading methods.

(i) Offloading based on NSGA-II (NSGA-II): based on the current environment, the NSGA-II algorithm is used to obtain the offloading strategy

(ii) Unsecured offloading (UO): regardless of the privacy-preserving of offloading data, the offloading decision is made in the current setting environment

Input: Evolutionary algebra gen; population size P; cross-over and mutation probability.$p_{cr} = 0.8$, $p_{mu} = 0.1$.
Output: Optimal offloading strategy opt.
1: initial population PA_1 by Algorithm 1
2: g =1
3: **while** $g < gen$ **do**
4: $PC_g = CH_g + PA_g$
5: PA_{g+1} = fast nondominated sort (PC_g) and crowding calculation(PC_g)
6: select(PA_{g+1}) by tournament selection strategy
7: CH_{g+1}= cross-over (PA_{g+1}) by Algorithm 2 and mutation (PA_{g+1}).
8: $g = g + 1$
9: **end while**
10: get O_{opt} by INSGA-II
11: calculate T_{opt}, E_{opt}, VAR_{opt} by $g_T(opt)$, $g_E(opt)$, $f_V(opt)$
12: get $T_{\max}$, $E_{\max}$, $VAR_{\max}$ from T_{opt}, E_{opt}, VAR_{opt}
13: get $T_{\min}$, $E_{\min}$, $VAR_{\min}$ from T_{opt}, E_{opt}, VAR_{opt}
14: **for** $j = 1 : P$ **do**
15: calculate T_{opt}, E_{opt}, VAR_{opt} by min-max standardization
16: calculate $F(opt_j) = \phi_1 T'_{optj} + \phi_2 E'_{optj} + \phi_3 VAR'_{optj}$
17: **end for**
18: $opt = F_{\min}(OPT)$

Algorithm 3: Obtain offloading strategy based on INSGA-II.

Table 2: Summary of key notations.

Parameters description	Value
Number of mobile users N	[5,60]
Number of servers M	7
System total bandwidth	$10MHZ$
Size of computation task T_n	$[300, 600]kb$
Uplink channel gains g	$127 + 30 \log_{10} d_{[km]}$
CPU frequency of servers $f_{s,m}$	$[10, 15]GHZ$
CPU frequency of users $f_{l,n}$	$[0.4, 1]\ GHZ$
Transmission power of users p_n^{up}	$[0.4, 1]W$

(iii) Offloading without considering system load (OWSL): the load problem of ES is not considered, and offloading decision is made based on the current environment

(iv) Local execution (LE): all UE's tasks are executed locally without offloading

(v) All offloading (AO): all UE's tasks are offloaded to ES for execution

(vi) Offloading based on genetic algorithm (GA): based on the current environment, improved GA algorithm is used to solve the optimization problem of offloading decision [35]

5.1. Comparison between the INSGA-II Algorithm and NSGA-II Algorithm. Figures 7–9 show the comparison of average delay, average energy consumption, and average load mean variance between INSGA-II and NSGA-II algorithms. Figure 7 shows that the average delay of the INSGA-II algorithm decreases with the increase in iterations and tends to stabilize when the iteration reaches the 15th generation. However, the NSGA-II algorithm tends to stable until 45 generations, and the average delay of NSGA-II algorithm is higher than that of the INSGA-II algorithm. Similarly, the average energy consumption of the INSGA-II algorithm is lower than that of the NSGA-II algorithm in Figure 8. Figure 9 shows the comparison of the average load mean variance between the two algorithms. When the INSGA-II algorithm and NSGA-II algorithm are iterated to 45 generations, the load mean variance of the INSGA-II

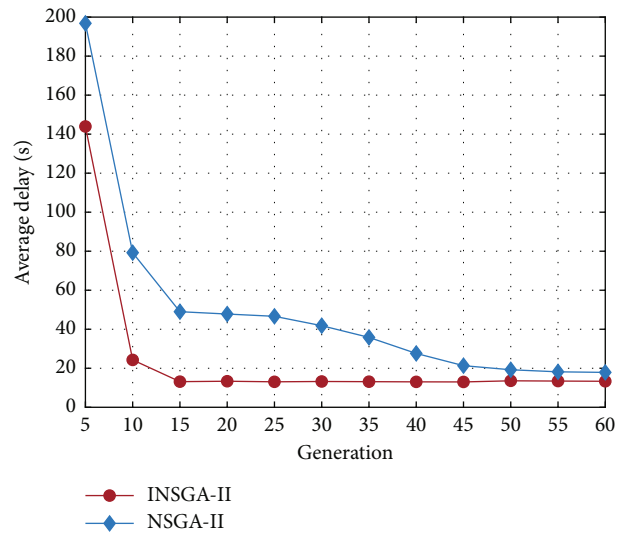


Figure 7: Influence of different evolutionary generations on average delay.

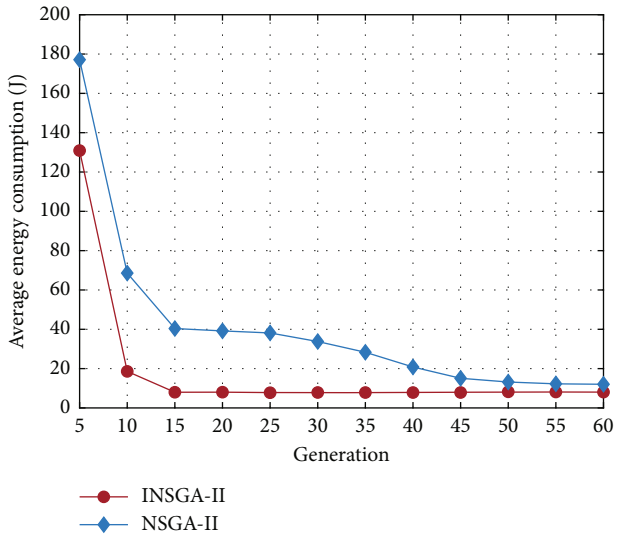


Figure 8: Influence of different evolutionary generations on average energy consumption.

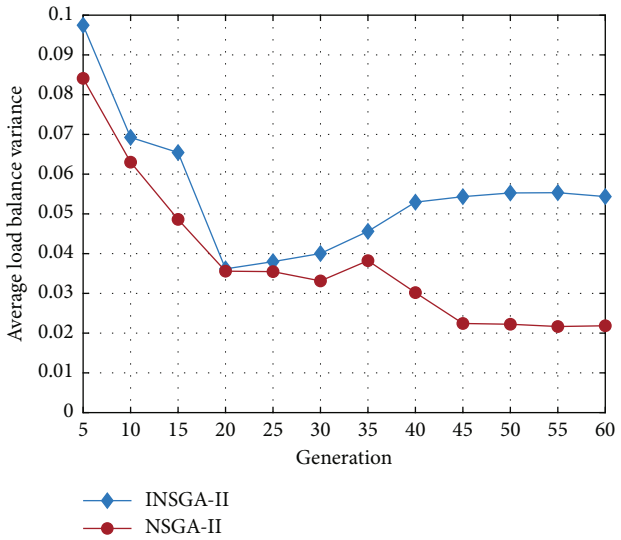


Figure 9: Influence of different evolutionary generations on average load balance variance.

algorithm is better than that of the NSGA-II algorithm. Compared with the NSGA-II algorithm, the INSGA-II algorithm has better global optimization ability, reduces the number of iterations, and requires less time delay and energy consumption.

5.2. Comparison of UEs' Energy Consumption. As shown in Figure 10, the influence of different number of UEs on average energy consumption is described. We compared energy consumption in five different scenarios. It can be seen that with the increase in the number of UEs, the average energy consumption of all schemes are growing. Because ECOAP optimizes the energy consumptions of users, the average energy consumption of ECOAP is relatively small. When the number of users is less than 20, the energy consumptions

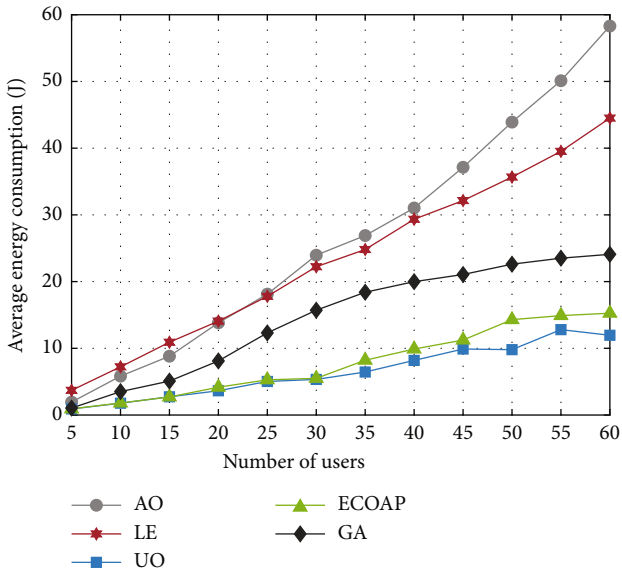


Figure 10: Influence of different numbers of UEs on average energy consumption.

of UO, ECOAP, GA, and AO are less than that of LE, and the energy consumption of UO and ECOAP is basically the same. However, the growth of users leads to additional energy consumption of data transmission, and the energy consumption of AO exceeds the energy consumption of local execution (LE). At the same time, the increase in offloading users leads to the increase of energy consumption for encryption and decryption, and the energy consumption of ECOAP exceeds that of UO.

5.3. Comparison of Tasks' Completion Delay. As shown in Figure 11, the comparison of average delay against different number of UEs is described. We can see that with the growth of UEs, the delays of all schemes are increasing. Because ECOAP optimizes the delay of completing tasks, the average delay of ECOAP is relatively small. When the number of users is less than 20, the delays of UO, ECOAP, GA, and AO is less than that of LE, and the delays of UO and ECOAP are basically the same. When the number of users exceeds 20, the offloading users will compete for limited wireless resources, which results in the delay of AO exceeding the latency of LE. In addition, with the growth of users, the possibility of encrypting offloading data becomes greater, so that the delay of ECOAP exceeds that of UO, but ECOAP increases the confidentiality of offloaded data.

5.4. Comparison of Load Mean Variance. As shown in Figure 12, the comparison of load mean variance against different number of UEs is described. We can see that ECOAP always performs best and load mean variance is the lowest. By optimizing the mean variance of system load, ECOAP can improve the system load to a certain extent and has a good performance in load balancing. However, OWSL does not consider the system load, so some ES may be overloaded or lightly loaded with the growth of users, which will reduce the performance of system.

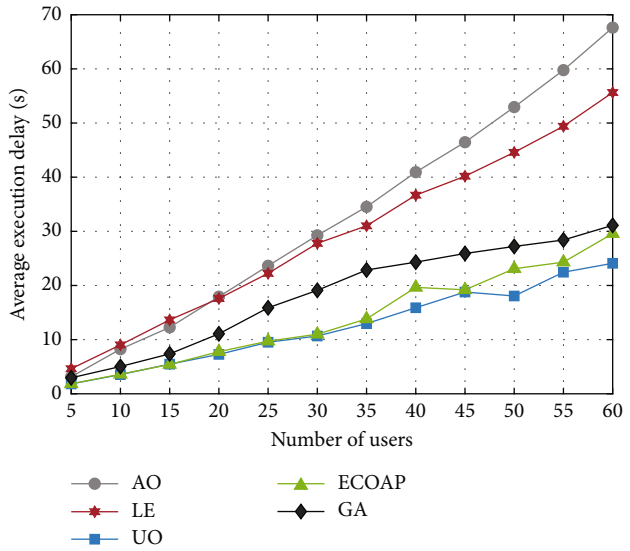


Figure 11: Influence of different number of UEs on average delay.

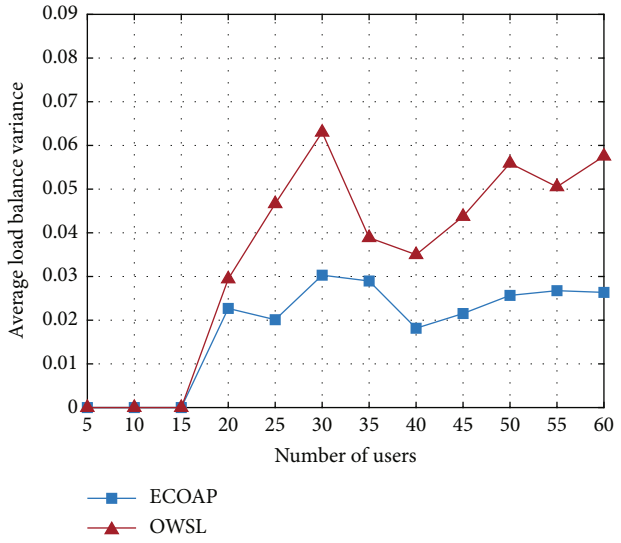


Figure 12: Influence of different numbers of UEs on average load mean variance.

5.5. Influence of UEs' Preferences. Figures 13–15 show the changes of delay, energy consumption, and load mean variance when user preferences vary from 0.1 to 0.9, where $w1$, $w2$, and $w3$ represent user preferences for delay, energy consumption, and system load, respectively. It can be seen from Figure 13 that under the premise that other parameters remain constant, the average delay of task completion decreases with the increase of $w1$. Similarly, we can observe from Figure 14 that the average energy consumption of UEs is decreasing as the weight $w2$ increases. Figure 15 shows that when the user's preference $w3$ for the load average variance increases, the load mean variance is reduced. In addition, the average delay to complete tasks and the average energy consumption of users will increase with the growth of users. This is because the growth of users leads to competition for limited resources, which leads to the increase in time and energy consumption.

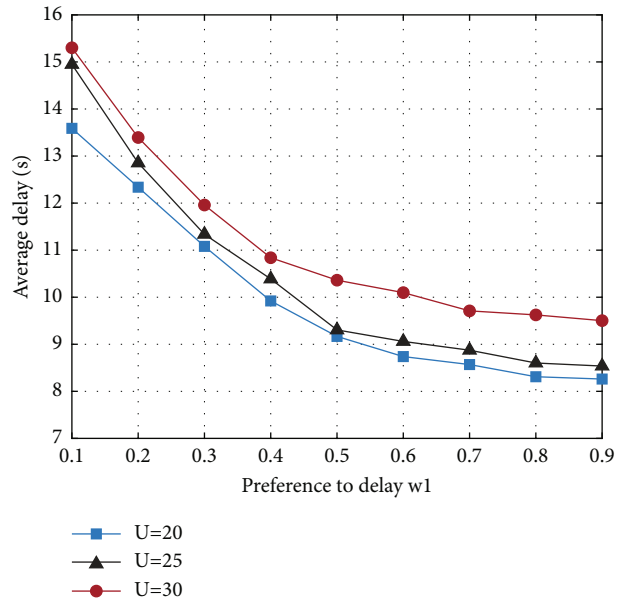


Figure 13: The influence of time preference on delay.

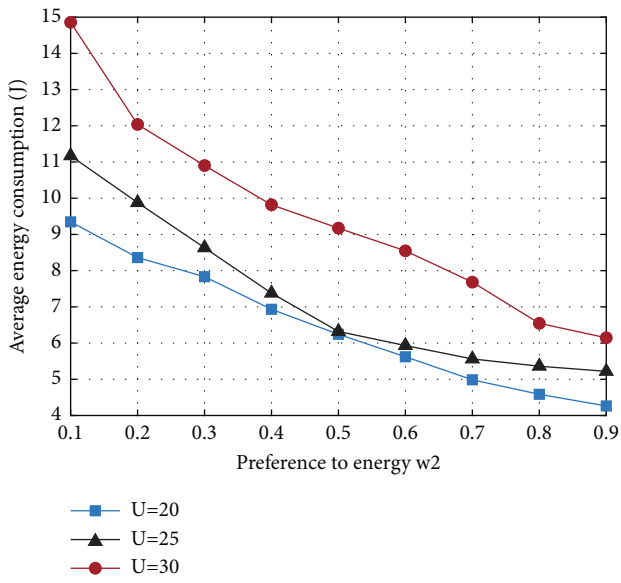


Figure 14: The influence of energy preference on energy consumption.

5.6. Engineering Applications. With the continuous evolution of modern industry towards intelligent direction, the number of industrial field equipment is increasing rapidly, and the demand for computing resources is increasing. As shown in Figure 16, edge computing is widely used in industrial Internet of things (IIoT) environment to provide localized computing resources with low latency and high reliability for factory equipment. Edge computing

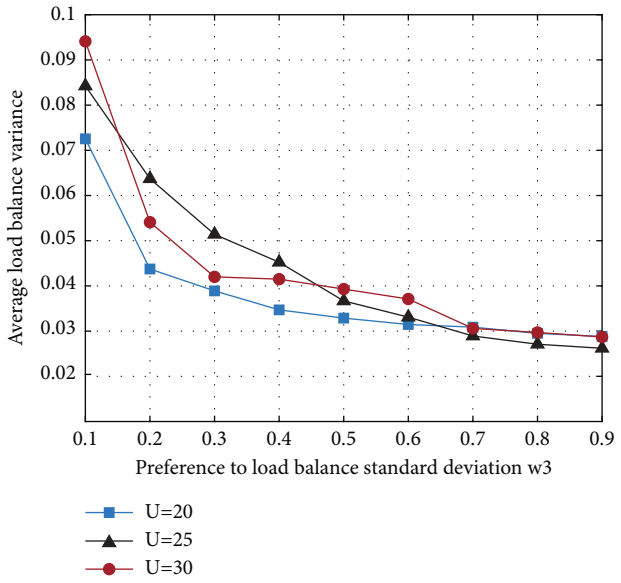


Figure 15: The influence of load preference on load mean variance.

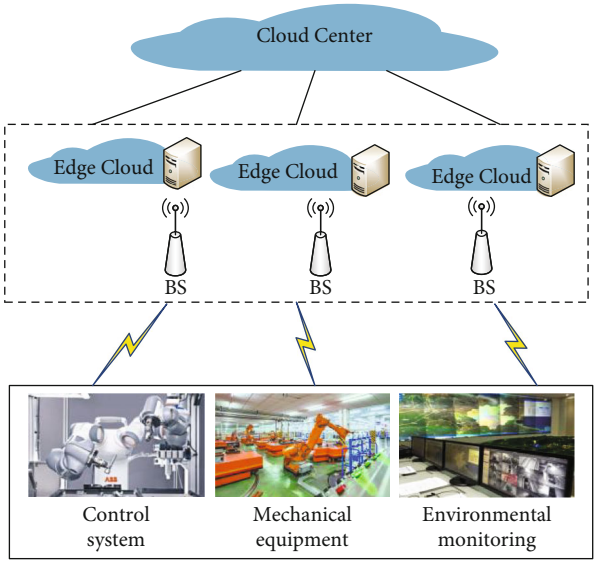


Figure 16: Industrial Internet of things (IIoT) based on edge computing.

distributes computing nodes in the factory production environment, bringing the computing resources closer to factory equipment.

However, in some industrial processes, the requirements for computing delay, energy consumption, and data privacy are particularly high. For example, IIoT may face problems of privacy leakage and risk warning needs real-time response. Therefore, as shown in Figure 17, we use the hybrid encryption method to encrypt the data offloaded from factory equipment to ensure the privacy of the offloaded data. Then, the INSGA-II algorithm is proposed to efficiently offload and obtain the optimal policy set. Finally, the Pareto optimal offloading strategy is selected, so as to

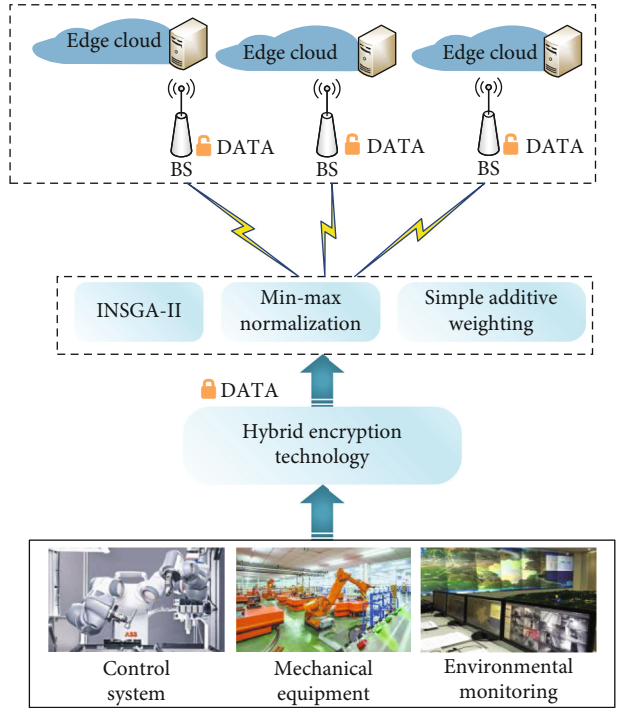


Figure 17: Application diagram of ECOAP algorithm in IIoT.

improve the localized computing services with low delay, low energy consumption, and high reliability for the field equipment of factory.

6. Conclusions and Future Work

In this paper, we propose an efficient computing offloading algorithm based on privacy protection for investigating the privacy protection and task offloading in the multicell MEC network. A hybrid encryption technology is introduced to protect the privacy of offloaded users. The encryption technology combines the advantages of AES and RSA encryption technology to improve the security and speed of encryption. To reduce the UEs' energy consumption and task completion delay in the case of encryption, we propose an improved NSGA-II algorithm (INSGA-II). Simulation results show that the ECOAP algorithm can realize the confidentiality of user privacy and effectively reduce the task completion time and the UEs' energy consumption. In future work, to further reduce energy consumption and improve spectrum efficiency, we will consider using NOMA as multiaccess scheme for uplink. In addition, we will take the mobility of users into consideration to be more in line with actual scenario.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (under Grant 61672033, Grant 61873280, Grant 61873281, Grant 61972416, Grant 61672248, and Grant 61902430), in part by the National Key Research and Development (under Project 2018YFC1406204), in part by the Key Research and Development Program of Shandong Province (under Grant 2019GGX101067), in part by the Natural Science Foundation of Shandong Province (under Grant ZR2019MF012), in part by the Taishan Scholars Fund (under Grant ZX20190157), in part by the Independent Innovation Research (under Project 18CX02152A), and in part by the Fundamental Research Funds for the Central Universities (under Grant 19CX02028A). We appreciate Dr. Neal Xiong for his initial contributions to improve this paper. He helps us reorganize, rewrite, and extend this paper.

References

[1] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collaborative mobile edge computing in 5g networks: new paradigms, scenarios, and challenges," *IEEE Communications Magazine*, vol. 55, no. 4, pp. 54–61, 2017.

[2] Y. Zhu, Q. He, J. Liu, B. Li, and Y. Hu, "When crowd meets big video data: cloud-edge collaborative transcoding for personal livecast," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 1, pp. 42–53, 2020.

[3] H. Li, J. Liu, K. Wu, Z. Yang, R. W. Liu, and N. Xiong, "Spatio-temporal vessel trajectory clustering based on data mapping and density," *IEEE Access*, vol. 6, pp. 58939–58954, 2018.

[4] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: the communication perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[5] B. Lin, F. Zhu, J. Zhang et al., "A time driven data placement strategy for a scientific workflow combining edge computing and cloud computing," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4254–4265, 2019.

[6] J. Yin, W. Lo, S. Deng, Y. Li, Z. Wu, and N. Xiong, "Colbar: a collaborative location-based regularization framework for QoS prediction," *Information Sciences*, vol. 265, pp. 68–84, 2014.

[7] Y. Qu and N. Xiong, "RFH: A resilient, fault-tolerant and high-efficient replication algorithm for distributed cloud storage," in *2012 41st International Conference on Parallel Processing*, pp. 520–529, Pittsburgh, PA, USA, 2012.

[8] Z. Kuang, Z. Ma, Z. Li, and X. Deng, "Cooperative computation offloading and resource allocation for delay minimization in mobile edge computing," *Journal of Systems Architecture*, vol. 118, pp. 1–9, 2021.

[9] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795–2808, 2016.

[10] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. S. Quek, "Offloading in mobile edge computing: task allocation and computational frequency scaling," *IEEE Transactions on Communications*, vol. 65, no. 8, pp. 3571–3584, 2017.

[11] L. Cui, C. Xu, S. Yang, J. Z. Huang, and N. Lu, "Joint optimization of energy consumption and latency in mobile edge computing for internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4791–4803, 2019.

[12] Z. Wei, B. Zhao, J. Su, and X. Lu, "Dynamic edge computation offloading for internet of things with energy harvesting: a learning method," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4436–4447, 2019.

[13] D. Han, W. Chen, and Y. Fang, "Joint channel and queue aware scheduling for latency sensitive mobile edge computing with power constraints," *IEEE Transactions on Wireless Communications*, vol. 19, no. 6, pp. 3938–3951, 2020.

[14] W. Fan, Y. Liu, B. Tang, F. Wu, and Z. Wang, "Computation offloading based on cooperations of mobile edge computing-enabled base stations," *IEEE Access*, vol. 6, pp. 22622–22633, 2018.

[15] Y. Lu, S. Wu, Z. Fang, N. Xiong, S. Yoon, and D. S. Park, "Exploring finger vein based personal authentication for secure IoT," *Future Generation Computer Systems*, vol. 77, pp. 149–160, 2017.

[16] H. Xiong, H. Zhang, and J. Sun, "Attribute-based privacy-preserving data sharing for dynamic groups in cloud computing," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2739–2750, 2019.

[17] H. Chen, X. Zhu, D. Qiu, L. Liu, and Z. Du, "Scheduling for workflows with security-sensitive intermediate data by selective tasks duplication in clouds," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 9, pp. 2674–2688, 2017.

[18] I. A. Elgendy, W. Zhang, Y. C. Tian, and K. Li, "Resource allocation and computation offloading with data security for mobile edge computing," *Future Generation Computer Systems*, vol. 100, pp. 531–541, 2019.

[19] Y. Wu, J. J. Shi, K. J. Ni et al., "Secrecy-based delay-aware computation offloading via mobile edge computing for internet of things," *IEEE Internet Things Journal*, vol. 6, no. 3, pp. 4201–4213, 2019.

[20] J. Wang, J. Pan, F. Esposito, P. Calyam, Z. Yang, and P. Mohapatra, "Edge cloud offloading algorithms: issues, methods, and perspectives," *ACM Computing Surveys*, vol. 52, no. 1, 2020.

[21] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: a survey," *IEEE Internet Things Journal*, vol. 5, no. 1, pp. 450–465, 2018.

[22] M. Wu, L. Tan, and N. Xiong, "A structure fidelity approach for big data collection in wireless sensor networks," *Sensors*, vol. 15, no. 1, pp. 248–273, 2015.

[23] J. Liu, Y. Mao, J. Zhang, and K. B. Letaief, "Delay-optimal computation task scheduling for mobile-edge computing systems," in *2016 IEEE International Symposium on Information Theory (ISIT)*, pp. 1451–1455, Barcelona, 2016.

[24] Z. Ning, P. Dong, X. Kong, and F. Xia, "A cooperative partial computation offloading scheme for mobile edge computing enabled internet of things," *IEEE Internet Things Journal*, vol. 6, no. 3, pp. 4804–4814, 2019.

[25] Y. Wu, L. P. Qian, K. Ni, C. Zhang, and X. Shen, "Delay-minimization nonorthogonal multiple access enabled multi-user mobile edge computation offloading," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 3, pp. 392–407, 2019.

[26] J. Zhang, J. Zhang, X. Hu et al., "Joint resource allocation for latency-sensitive services over mobile edge computing networks with caching," *IEEE Internet Things Journal*, vol. 6, no. 3, pp. 4283–4294, 2019.

[27] X. Chen, Y. Cai, L. Li, M. Zhao, B. Champagne, and L. Hanzo, "Energy-efficient resource allocation for latency-sensitive mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 2, pp. 2246–2262, 2020.

[28] A. Al-Shuwaili and O. Simeone, "Energy-efficient resource allocation for mobile edge computing-based augmented reality applications," *IEEE Wireless Communications Letters*, vol. 6, no. 3, pp. 398–401, 2017.

[29] L. Yang, H. Zhang, M. Li, J. Guo, and H. Ji, "Mobile edge computing empowered energy efficient task offloading in 5G," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 6398–6409, 2018.

[30] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 1784–1797, 2017.

[31] F. Mashhadi, S. A. S. Monroy, A. Bozorgchenani, and D. Tarchi, "Optimal auction for delay and energy constrained task offloading in mobile edge computing," *Computer Networks*, vol. 183, article 107527, 2020.

[32] C. Du, Y. Chen, Z. Li, and G. Rudolph, "Joint optimization of offloading and communication resources in mobile edge computing," in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 2729–2734, Xiamen, China, 2019.

[33] H. Guo, J. Liu, J. Zhang, W. Sun, and N. Kato, "Mobile-edge computation offloading for ultradense IoT networks," *IEEE Internet Things Journal*, vol. 5, no. 6, pp. 4977–4988, 2018.

[34] Z. H. Fakhri, M. Khan, F. Sabir, and H. S. Al-Raweshidy, "A resource allocation mechanism for cloud radio access network based on cell differentiation and integration concept," *IEEE transactions on network science and engineering*, vol. 5, no. 4, pp. 261–275, 2018.

[35] Z. Tong, X. M. Deng, H. J. Chen, and J. Mei, "DDMTS: a novel dynamic load balancing scheduling scheme under SLA constraints in cloud computing," *Journal of Parallel and Distributed Computing*, vol. 149, pp. 138–148, 2021.

[36] F. Zhao, C. Li, and C. F. Liu, "A cloud computing security solution based on fully homomorphic encryption," in *16th International Conference on Advanced Communication Technology*, pp. 485–488, Phoenix Park, PyeongChang Korea, 2014.

[37] Y. Yao, N. Xiong, J. H. Park, L. Ma, and J. Liu, "Privacy-preserving max/min query in two-tiered wireless sensor networks," *Computers & Mathematics with Applications*, vol. 65, no. 9, pp. 1318–1325, 2013.

[38] J. Wan, B. Chen, S. Wang, M. Xia, D. Li, and C. Liu, "Fog computing for energy-aware load balancing and scheduling in smart factory," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4548–4556, 2018.

[39] G. Soni and M. Kalra, "A novel approach for load balancing in cloud data center," in *2014 IEEE International Advance Computing Conference*, pp. 807–812, ITM University Gurgaon, India, 2014.

[40] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 4, pp. 974–983, 2015.

[41] X. Lyu, H. Tian, C. Sengul, and P. Zhang, "Multiuser joint task offloading and resource optimization in proximate clouds," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3435–3447, 2017.

[42] B. Huang, Z. Li, P. Tang et al., "Security modeling and efficient computation offloading for service workflow in mobile edge computing," *Journal of Ambient Intelligence and Humanized Computing*, vol. 97, pp. 755–774, 2019.

[43] R. M. Yin, J. Wang, J. Yuan, and S. X. Wang, "Weak key analysis for chaotic cipher based on randomness properties," *Science China Information Sciences*, vol. 55, no. 5, pp. 1162–1171, 2012.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4592428, 14 pages
https://doi.org/10.1155/2022/4592428

Research Article

Burner: Recipe Automatic Generation for HPC Container Based on Domain Knowledge Graph

Shuaihao Zhong[1], **Duoqiang Wang**[1], **Wei Li**[2], **Feng Lu**[1] **and Hai Jin**[1]

[1]*National Engineering Research Center for Big Data Technology and System, Services Computing Technology and System Lab, Cluster and Grid Computing Lab, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China*
[2]*Australia-China Joint Research Centre for Energy Informatics and Demand Response Technologies, Centre for Distributed and High Performance Computing, School of Computer Science, University of Sydney, Sydney, Australia*

Correspondence should be addressed to Shuaihao Zhong; shuaihaozhong@hust.edu.cn

Received 25 February 2022; Revised 4 April 2022; Accepted 20 April 2022; Published 25 May 2022

Academic Editor: Yingjie Wang

As one of the emerging cloud computing technologies, containers are widely used in academia and industry. The cloud computing built by the container in the high performance computing (HPC) center can provide high-quality services to users at the edge. Singularity Definition File and Dockerfile (we refer to such files as recipes) have attracted wide attention due to their encapsulation of the application running environment in a container. However, creating a recipe requires extensive domain knowledge, which is error-prone and time-consuming. Accordingly, more than 34% of Dockerfiles in Github cannot successfully build container images. The crucial points about recipe creation include selecting the entities (base images and packages) and determining their relationships (correct installation order for transitive dependencies). Since the relationships between entities can be expressed accurately and efficiently by the knowledge graph, we introduce knowledge graph to generate high-quality recipes automatically. This paper proposes an automatic recipe generation system named Burner, enabling users with no professional computer background to generate the recipes. We first develop a toolset including a recipe parser and an entity-relationship miner. Our two-phase recipe parsing method can perform abstract syntax tree (AST) parsing more deeply on the recipe file to achieve entity extraction; the parsing success rate (PSR) of the two-phase parsing method is 10.1% higher than the one-phase parsing. Then, we build a knowledge base containing 2,832 entities and 62,614 entity relationships, meeting the needs of typical HPC applications. In the test of image build, the singularity image build success rate reaches 80%. Compared with the ItemCF recommendation method, our recommendation method TB-TFIDF achieves a performance improvement by up to 50.86%.

1. Introduction

With the rapid development of IoT technology, the application scenarios are very wide. When the computing power of edge computing is limited in IoT applications, high-performance computing cloud can supplement the powerful computing power. Container technology is widely popular due to its lightweight and convenience. At the same time, researchers in HPC have also recognized the value of containers. Singularity [1] is currently the most widely used container technology in the HPC field, and many optimizations have been made for HPC applications. First, Singularity can prevent user privilege escalation within the container. Secondly, it can make full use of the host's high-speed interconnect hardware such as InfiniBand, simplifying access to acceleration devices such as GPUs. By now, most of the world's top HPC centers use Singularity as a solution for containerizing HPC applications in production environments.

Container technology simplifies the packaging of applications so that the dependent environment can be easily maintained. The encapsulation of the application running environment in container technology depends on recipes.

The recipe is the core of implementing application-dependent environment encapsulation which is a script written based on domain-specific language (DSL) for building container images. It records all instructions on how to build the application running environment. The use of recipes improves the transparency of the research process and facilitates the reproduction of scientific research results [2–4]. However, the effort involved in manually constructing an environment specification is non-trivial. An experienced developer may spend 20 minutes to 2 hours creating a recipe for an application and often fails to build an accurate specification [5]. Common challenges in writing recipes include selecting base images and packages and determining the correct installation order for transitive dependencies.

Henkel et al. designed the Binnacle toolset [6] to parse the 178,000 Dockerfiles present in the collected Github projects. This toolset is capable of mining semantic rules and best practices in Dockerfiles, providing friendly suggestions to Dockerfile developers. Unfortunately, the Binnacle toolset cannot be directly applied to Singularity recipe parsing, nor can it mine dependencies between packages. DockerizeMe [7] reproduces the running environment of Python code by building a Docker image and uses a combination of static analysis and dynamic analysis to solve the import error problem in Python. However, DockerizeMe mainly analyzes the Python language, which is only suitable for specific scenarios and cannot deal with the diversity of software systems.

HPC Container Maker [8] is an open source tool to make it easier to generate container specification files. HPCCM can generate Dockerfiles or Singularity Definition Files from a high level Python recipe. However, HPCCM essentially uses the Python to define a set of its own recipe specifications, which has relatively high requirements for users. On the other hand, due to the lack of domain knowledge, HPCCM cannot provide users with recommendations for key entities. Therefore, users must have relatively professional computer knowledge (such as Python and recipe syntax specifications) to implement customized recipes for HPC applications.

There are two challenges to realize the automatic generation of recipes in the HPC field:

(i) (C1) How to parse recipe files and extract entities and entity relationships from them

(ii) (C2) How to apply the obtained entities and entity relationships to the automatic generation of recipes

To address (C1), we first design and implement a two-phase parsing method for Singularity recipes and a relationship miner to extract key entities of recipes and mine relationships between entities. For (C2), we consider that the dependencies of software packages can be expressed more efficiently with graph data structures, so we store the acquired knowledge in a standardized graph database such as Neo4j. The knowledge graph provides data support for the automatic generation of recipes, which has anexcellent scalability. In the automatic generation of recipes, we improve the tag-based recommendation method to meet HPC users' personalized and diverse needs.

In summary, we make four core contributions:

(1) A unique toolset is designed for Singularity recipes to automatically extract the knowledge required for image construction and mine the associations between entities

(2) We build a knowledge graph of HPC containers to provide support for automatic recipe generation. The knowledge graph also provides functions such as entity recognition and entity-relationship query

(3) An improved recommendation method based on TF-IDF is designed, significantly improving recommendation performance

(4) Burner: an automatic recipe generation system. It is worth mentioning that Burner supports both Singularity Definition File and Dockerfile rule specifications

The original recipe dataset and parsing results can be obtained at https://github.com/jhshz520/BurnerRecipe. The rest of this paper is organized as follows: Section 2 reviews related work. Section 3 presents the overall design of the Burner. Section 4 introduces the construction of domain knowledge graph, mainly including two-phase parsing of recipes, entity extraction and entity relationship mining. The automatic generation of recipes based on knowledge graphs is described in Section 5. Section 6 is the performance evaluation of our toolset and Burner system. The last section draws conclusions and proposes future work.

2. Related Work

2.1. Container Technology in HPC. Charliecloud is an open source software based on the user-defined software stack (UDSS), emphasizing that it can be executed without users having root permissions. Charliecloud is a lightweight container implementation with a small code size of only about 800 lines. However, its functions, portability, and dependencies are slightly insufficient, and it cannot provide a powerful reproduction mechanism [9]. NERSC cooperated with Cray to develop Shifter [10, 11]. The main idea of Shifter is to reuse some components of Docker workflow and improve the runtime engine to meet the needs of HPC applications. Shifter reuses key components of the Docker ecosystem, rewriting the Docker runtime. However, the setup and management of Shifter are also relatively complex. Sarus [12] builds around the OCI specification, uses runc as the container runtime, and extends the functionality of HPC use cases by using OCI Hook, but it is not much different from Charliecloud and Shifter, all of them need to be used with the modified Docker containers to achieve targets for applications in HPC. Singularity is currently the best container solution in the HPC environment. It has a unique security model that allows untrusted users to safely run untrusted containers on multi-tenant systems. A special image format Singularity Image Format (SIF) is used to package and distribute containers. This compressed single-layer image format greatly reduces the storage space of the image and

facilitates the distribution of the image with better performance. In addition, Singularity implements cryptographic signature and verification with excellent portability and repeatability [13]. Singularity allows user-defined/managed/created containers to be easily integrated into existing HPC workflows and also provides compatibility with older OS versions via the setuid launcher. As of December 2021, Singularity has three major version iterations with many useful features.

2.2. Recipe Analysis. Singularity appeared in recent few years, and the application scenarios are not as extensive as Docker. At present, there is still a lack of research on Singularity Definition File, but the existing researches on Dockerfile have great reference significance. Cito et al. [14] conducted an exploratory analysis of the Docker container ecosystem on Github, and the research dataset contained more than 70,000 Dockerfiles. After comparing the most popular top100 and top1000 projects, it was found that up to 34% of the Dockerfiles in these projects could not be successfully built due to various problems and 28.6% of the quality problems were caused by the lack of version tags. Schermann and Zumberi [15] collected structured data about the status and changes of Dockerfiles from over 15,000 projects on Github and stored them in a PostgreSQL database. Zhang et al. [16] studied the impact of Dockerfile evolution trajectory on Dockerfile quality and corresponding image build latency. It was found that the fewer the number of image layers and the larger the space occupied by each layer of images, the fewer image quality problems and the shorter the build latency. By using the Dockerfile Lint tool Hadolint [17] to perform static analysis on a large number of Dockerfiles, Lu et al. discovered a problem in Dockerfiles that they called "Temporary File Smell." In the process of image building using Dockerfile, due to Docker's Copy On Write mechanism, inappropriate writing order of Dockerfile instructions will result in redundant temporary files in the Docker image [18, 19]. Yin et al. [20] proposed the STAR method, which solved the tag recommendation problem for Docker image repositories without training data. Hassan et al. [21] developed the Rudsea tool, which could implement Dockerfile update prompts based on analysis of changes in the software environment.

2.3. Software Domain Knowledge Graph. Knowledge graphs are not only widely used in search engines, question answering systems [22], and medical service support, but also play an important role in software reuse. Lin et al. formed an intelligent development environment IntelliDE [23] by aggregating, mining and analyzing software big data, and providing assistance to developers in the software development life cycle. DoekerPedia [24] proposed by Osorio et al. is the first known knowledge graph related to Docker images. Clair [25] is used to detect vulnerabilities in image instances to obtain information about software package versions and their vulnerabilities. DockerizeMe proposed by Horton et al. [7] builds a knowledge base of dependencies between Python packages and APT packages.

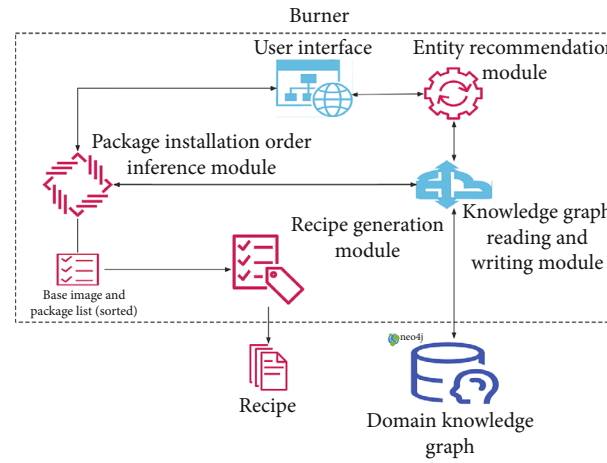


Figure 1: Design of Burner.

3. Burner

The main purpose of Burner is to solve the problem of automatic recipe generation. We solve the dependency problem by building an offline knowledge base and design an inference algorithm to return dependencies in a feasible installation order. We use the Django framework to implement Burner as a web application that researchers can use by visiting the website.

As shown in Figure 1, Burner uses the knowledge graph to automatically generate recipes. The core modules of Burner include the knowledge graph reading and writing module, which mainly provide data support for other modules. The entity recommendation module can recommend entities such as base images and software packages according to the Tag selected by a user. The software package installation order inference module can infer the order of the software package entities selected by the user to form an ordered software package installation list. Finally, the recipe generation module generates instructions according to the rules of the recipe, and saves the generated instructions in the form of files.

4. Construction of Knowledge Graph in HPC Container Domain

The knowledge graph is the cornerstone for our automatic generation of recipes and can provide strong support for dependency inference. Therefore, we start with the construction of domain knowledge graph to illustrate our work. As shown in Figure 2, the construction of knowledge graph in the field of HPC container mainly includes four parts: raw data acquisition, recipe parsing, knowledge fusion, and knowledge storage.

4.1. Ontology of Knowledge Graph. The ontology of the knowledge graph in the HPC container domain is shown in Figure 3, which includes 4 entity types and their attributes and 8 entity relationships.

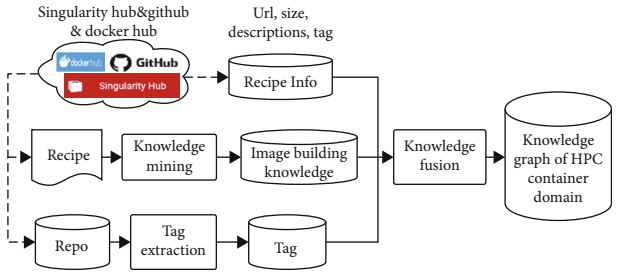


Figure 2: Overview of knowledge graph construction.

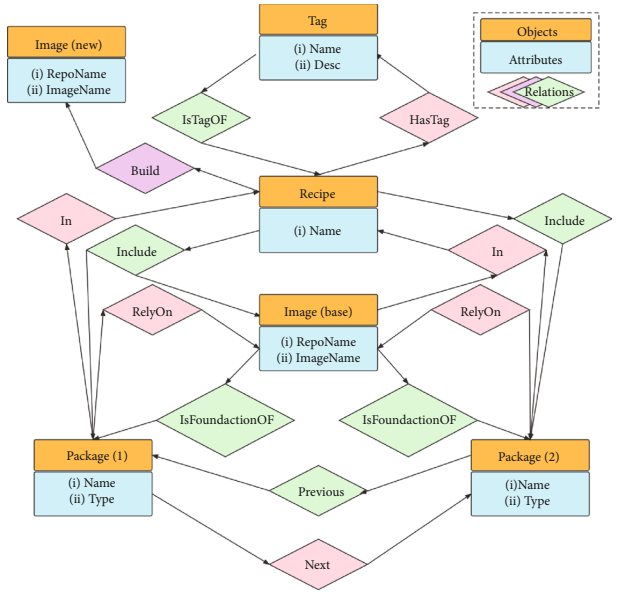


Figure 3: Ontology of knowledge graph in HPC container domain.

4.2. Data Acquisition. The data mainly comes from Singularity Hub, a publicly available platform for building and deploying scientific containers, which provides great convenience for reproducible scientific fields [26]. We use a customized crawler to collect and organize the recipes and their authors, tags, and other information. The raw data we obtained contains 530 tags and more than 1000 published recipes for HPC applications.

4.3. Two-Phase Parsing of Recipe. Code 1 is an example of a Singularity Definition File, where bash statements are usually nested [27]. We design a two-phase parsing method to parse the nested bash statements in recipe. The first stage is the instruction parsing, and AST parsing is performed according to the grammar specification defined by Singularity. The second stage parses the bash statements nested in the ASTs.

The first stage identifies each instruction according to the grammar of the recipe. The Singularity Definition File has different instruction blocks. Except for *Bootstrap* instruction and *From* instruction, all other instructions start with %. Therefore, regular expressions can be used to match and divide instructions, and each instruction can be parsed into an AST node, as shown in Figure 4.

```
1 Bootstrap: docker
2 From: ubuntu:16.04
3
4 %post
5 apt-get -y update
6 apt-get -y install fortune cosway lolcat
7
8 %runscript
9 fortune | cowsay | lolcat
```

Code 1: Singularity Definition File fragment named lolcow.def..

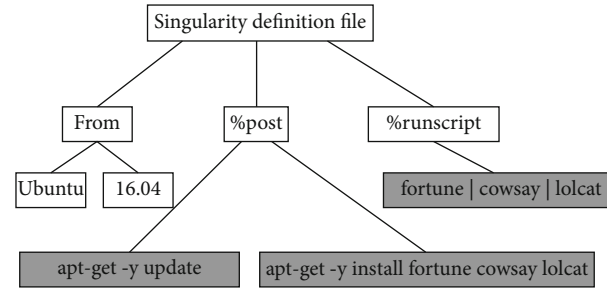


Figure 4: The first stage of AST parsing.

Table 1: 50 most commonly used Bash commands in recipes.

Categories	Name
System management	adduser, useradd, sudo, groupadd, nproc
File management	cp, rm, chmod, find, ln, chown, mv, mktemp
Disk management	cd, mkdir, pwd
System setting	set, export, gpg, ldconfig, sha256sum
Backup and compression	tar, unzip
Document editing	grep, sed, echo, wc
Package management tool	apt, apt-get, apt-key, apt-add-repository, yum, npm, yarn, gem, dpkg, dnf
Download tool	wget, curl, git, pip
Script run tool	bash, sh, python, php, go
Build tool	make, cmake, config
Reserved word	true

In the second stage of parsing, through command analysis, it is known that the information of the base image is in the "From" command field of the recipe and information of the packages is mainly in the "%post," "%environment," and "%runscript" instructions. However, the bash statements are often nested in the "%post" and "%runscript," which are numerous and varied.

It is impractical to design corresponding parsing methods for all Bash instructions, so we classified and counted these Bash instructions and found that 80% of the Bash command line calls are included in the 50 most commonly used commands. The names and classifications of the 50 commands are shown in Table 1. In this paper, we design a Bash statement parser for these 50 instructions by

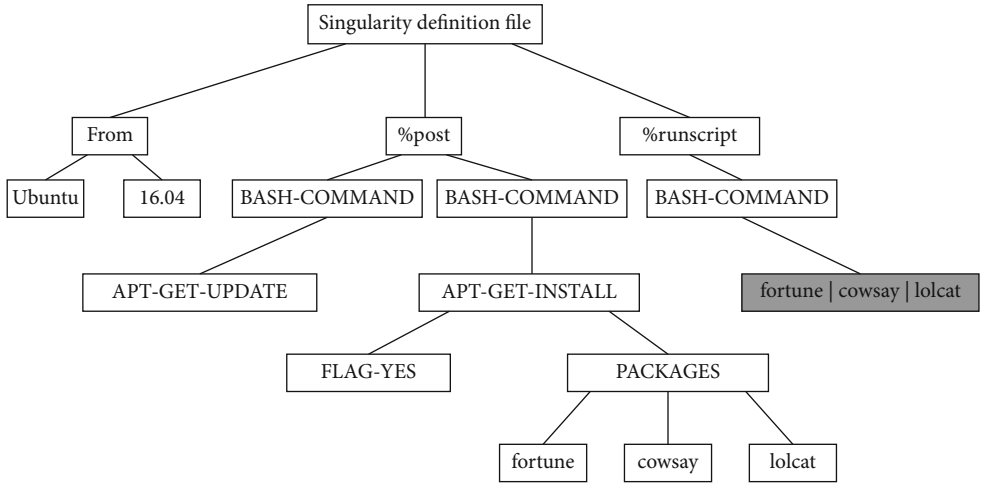


Figure 5: The second stage of AST parsing.

Input: packages_list, min_support, minConf
Output: package_pair_list
1 C1 = create_C1(packages_list)
2 L1, support_data = generate_L1_by_C1(packages_list, C1, min_support)
3 L = [L1]
4 C2 = aprioriGen(L[0])
5 L2, support_data = scanPackagelist(packages_list, C2, min_support)
6 L = L.append(L2)
7 package_pair_list = generateRules(L, support_data, minConf)

Algorithm 1: Software Package Association Mining Algorithm.

referring to the command manual and official documents of these instructions. The Bash statement parser is implemented by modifying the shellcheck tool [28].

As shown in Figure 5, after the second stage of parsing, the AST is generated. The parsing of commonly used Bash statements greatly enriches the content of the abstract syntax tree, which also provides a foundation for the extraction of software package entities and the mining of dependencies.

In the parsing example, *apt* – get – *y update*, *apt* – get – *y installfortunecowsaylolcat*, and *fortune*|*cowsay*|*lolcat* cannot be parsed in the first stage, but in the second stage, *apt* – get is one of the common commands, which can be further recognized and parsed by the Bash statement parser, while *fortune*|*cowsay*|*lolcat* cannot be further parsed because it is not a common command.

4.4. Entity Extraction and Entity Relationship Mining. Entity extraction can be performed from the ASTs generated by parsing. The entities we focus on are mainly base images and software packages. The information about a base image can be obtained from the child nodes of *From* node. We can traverse the subtree with %*post* and %*runscript* nodes as the root node to find the PACKAGES nodes for package information. During the traversal process, we can obtain the installation method of the packages from nodes such as APT-GET-INSTALL, YUM-INSTALL, and PIP-INSTALL. While traversing the tree, the appearance order of software packages in the PACKAGES node is also recorded, which is convenient for subsequent mining of software package dependencies.

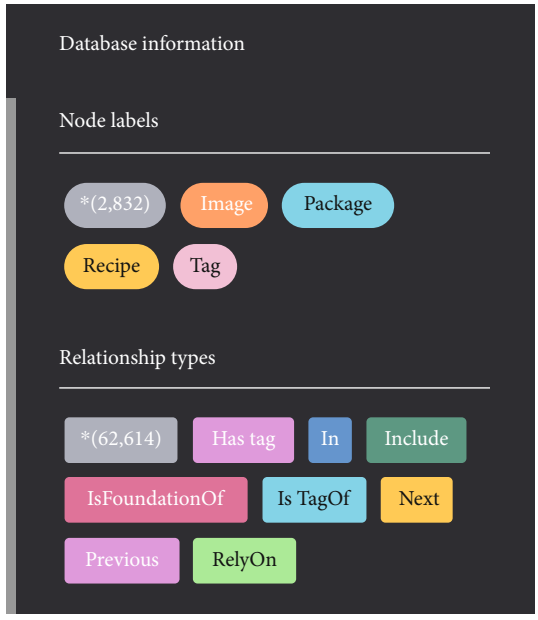


Figure 6: Information of Neo4j domain knowledge graph.

Table 2: Entities and their properties.

Node type	Attributes	Explanation
Recipe	Name	The name of the recipe
Image	RepoName	The name of the project where the base image is located
Image	ImageName	The name of the base image
Package	Name	The name of the package
Package	Type	The type of the package installation method, there are four types in total: pip, pip3, apt-get, and yum
Tag	Name	The name of the tag
Tag	Desc	The description of the tag

Table 3: Entity relationships.

Relation type	Explanation
Include	The relationship between recipe and its base image or software package, indicating that recipe contains a certain base image or a software package
In	The relationship between base image or software package and recipe, indicating that the base image or software package is included in the recipe
RelyOn	The relationship between package and base image means that the package depends on the base image for installation and configuration
IsFoundationOf	The relationship between base image and package, which means that the image is a starting image for the package to install and configure
IsTagOf	The relationship between tag and recipe, indicating that the tag is a label of the recipe
HasTag	The relationship between recipe and tag, indicating that the recipe has the tag it points to
Previous	The relationship between packages, indicating that the current package appears in the recipe before the package it points to
Next	The relationship between packages, indicating that the current package appears in the recipe after the package it points to

The main purpose of the software package entity association mining is to find the predecessor and successor relationships between software packages [29]. We use the Apriori algorithm to mine package dependencies [30]. If the confidence level of the association rule $pkg1 \longrightarrow pkg2$ is 1.0, it means that $pkg2$ can be installed under the condition that the package $pkg1$ is known to be installed; then, we can consider that the package $pkg2$ is a dependent package of the package $pkg1$. The mining algorithm is shown in Algorithm 1. We set min_support as the reciprocal of the minimum frequency of software packages so that the dependencies between software packages can be mined to the greatest extent. The minimum confidence is set to the most commonly used 0.8.

4.5. Knowledge Fusion and Knowledge Graph Construction. Knowledge fusion [31] is to unify and standardize the knowledge extracted from different recipes. The acquired knowledge is uniformly encoded with all entities as nodes and all entity relationships as edges. This unified code is the unique identification of the entity or entity relationship in the knowledge graph. Finally, the knowledge is stored in the graph database Neo4j (see Figure 6), which contains 2832 entities and 62614 relationships. The standardized knowledge base can provide support for the customized generation of subsequent recipes. The entities and their attributes are shown in Table 2, and the entity relationships are shown in Table 3.

5. Implementation of Burner

The most important modules of Burner are the entity recommendation module and the installation order inference module. In the recommendation module, we improve the tag-based recommendation method, which can avoid the influence of popular tags and popular items on the recommendation effect. In the installation order inference module, we use a graph algorithm to supplement package dependencies and determine the package installation order.

5.1. Tag-Based Base Image and Package Recommendation. Tag is the label that the user marks on the recipe, but a recipe contains a base image and multiple package entities. There is no direct relationship between tags and these entities. The most commonly used software such as *git* and *wget* are widely present in recipes. How to find the entity in the recipes that can well represent the Tag is a problem worth thinking about.

In Figure 7, we count the number of occurrences of tags; it can be observed that tags conforms to the long-tailed distribution [32]. In order to better meet the needs of user personalization, we design a tag-based recommender system inspired by the idea of TF-IDF [33]. The simplest tag-based recommendation method counts the number of times tagged by tags to recommend items. In the actual system, according to the tags selected by the user, the corresponding most popular items are searched for the recommendation. However,

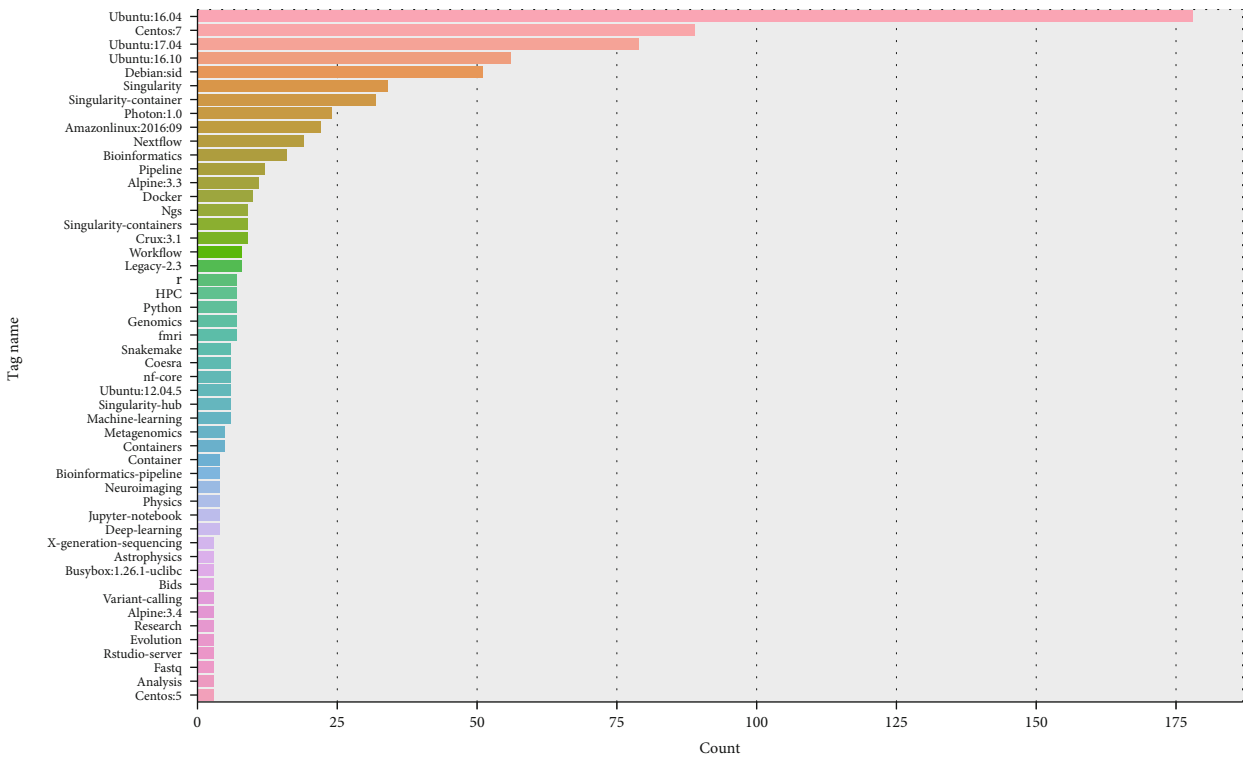


Figure 7: Statistics of Tag occurrences.

the apparent disadvantage of this recommendation method is that popular tags and popular items have a considerable weight, which dramatically reduces the novelty of the recommendation results. To this end, we have optimized the tag-based recommendation algorithm by drawing on the idea of TF-IDF. The core of our recommendation algorithm is based on the fact that a software package has appeared under a certain tag and hardly appears in other tags, so it can be considered that the software package is the core package under this tag.

As shown in Formulas (1), (2), and (3), $n_{i,j}$ represents the number of times that pkg_i is marked with tag_j, and $\sum_k n_{k,j}$ represents the total number of times that all software packages are marked with tag_j. $|Tag|$ indicates the total number of tags, $|\{j : pkg_i \in t_j\}|$ represents the number of tags used to mark pkg_i by the user. In Formula (3), $|\{j : pkg_i \in t_j\}| + 1$ can prevent the denominator from being 0:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}, \tag{1}$$

$$idf_i = \lg \frac{|Tag|}{|\{j : pkg_i \in t_j\}| + 1}, \tag{2}$$

$$Weight_{i,j} = tf_{i,j} \times idf_i. \tag{3}$$

5.2. Dependency Complement and Order Inference. Algorithms 2 and 3 show the complete process of package dependency complementation and package order inference. Algorithm 2 is implemented based on depth-first search

```
Input: core_pkgs
Output: complete_pkgs_set
1 foreach pkg in core_pkgs do
2     dp_pkgs = searchPredecessorByDFS(pkg)
3     complete_pkgs_set.add(dp_pkgs)
4 end
```

Algorithm 2: Package Dependency complementary Algorithm.

(DFS) by taking advantage of the transitive nature of package dependencies. After the user specifies the core packages related to the application, some dependency packages that these core packages depend on may not be included. The representation of package dependencies in the graph is that there is a directed edge between the package and the dependent package. After obtaining the dependencies of the software package list to be installed, the inference module will add dependency packages together with the core software packages specified by the user to the final set of software packages to be installed.

Figure 8 shows the possible dependencies of software packages in the graph. In actual use, we only consider the relationship of Previous, because Previous and Next appear in pairs and their functions are equivalent. The packages pointed to by the edges of Previous in the subgraph are in the first order.

The inference of the software package order in Algorithm 3 is to use the topological sorting method to sort all the software packages to be installed. The core idea of

```
Input: complete_pkgs_set
Output: sorted_pkgs_list
sub_graph = exractSubGraphFromKG(complete_pkgs_set)
out_degree_count = countOutDegreeForGraph(sub_graph)
while sub_graph is not empty do
    foreach pkg in zeroOutDegree(sub_graph) do
        sorted_pkgs_list.append(pkg)
        foreach pkg_next in nextNode(pkg) do
            pkg_next_out_degree -=1
        end
        remove_node(pkg, sub_graph)
    end
end
```

Algorithm 3: Package installation order inference Algorithm.

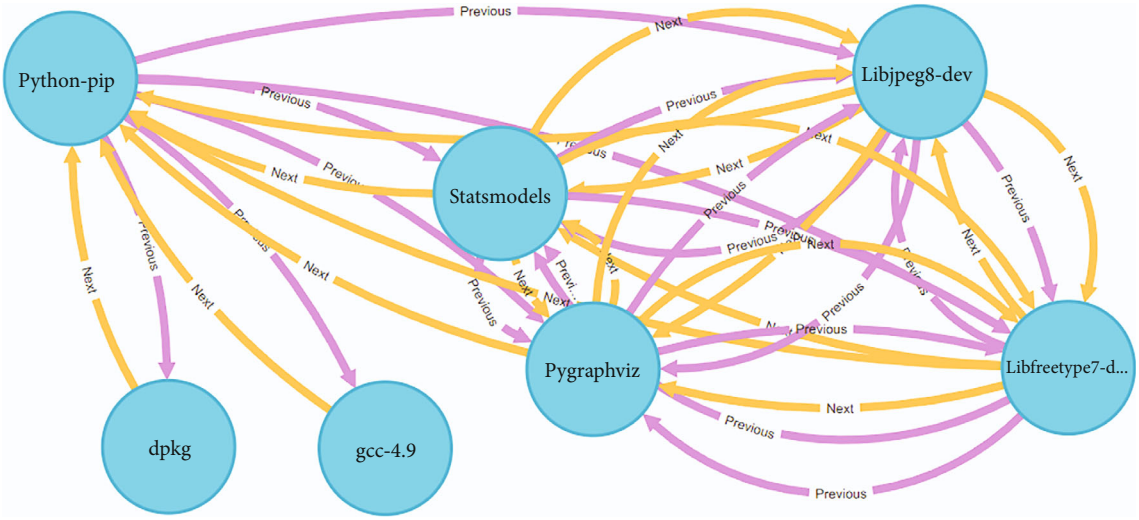


Figure 8: Example of a subgraph of package dependencies in Burner.

topological sorting is to continuously remove the nodes with zero out-degree in the subgraph of the software package until the subgraph is empty. If a node in the subgraph has no out-degree, the software package represented by this node can be installed directly without depending on other software packages. During the iteration, the node with out-degree zero is added to the sorted list, and all dependent edges of the node are removed from the subgraph. By repeating the above process until the graph is empty, the installation order of the packages is finally obtained.

6. Evaluation

6.1. Burner Demonstration. Burner is very friendly for HPC users even with no computer background. In the process of generating recipes using Burner, users do not need to perform any text input operations; during the entire interaction, they can complete the generation of customized recipes only by selecting operations.

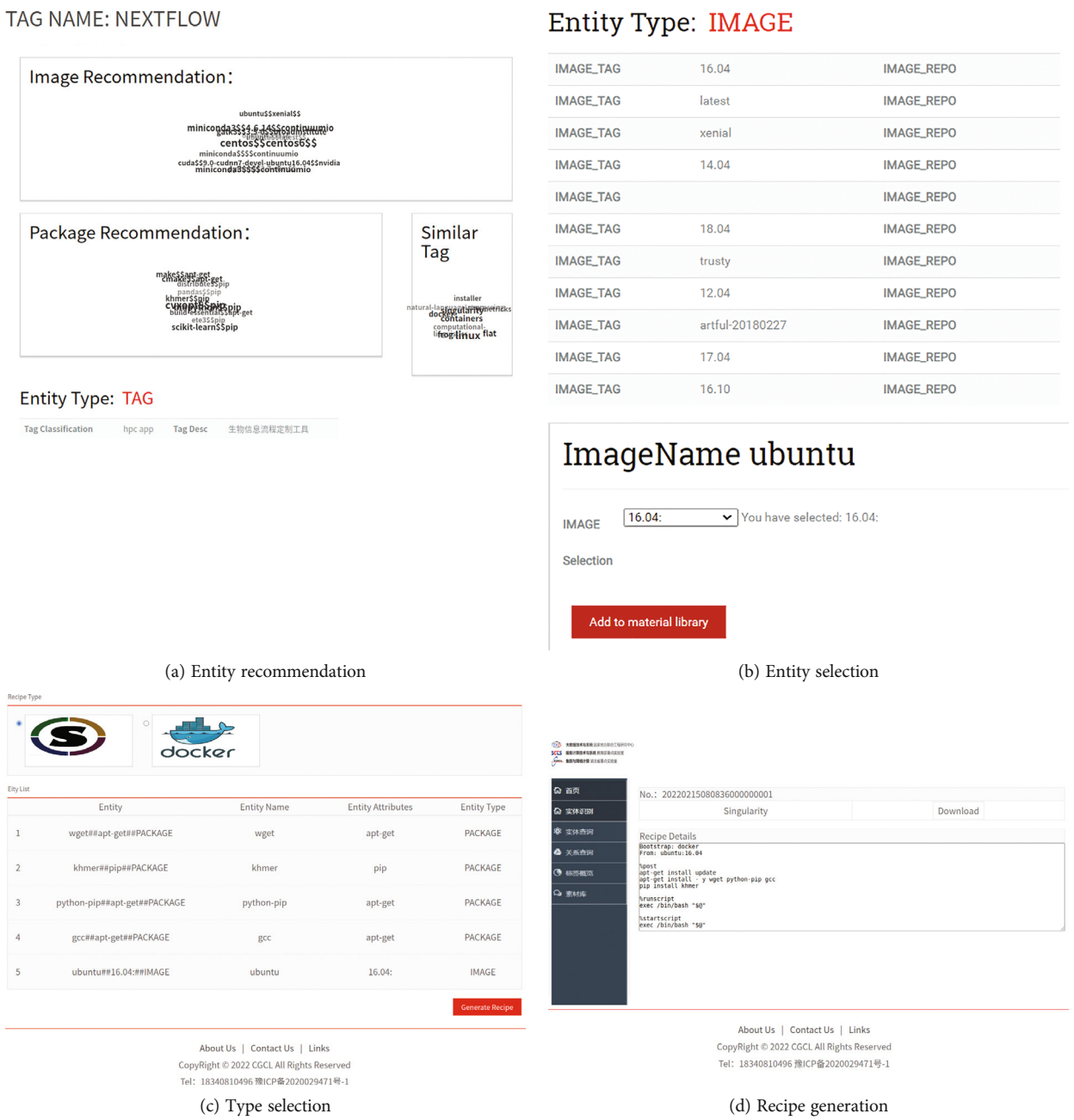
In Figure 9, we take the *nextflow* tag as an example to demonstrate the generation of Singularity Recipe. Figure 9(a) shows the recommendation of software packages and base images based on the tags selected by the user, and the recommended entities are displayed in a dynamic word cloud. Then, the users can select the required base image and software packages to add them to the material library (see Figure 9(b)), and the back end of the material library use the Redis database to quickly perform operations such as additions and deletions. As shown in Figure 9(c), after completing the selection of materials, the users can also specify the type of recipe. Currently, the system supports Singularity Definition File and Dockerfile. Users can preview the generated recipe online or perform operations such as edit, download, and delete (see Figure 9(d)).

6.2. Metrics

6.2.1. Evaluation Metrics for Recipe Parser

$$PSR = \frac{|Node_{total}| - |Node_{unknown}|}{|Node_{total}|}. \quad (4)$$

To quantify the performance of our recipe parser, we define a parsing success rate in this paper. After the



(a) Entity recommendation

(b) Entity selection

(c) Type selection

(d) Recipe generation

Figure 9: Recipe generation demonstrate.

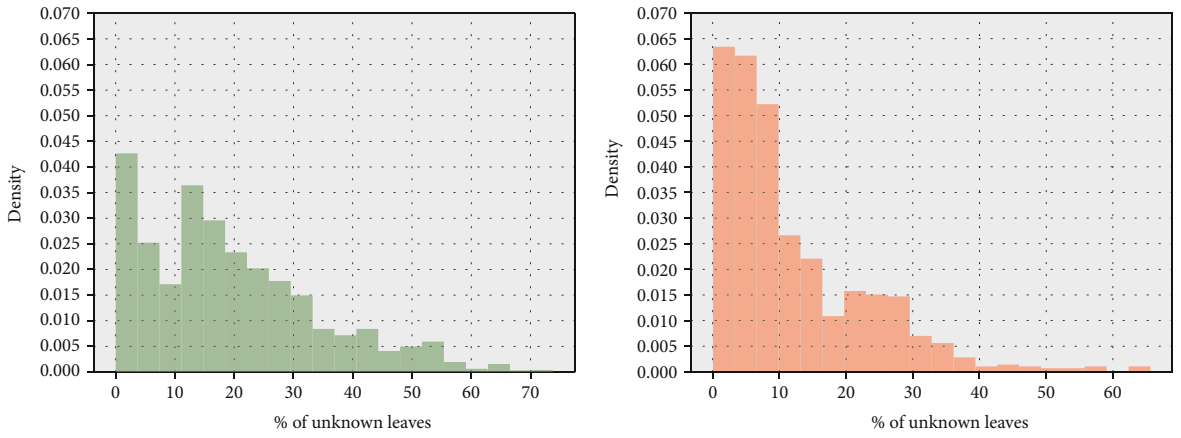
parsing is completed, we uniformly mark the nodes that cannot be parsed as UNKNOWN (gray nodes in Figures 4 and 5). We use $|Node_{total}|$ represents the total number of nodes in the AST, and $|Node_{unknown}|$ represents the number of UNKNOWN nodes. The definition of parsing success rate (PSR) is shown in Formula (4), a larger value of PSR means that the parser has stronger performance.

6.2.2. Evaluation Metrics for Recommendation Performance. In the practical application, we use tags for *TopN* recommendation [34]. Recommended entities include base images and packages. We use the mainstream *Precision*, *Recall* and *F1* to measure the recommendation performance of the system. The definitions of recommendation metrics are shown in Formulas (5), (6), and (7):

$$Precision = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |R(u)|}, \tag{5}$$

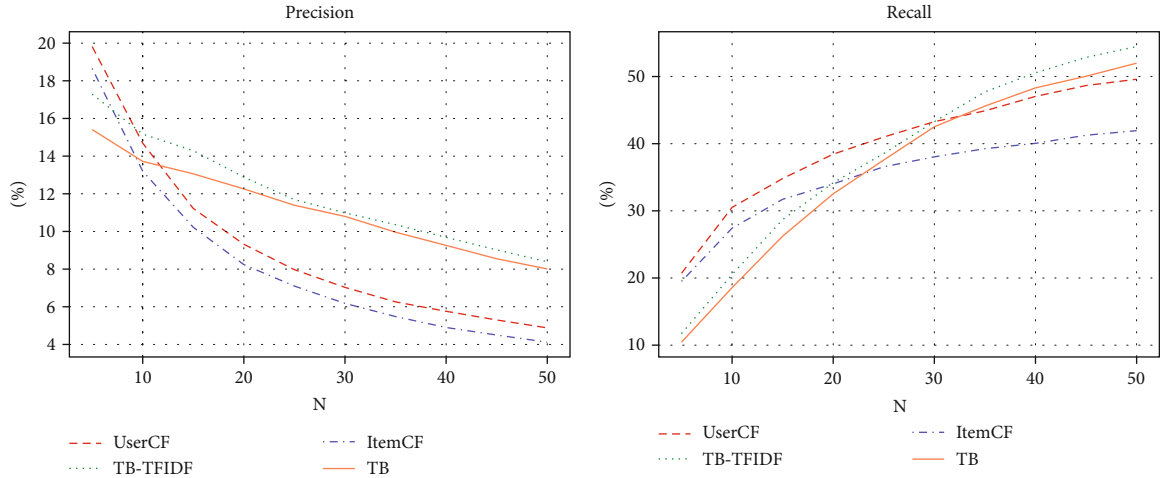
$$Recall = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |T(u)|}, \tag{6}$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall}. \tag{7}$$



(a) Stage I (b) Stage II

Figure 10: Density histograms of UNKNOWN nodes for two parsing stages.



(a) Precision (b) Recall

Figure 11: Precision and recall of four recommendation methods.

6.2.3. Evaluation Metrics for Image Build

$$BSR = \frac{|Recipe_{bs}|}{|Recipe_{total}|}. \tag{8}$$

We use the build success rate (BSR) as a functional indicator of the system. Whether the image can be successfully built can intuitively represent the quality of the generated recipes. The definition of BSR is shown in Formula (8). $|Recipe_{total}|$ represents the total number of generated recipes, and $|Recipe_{bs}|$ represents the number of recipes that can successfully build the container images.

6.3. Experimental Results and Analysis

6.3.1. Two-Phase Parsing Method. We performed statistical analysis on ASTs generated by 1000 recipes. The density histograms of the distribution of UNKNOWN nodes in two parsing stages are shown in Figure 10.

After the first stage of parsing, 28.3% of the nodes are marked as UNKNOWN as shown in Figure 10(a). As shown

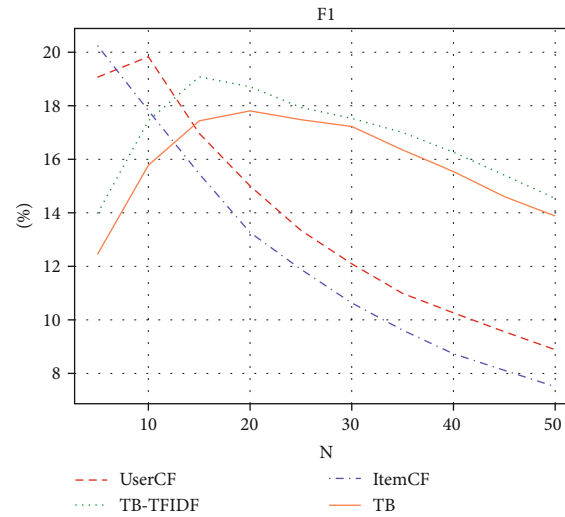


Figure 12: F1 of the four recommendation methods.

Table 4: Performance comparison of the four recommendation methods.

N	Precision				Recall				F1			
	UserCF	ItemCF	TB-TFIDF	TB	UserCF	ItemCF	TB-TFIDF	TB	UserCF	ItemCF	TB-TFIDF	TB
5	19.82	18.66	17.29	15.41	20.7	19.49	11.72	10.44	19.066	20.25	13.97	12.447
10	14.69	13.2	15.17	13.72	30.51	27.42	20.51	18.56	19.832	17.821	17.44	15.777
15	11.22	10.22	14.29	13.06	34.81	31.72	28.7	26.22	16.97	15.459	19.08	17.435
20	9.32	8.24	12.88	12.26	38.44	34.01	34.18	32.53	15.003	13.266	18.71	17.808
25	7.97	7.1	11.69	11.39	40.99	36.56	38.54	37.57	13.345	11.891	17.939	17.48
30	7.03	6.18	10.99	10.8	43.28	38.04	43.28	42.52	12.095	10.633	17.529	17.225
35	6.26	5.48	10.36	9.96	44.89	39.25	47.71	45.6	10.988	9.617	17.004	16.349
40	5.76	4.9	9.69	9.26	47.04	40.05	50.56	48.31	10.263	8.732	16.263	15.541
45	5.3	4.5	9.02	8.55	48.66	41.26	52.85	50.04	9.559	8.115	15.409	14.605
50	4.88	4.12	8.39	8.01	49.6	41.94	54.47	51.99	8.886	7.503	14.54	13.881

in Figure 10(b), in the second stage, the density of recipes with high PSR increases significantly, and the PSR in some ASTs even reaches 100%. On average, only 18.2% of the nodes could not be parsed in the second stage, and the PSR increased by 10.1%. Then, we analyzed the recipes with low PSR and found that the main reason was that some Bash commands in these recipes were not commonly used or the Bash statements were nested too deeply. The results show that our two-phase parsing method can effectively perform recipe parsing and entity extraction.

6.3.2. Recommendation Performance Test. Taking recommending software packages to users as an example, we compare the performance of four recommendation methods. The four methods are UserCF, ItemCF, TB, and TB-TFIDF. UserCF and ItemCF do not use tag information, only use user and item information as input. TB simply uses statistical information, and TB-TFIDF uses TF-IDF to greatly improve TB. There are two hyperparameters K and N in UserCF and ItemCF. K represents the selection of K users with the most similar interests to the recommended user, and N represents the number of items recommended to the user. After grid search and tuning, the optimal value of K is set to 80. Figures 11 and 12 show the performance of the four recommendation methods under different N values; the detailed results are shown in Table 4.

It can be observed that with the increase of N, the *Precision* of UserCF and ItemCF has a relatively significant decline. The recommendation method of TB-TFIDF is relatively stable, and the *Precision* is usually above 10%. From the $F1$ value that measures the overall performance of the recommender system, the effect of TB-TFIDF is also the best. The average number of packages contained in each recipe is 23. In practical applications, we set the N value to 20 or 25. Experiments show that the recommendation performance TB-TFIDF is best. The TB-TFIDF recommendation method also does not have the problem of cold start, which is more in line with the actual application scenario.

6.3.3. Image Build Test. In the image build test, 50 different tags are selected from the tag list by executing a random function. For each Tag, entity recommendation and automatic recipe generation are performed through the Burner system. In the recipe corpus of this paper, operating system images account for a large proportion of all base images; these operating system images are dominated by Ubuntu and Centos; the sum of Ubuntu and Centos accounts for more than 80% (see Figure 13). Therefore, for the sake of standardization, the base images are uniformly designated as ubuntu: 16.04 and centos: 7. For the same Tag, we used Burner to generate two types of recipes (Singularity Definition File and Dockerfile).

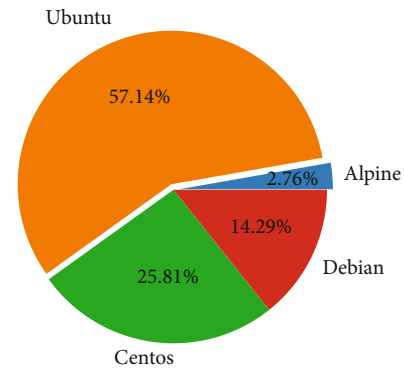


Figure 13: Types and proportions of OS images in recipes.

Table 5: Results of image build test.

Container type	Base image	Build successfully recipes	BSR
Singularity	Ubuntu	20	80%
Singularity	Centos	14	56%
Dockerfile	Ubuntu	18	72%
Dockerfile	Centos	15	60%

As shown in Table 5, in the Singularity Definition File image build test, it was found that 34 of the 50 recipes generated by Burner could successfully build the images; the average image construction success rate reached 68%. Among Singularity recipes, 20 of the 25 recipes with Ubuntu: 16.04 as the base image could successfully build the container images; the image building success rate reached 80%. At the same time, 14 of the 25 recipes with

Table 6: Results of Dockerfile detection with Hadolint.

Rule	Default severity	Result	Description
DL3006	Warning	0	Always tag the version of an image
DL3007	Warning	0	Pin the version explicitly to a release tag
DL3038	Warning	0	Use the-y switch to avoid manual install package
DL3059	Info	0	Multiple consecutive RUN instructions
DL3061	Error	0	Dockerfile must begin with FROM
DL4003	Warning	0	Multiple CMD instructions found
DL3007	Warning	0	MAINTAINER is deprecated
DL3008	Warning	23	Pin package versions in apt-get install
DL3033	Warning	19	Pin package versions in yum install
DL3013	Warning	21	Pin package versions in pip
DL4001	Warning	19	Either use W get or curl but not both
DL3049	Info	50	Label is missing

the Centos: 7 as the base image could successfully build the container images. The difference between the results of Ubuntu and Centos was caused by insufficient Centos recipe samples and limited entity knowledge extracted.

After the image build test, we used the Hadolint tool to detect the Dockerfile, and the results are shown in Table 6. Violations such as DL3006 and DL4000 have been eliminated in the automatically generated Dockerfile; DL3008 and DL3013 have also been improved. The reason of DL3008 and DL3013 cannot be eliminated is that the knowledge of software packages and their versions in the current knowledge base is not sufficient. It can be foreseen that with the enrichment of the knowledge base, DL3006 and DL4000 problems will be improved.

Finally, we analyzed the build logs of the recipes that failed to build, found that the reasons for the failure included environment variable setting errors, missing compilation statements, and "apt-get update" network errors. To solve the above problems, it is necessary to manually further increase the configuration of environment variables and other measures. The highest BSR is 80%, which proves the system can better help users to write recipes, and the Hadolint detect results also prove that the recipes automatically generated by Burner have high quality.

7. Conclusion and Future Work

Compared with the one-phase parsing method, the two-phase parsing method we designed can parse recipes more efficiently. We use the extracted knowledge to build a relatively complete domain knowledge base. On the one hand, this knowledge graph can realize the fine-grained representation of knowledge. On the other hand, the use of graph data and graph algorithms can better solve the problem of dependencies. The automatic generation of recipes using knowledge can greatly reduce the burden of related developers. The recipe automatic generation system Burner can meet the individual needs of different users on the basis of improving the correctness of recipes. The design of Burner revolves around the two core issues of automation and personalization, and the automatic generation of recipes is finally achieved through the construction of knowledge base and the recommendation of entities.

At present, the amount of knowledge in the prototype system is still relatively small, and the dependency inference through this knowledge base may lack version information. In the future, higher-quality recipe generation can be achieved by expanding the scale of the knowledge base. In addition, the software packages in our knowledge base are all officially packaged software (OPS) registered in public repositories and can be installed using package management tools such as apt-get and pip. Some unofficially packaged software (UOPS) cannot be automatically downloaded and installed by package management tools. These UOPS usually need to specify the download address and also need to perform operations such as decompression and switching directory compilation to install. Further research is required for UOPS.

The new versions of Docker and Singularity have added a multi-stage build function, which supports the separation of the compilation environment and the running environment, allowing multiple FROM instructions to appear. This new feature can greatly reduce the size of the final image. We tend to support the multi-stage build function. We will collect the recipe application examples of multi-stage build and improve our research to support the multi-stage build function.

Data Availability

The original recipe dataset and parsing results can be accessed at https://github.com/jhshz520/BurnerRecipe.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key Research and Development Program of China under grant 2018YFB0204002.

References

[1] G. M. Kurtzer, V. Sochat, and M. W. Bauer, "Singularity: scientific containers for mobility of compute," *PLoS One*, vol. 12, no. 5, article e0177459, 2017.

[2] D. Nüst, V. Sochat, B. Marwick et al., "Ten simple rules for writing dockerfiles for reproducible data science," *PLoS Computational Biology*, vol. 16, no. 11, article e1008316, 2020.

[3] C. Boettiger, "An introduction to docker for reproducible research," *ACM SIGOPS Operating Systems Review*, vol. 49, no. 1, pp. 71–79, 2015.

[4] P. Souza, G. Kurtzer, C. Gomez-Martin, and P. C. Silva, "Hpc containers with singularity," in *Third EAGE Workshop on High Performance Computing for Upstream*, pp. 1–5, European Association of Geoscientists & Engineers, 2017.

[5] E. Horton and C. Parnin, "Gistable: evaluating the executability of python code snippets on github," in *2018 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pp. 217–227, IEEE, Madrid, Spain, 2018.

[6] J. Henkel, C. Bird, S. K. Lahiri, and T. Reps, "Learning from, understanding, and supporting devops artifacts for docker," in *2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE)*, pp. 38–49, IEEE, Seoul, Korea, 2020.

[7] E. Horton and C. Parnin, "Dockerizeme: automatic inference of environment dependencies for python code snippets," in *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, pp. 328–338, IEEE, Montreal, QC, Canada, 2019.

[8] S. McMillan, "Making containers easier with hpc container maker," in *Proceedings of the SIGHPC Systems Professionals Workshop (HPCSYSPROS 2018)*, Dallas, TX, USA, 2018.

[9] R. Priedhorsky and T. Randles, "Charliecloud: Unprivileged containers for user-defined software stacks in hpc," in *Proceedings of the international conference for high performance computing, networking, storage and analysis*, pp. 1–10, New York, 2017.

[10] D. M. Jacobsen and R. S. Canon, "Contain this, unleashing docker for hpc," in *Proceedings of the Cray User Group*, pp. 33–49, Chicago, 2015.

[11] L. Gerhardt, W. Bhimji, S. Canon et al., "Shifter: containers for hpc," *Journal of physics: Conference series*, vol. 898, article 082021, 2017.

[12] L. Benedicic, F. A. Cruz, A. Madonna, and K. Mariotti, "Sarus: highly scalable docker containers for hpc systems," in *International Conference on High Performance Computing*, pp. 46–60, Springer, Cham, 2019.

[13] D. Godlove, "Singularity: simple, secure containers for compute-driven workloads," in *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (learning)*, pp. 1–4, New York, 2019.

[14] J. Cito, G. Schermann, J. E. Wittern, P. Leitner, S. Zumberi, and H. C. Gall, "An empirical analysis of the docker container ecosystem on github," in *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*, pp. 323–333, IEEE, Buenos Aires, Argentina, 2017.

[15] G. Schermann, S. Zumberi, and J. Cito, "Structured information on state and evolution of dockerfiles on github," in *Proceedings of the 15th International Conference on Mining Software Repositories*, pp. 26–29, IEEE, New York, 2018.

[16] Y. Zhang, G. Yin, T. Wang, Y. Yu, and H. Wang, "An insight into the impact of dockerfile evolutionary trajectories on quality and latency," in *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 1, pp. 138–143, IEEE, 2018.

[17] "hadolint/hadolint," https://github.com/hadolint/hadolint.

[18] Z. Lu, J. Xu, Y. Wu, T. Wang, and T. Huang, "An empirical case study on the temporary file smell in dockerfiles," *IEEE Access*, vol. 7, pp. 650–659, 2019.

[19] J. Xu, Y. Wu, Z. Lu, and T. Wang, "Dockerfile tf smell detection based on dynamic and static analysis methods," in *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, pp. 185–190, IEEE, Milwaukee, WI, USA, 2019.

[20] K. Yin, W. Chen, J. Zhou, G. Wu, and J. Wei, "Star: a specialized tagging approach for docker repositories," in *2018 25th Asia-Pacific Software Engineering Conference (APSEC)*, pp. 426–435, IEEE, Nara, Japan, 2018.

[21] F. Hassan, R. Rodriguez, and X. Wang, "Rudsea: recommending updates of dockerfiles via software environment analysis," in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, pp. 796–801, IEEE, New York, 2018.

[22] Y. Chen, J. Kuang, D. Cheng, J. Zheng, M. Gao, and A. Zhou, "Agrikg: an agricultural knowledge graph and its applications," in *International conference on database systems for advanced applications*, pp. 533–537, Springer, Chiang Mai, Thailand, 2019.

[23] Z.-Q. Lin, B. Xie, Y.-Z. Zou et al., "Intelligent development environment and software knowledge graph," *Journal of Computer Science and Technology*, vol. 32, no. 2, pp. 242–249, 2017.

[24] M. Osorio, C. B. Aranda, and H. Vargas, "Dockerpedia: a knowledge graph of docker images and Their Metadata," *International Journal of Software Engineering and Knowledge Engineering*, vol. 32, no. 1, pp. 71–89, 2022.

[25] "An open-source tool from CoreOS designed to identify known vulnerabilities in Docker images," https://github.com/coreos/clair.

[26] V. V. Sochat, C. J. Prybol, and G. M. Kurtzer, "Enhancing reproducibility in scientific computing: metrics and registry for singularity containers," *PLoS One*, vol. 12, no. 11, article e0188511, 2017.

[27] "The container system for secure high performance computing," https://apptainer.org/docs/user/main/.

[28] V. Holen, "A shell script static analysis tool," https://github.com/koalaman/shellcheck.

[29] D. M. German, J. M. Gonzalez-Barahona, and G. Robles, "A model to understand the building and running interdependencies of software," in *14th Working Conference on Reverse Engineering (WCRE 2007)*, pp. 140–149, IEEE, Vancouver, BC, Canada, 2007.

[30] R. Agrawal, T. Imielin'ski, and A. Swami, "Mining association rules between sets of items in large databases," *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, vol. 22, no. 2, pp. 207–216, 1993.

[31] R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: Principles and methods," *Data & Knowledge Engineering*, vol. 25, no. 1-2, pp. 161–197, 1998.

[32] Y. Park, "The adaptive clustering method for the long tail problem of recommender systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 8, pp. 1904–1915, 2013.

[33] J. Ramos, "Using tf-idf to determine word relevance in document queries," *Proceedings of the first instructional conference on machine learning*, vol. 242, no. 1, pp. 29–48, 2003.

[34] J. Lin, H. Pu, Y. Li, and J. Lian, "Intelligent recommendation system for course selection in smart education," *Procedia Computer Science*, vol. 129, pp. 449–453, 2018.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7830508, 13 pages
https://doi.org/10.1155/2022/7830508

Research Article

A Scalable Blockchain-Based Integrity Verification Scheme

Zequan Zhou,[1] Xiling Luo,[1,2] Yi Bai,[1,2] Xiaochao Wang,[1,2] Feng Liu,[1] Gang Liu,[3] and Yifu Xu[1]

[1]*School of Electronic and Information Engineering, BeiHang University, Beijing, China*
[2]*Beihang Hangzhou Innovation Institute, Hangzhou, Zhejiang, China*
[3]*Zhejiang Scientific Research Institute of Transport, Hangzhou, Zhejiang, China*

Correspondence should be addressed to Xiling Luo; luoxiling@buaa.edu.cn

Received 2 February 2022; Revised 3 March 2022; Accepted 4 April 2022; Published 10 May 2022

Academic Editor: Yingjie Wang

Ensuring the integrity of remote data is the prerequisite for implementing cloud-edge computing. Traditional data integrity verification schemes make users spend a lot of time regularly checking their data, which is not suitable for large-scale IoT (Internet of Things) data. On the other hand, the introduction of a third-party auditor (TPA) may bring about greater privacy and security issues. We use blockchain to address the problem of TPA. However, implementing dynamic integrity verification with blockchain is a bigger challenge due to the low throughput and poor scalability of blockchain. More importantly, whether there is a security problem with blockchain-based integrity verification is not yet known. In this paper, we propose a scalable blockchain-based integrity verification scheme that implements fully dynamic operations and blockless verification. The scheme builds scalable homomorphic verification tags based on ZSS (Zhang-Safavi-Susilo) short signatures. We exploit smart contract technology to replace TPA for integrity verification tasks, which not only eliminates the risk of privacy leakage but also resists collusion attacks. Furthermore, we formally define a blockchain-based security model and prove that our scheme is secure under the security assumption of cryptographic primitives. Finally, the mathematical analysis of our scheme shows that both the communication complexity and the communication complexity of an audit are $O(c)$, in which c is the number of challenge blocks. We compare our scheme with other schemes, and the results show that our scheme has the lowest time consumption to complete an audit.

1. Introduction

The rapid development of the Internet of Things (IoT) brings huge amounts of data. IoT devices store data on the cloud for cloud-edge computing. However, ensuring the integrity of remote data is the prerequisite for implementing cloud-edge computing [1].

Traditional cloud data integrity verification schemes [2, 3] rely on techniques such as message authentication codes and hash functions to let users know the status of their data. Nonetheless, these heuristics have large computation and communication overheads since users need to retrieve all data. Some schemes [4, 5] reduce the verification overhead of the integrity verification system by constructing homomorphic verification tags. While these schemes enable quick auditing of data, users still need to spend a lot of time auditing their data periodically. To reduce the auditing burden on users, third-party auditors (TPAs) [6] are introduced to perform auditing tasks on cloud data. However, in real-world scenarios, TPAs are not completely trustworthy, and there are two threats [7–9]. First, a malicious TPA may extract data privacy by auditing the same data blocks over and over again. Second, a malicious TPA may collude with cloud servers to produce fake audit results.

Fortunately, blockchain smart contract technology [10, 11] makes it possible to address these issues simultaneously. Smart contracts are encapsulated scripts that can be automated for execution. Therefore, we can use smart contracts to perform auditing tasks instead of TPAs. However, the low throughput and poor scalability of blockchain make it difficult for blockchain to be used in dynamic cloud storage. Therefore, it is a huge challenge to address the scalability of

integrity verification schemes in blockchain network environments. More importantly, whether the security of integrity verification schemes is affected in the open network environment of blockchain should be noticed. To the best of our knowledge, there is no scheme that gives formal security proof. Therefore, it is essential to give proof of security for blockchain-based integrity verification schemes.

In this paper, we propose a scalable blockchain-based integrity verification scheme that enables fully dynamic actions such as insertion, deletion, and modification to address the issues raised above. We create a scalable homomorphic verification tag based on the ZSS (Zhang-Safavi-Susilo) short signature, which uses basic cryptographic hash functions such as SHA-1 or MD5 and does not require expensive specific hash algorithms to accomplish scalability. The scheme supports blockless verification that allows users to audit their data without retrieving all of it. In addition, we use blockchain smart contract technology instead of TPA for the task of integrity verification, which not only eliminates the risk of privacy leakage but also protects against collusion attacks. To evaluate the level of security of our scheme in a blockchain environment, we formally define a blockchain-based security model and demonstrate that the scheme is secure against adaptive chosen message attacks under the security assumption of cryptographic primitives.

1.1. Contributions. The following are the main contributions of this paper:

(1) We propose a scalable blockchain-based integrity verification (SBB-IV) scheme that implements fully dynamic operations and blockless verification. The scheme achieves scalability under blockchain networks by building scalable homomorphic verification tags (HVTs) based on ZSS short signatures, which use general cryptographic hash functions and do not require expensive special hash functions

(2) We exploit smart contract technology to replace TPA for integrity verification tasks, which not only eliminates the risk of privacy leakage but also resists collusion attacks. Furthermore, we formally define a blockchain-based security model that captures the semantic security of adaptive chosen message attacks (CMA). We show that the SBB-IV scheme is secure against adaptive CMA under the security assumption of the q-CAA problem

(3) The mathematical analysis of our scheme shows that both the communication complexity and the communication complexity of the scheme are $O(c)$, in which c is the number of challenge blocks. In addition, we do a series of tests on Hyperledger Fabric V2.2 and compare our scheme to the current state-of-the-art. Our technique is more efficient, as it takes only 2.3 seconds to conduct an audit when 1% of the data blocks are faulty

1.2. Paper Organization. The remainder of this work is arranged in the following manner. We provide an overview of related works in Section 2. Preliminaries are shown in Section 3. The network, threat, framework, protocol, and security model are all shown in Section 4. The detailed algorithms are presented in Section 5. We examine the correctness, dynamic, and security in Section 6. The mathematical analysis and experimental results are presented in Section 7. The paper comes to a close with Section 8.

2. Related Works

2.1. Traditional Data Integrity Verification. Provable data possession (PDP) [12] and proofs of retrievability (POR) [13] are two types of data integrity verification models. The PDP model was formally specified by Ateniese et al. [12], who presented an HVT based on RSA (Rivest-Shamir-Adleman) signatures. They separated the data into blocks and calculated the HVTs for each one. The user then chose a fixed number of blocks for verification at random. Although the sampling approach decreases the computing cost from linear to constant, the scheme is not capable of dynamic operations due to the fixed index of blocks. Juels et al. [13] presented a sentinel-based POR technique in which data segments (sentinels) were randomly inserted into the full data encoded using error correction codes. Due to the limited number of sentinels, it can only undertake limited auditing. BLS (Boneh-Lynn-Shacham) signatures were utilized by Shacham et al. [14] to create HVTs, which reduces communication overhead because the BLS signature is shorter than the RSA signature. Wang et al. [8] described how to build a dynamic PDP system using Merkle tree, an authenticated data structure. Similarly, Erway et al. [15, 16] proposed a skip-list-based dynamic-PDP (DPDP) system. Instead of using a fixed index, these data structures indicate block positions in terms of the order of leaf nodes, allowing blocks to be dynamically inserted at varied locations. However, because these data structures require supplementary information to validate the leaf node placements, they have a computational and communication complexity of $O(\log n)$, making them unsuitable for large-scale data. By first encrypting the data and then providing some precomputed hashes of the encrypted data to the TPA, Shah et al. [6, 17, 18] introduced a TPA to audit the data. The TPA, on the other hand, will be unable to continue auditing after the hashes run out. Furthermore, a hostile TPA may collect information by auditing the same data blocks over and over again. Although random mask approaches [5, 9, 19, 20] have been devised to obscure the linear combination of data and prevent the TPA from extracting it, they are still ineffective in preventing collusion attempts.

2.2. Blockchain-Based Data Integrity Verification. By replacing the integrity management service of centralized nodes with a fully decentralized data integrity service, Liu et al. [10] proposed a blockchain-based Internet of Things (IoT) data integrity service framework that eliminates TPA. However, as they only implemented the proposed protocol's basic features, the efficiency of building smart contracts for IoT devices is insufficient for large-scale IoT data. To assure data availability and privacy, Liang et al. [23] suggested a

Table 1: A comparison between our scheme and the state of art. (Comp. indicates computational complexity and Comm. indicates communication complexity; n indicates the number of all blocks, and c indicates the number of blocks to be audited; – indicates that the scheme does not involve the item.).

(a)

	Wang [8]	Erway [15]	Wang [7]	Hao [21]	Liu [10]	Yue [22]	Our scheme
With help of TPA	Yes	No	Yes	No	No	No	No
Public auditability	Yes	No	Yes	Yes	Yes	Yes	Yes
Privacy protection	No	—	Yes	Yes	Yes	Yes	Yes
Data dynamics	Yes	No	No	Yes	No	Yes	Yes
Support for sampling	Yes	Yes	Yes	No	Yes	Yes	Yes
Blockless	Yes	Yes	Yes	Yes	Yes	No	Yes

(b)

Comm.	$O(c \cdot \log n)$	$O(\log n)$	$O(c)$	$O(1)$	$O(c)$	$O(c \cdot \log n)$	$O(c)$
CSP comp.	$O(c \cdot \log n)$		$O(c)$	$O(n)$	$O(1)$	$O(c \cdot \log n)$	$O(c)$
Audit comp.	$O(c \cdot \log n)$		$O(c)$	$O(n)$	$O(1)$	$O(c \cdot \log n)$	$O(c)$

decentralized and dependable cloud data source protection architecture. The architecture used tamper-proof blockchain records and embedded data provenance in blockchain transactions, with auditors verifying the data's origins based on the information in the blocks. Paying the blockchain miners, on the other hand, would be prohibitively expensive for cloud customers. To address the problem of unreliability in traditional verification procedures, Yue et al. [22, 24] presented a blockchain-based P2P cloud storage data integrity verification methodology. The approach used the Merkle-tree to verify data integrity and examined system performance using various Merkle-tree architectures. Wang et al. [25] proposed a decentralized architecture to tackle the traditional paradigm's single-point trust problem through communal trust. The architecture built a public protocol that maintains the data state under public scrutiny and prevents storage parties from engaging in fraudulent activities. For large-scale IoT data, Wang et al. [11] developed a blockchain-based data integrity verification system. They constructed a prototype system of edge computing processors near IoT devices to preprocess large-scale IoT data and performed data integrity verification in the form of transactions. None of the aforementioned approaches provide formal proof of security, and the security of integrity verification in a blockchain network setting remains an open question. We compared our scheme with the state-of-art, as shown in Table 1.

3. Preliminaries

3.1. Smart Contract. A blockchain is a distributed database that uses encryption, hashing, timestamping, consensus mechanisms, and other techniques [26]. All operations (transactions) are recorded on the blockchain, which is a chained data structure with tamper-proof features. A smart contract is a blockchain-based event-driven program [27]. It is contained within a virtual node that allows automated script execution and data processing in response to event triggers. Smart contracts, like transactions on the blockchain, offer distributed storage and tamper-proof characteristics. Being different from traditional executable programs, smart contracts are distributed and run according to preset rules that create communication protocols between communicating parties [28]. As a consequence, smart contracts enable traceable and irreversible activities without the involvement of a third party.

3.2. ZSS Signature. Let g be the generator of the group $\mathbb{G}$ which is a cyclic additive group with the large prime order p. Allow $\mathbb{G}_T$ to be a cyclic multiplicative cyclic group of order p. Let $e : \mathbb{G} \times \mathbb{G} \longrightarrow \mathbb{G}_T$ be a bilinear pairing if it satisfies the following properties:

(1) Bilinear: $\forall P, Q \in \mathbb{G}$, and $a, b \in \mathbb{Z}_p$, the equation $e(aP, bQ) = e(P, Q)^{ab}$ holds

(2) Computability: $\forall P, Q \in \mathbb{G}$, there is an effective algorithm to calculate $e(P, Q)$

(3) Nondegenerate: $\exists P, Q \in \mathbb{G}$, such that $e(P, Q) \neq 1$, which means that the map does not send all pairs in $\mathbb{G} \times \mathbb{G}$ to the identity in $\mathbb{G}_T$. The ZSS signature [29] includes three algorithms: *KeyGen*, *Sign*, and *Verify*. Let $H : \{0,1\}^* \longrightarrow \{0,1\}^\lambda$ be a secure hash function.

(i) *KeyGen*. Randomly select an integer $\alpha \longleftarrow \mathbb{Z}_p^*$, and compute αg. The private key is $sk = \alpha$, and the public key is $pk = \alpha g$

(ii) *Sign*. Given a message $m \in \{0,1\}^*$, the signature is $Sig = \frac{1}{H(m) + \alpha} g$

(iii) *Verify*. Given a signature *Sig*, a public key *pk*, and a message m, compute $H(m)$, and verify the equation:

$$e(g, g) = e(H(m)g + pk, Sig). \quad (1)$$

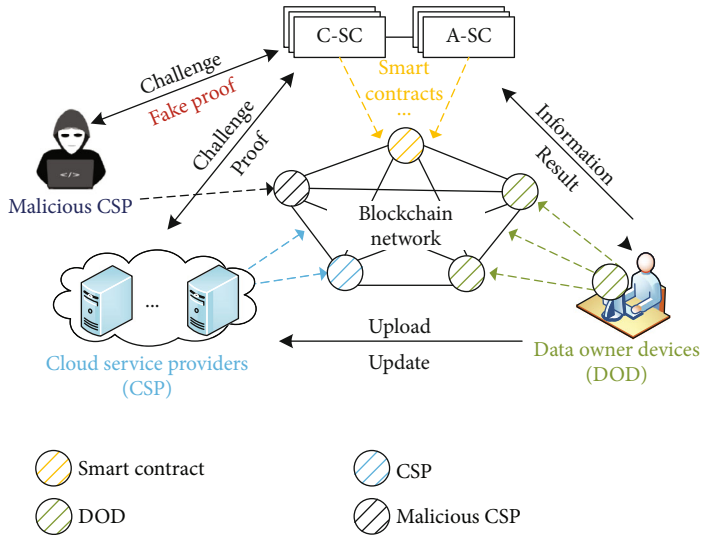


Figure 1: Network model.

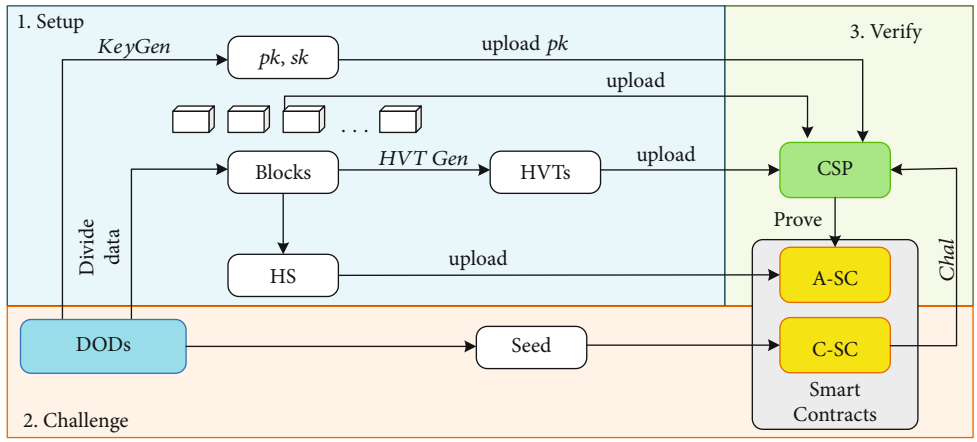


Figure 2: Verification protocol.

If the equation holds, the signature is valid; otherwise, the signature is invalid.

4. Approach Overview

4.1. Network and Threat Model. Figure 1 depicts the SBB-IV scheme's network model, which consists of three entities: data owner devices (DODs), cloud service providers (CSPs), and smart contracts.

(1) DODs: DODs act as nodes on the blockchain network, outsourcing users' data to CSPs and paying for the execution with smart contracts

(2) CSPs: Data storage and maintenance services are provided by CSPs, which are connected to the blockchain network as nodes

(3) Smart contracts: Smart contracts are virtual nodes that contain automated scripts. They cannot be destroyed or modified by any enemy

DODs outsource users' data to CSPs and pay for the execution through smart contracts on the blockchain network. Smart contracts issue a challenge to audit cloud data integrity. Based on the proof created by CSPs, smart contracts use the verification algorithm to check the proof's validity and deliver the outcomes to DODs. Finally, the blockchain keeps track of everything. The TPA collusive attack is avoided in this process because smart contracts are automated execution scripts. As a result, only the threat model described below is considered in this paper.

4.2. Malicious CSP. The malicious CSP knows the data and the public information; the purpose of the malicious CSP is to cheat smart contracts. That is, the malicious CSP owns the knowledge $<Data, public\ information>$ and wants to find fake proof to pass the verification of smart contracts.

4.3. Protocol. The SBB-IV scheme is a collection of five polynomial-time algorithms: $KenGen$, $HVTGen$, $Challenge$, $Response$, and $VerifyProof$ (see detailed algorithm in Section 5-B). Based on the scheme, we create an integrity

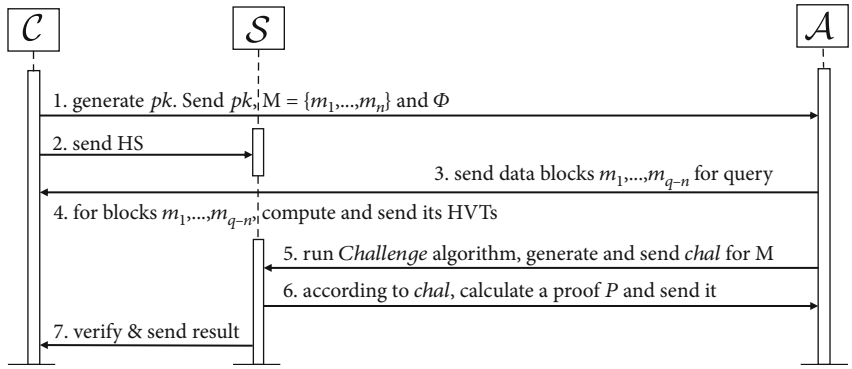


Figure 3: Blockchain-based security model.

verification protocol. Setup, Challenge, and Verify are the three stages of the protocol, as shown in Figure 2.

In the Setup stage, DOD uses the algorithm $(pk, sk) \longleftarrow KeyGen(1^{\kappa})$ to create the pair of keys. It runs the algorithm $(\mathbf{\Phi}, \mathbf{HS}) \longleftarrow HVTGen(sk, \mathbf{M})$ to compute a HVTs sequence and a hash sequence, then uploads the blocks $\mathbf{M} = \{m_1, m_2, \cdots, m_n\}$ and the HVTs sequence $\mathbf{\Phi}$ to CSP, and saves the hash sequence **HS** to audit smart contracts (A-SC).

In the Challenge stage, DOD sends a random *seed* to the challenge smart contract (C-SC). Then, using the algorithm $chal \longleftarrow Challenge(seed)$, C-SC produces a challenge and transmits it to CSP and A-SC.

In the Verify stage, CSP creates a proof using the procedure $P \longleftarrow Response(pk, chal, \mathbf{\Phi}, \mathbf{M})$ and delivers it to A-SC, according to *chal*. The proof is then verified by A-SC using the algorithm $VerifyProof(pk, chal, P, \mathbf{HS})$, and the result is sent to DOD.

4.4. Blockchain-Based Security Model. An interactive game between a challenger $\mathscr{C}$, a smart contract $\mathscr{S}$, and an adversary $\mathscr{A}$ defines the blockchain-based security model. In the Setup phase, we convey the challenged data **M** to the adversary to capture the semantic security of adaptive chosen message attack. As a result, in the Query phase, the adversary might adaptively pick multiple data blocks for the HVT query. The game is played in the following manner:

(i) Setup: $\mathscr{C}$ generates and sends a public key *pk* to $\mathscr{A}$. Then, $\mathscr{C}$ sends a data $\mathrm{M} = \{m_1, m_2, \cdots, m_n\}$ and its HVTs sequence Φ to $\mathscr{A}$. Finally, $\mathscr{C}$ sends the hash sequence HS to $\mathscr{S}$

(ii) Query: $\mathscr{A}$ adaptively makes queries; the selected $q - n$ blocks $m_1, \cdots, m_{q-n}$ is sent to $\mathscr{C}$. According to queries, $\mathscr{C}$ computes HVTs for all m_j, where $1 \le j \le (q - n)$, and returns the HVLs. Note that m_j can be a block of the data $M = \{m_1, m_2, \cdots, m_n\}$

(iii) Challenge: $\mathscr{S}$ generates *chal* for M by running the *Challenge* algorithm

(iv) Forge: According to the challenge *chal*, $\mathscr{A}$ calculates a proof P for M and sends it to $\mathscr{S}$

(v) Verify: $\mathscr{S}$ verifies the proof P by executing the algorithm $VerifyProof$. If P is valid, $\mathscr{A}$ wins the game

The game process can be referred to as Figure 3. The security definition of the scheme is as follows.

Definition 1. If any probabilistic polynomial-time adversary $\mathscr{A}$ cannot win the game with nonnegligible probability, the SBB-IV scheme is secured against the adaptive chosen messages attack when the integrity of the remote data **M** is violated.

5. Our Schemes

5.1. Notations. We employ a pseudorandom permutation function (PRP), $\pi : \{0,1\}^{log_2(n)} \times \{0,1\}^{\kappa} \longrightarrow \{0,1\}^{log_2(n)}$, and a pseudorandom function (PRF), $f : \{0,1\}^{*} \times \{0,1\}^{\kappa} \longrightarrow \mathbb{Z}_p$, in addition to the symbols specified in the preceding section. Aside from that, $H : \{0,1\}^{*} \longrightarrow \{0,1\}^{\lambda}$ is a secure generic hash function.

5.2. Scheme Detail. The SBB-IV scheme is described in full in this section:

(1) $KeyGen(1^{\kappa}) \longrightarrow (pk, sk)$. The algorithm selects a random number from the ring, $\alpha \longleftarrow \mathbb{Z}_p^*$, as a private key *sk*, and then computes $Y = \alpha g$ as a public key, according to the security parameter κ

(2) $HVTGen(sk, \mathbf{M}) \longrightarrow (\mathbf{\Phi}, \mathbf{HS})$. The algorithm firstly splits a data **M** into n equal length blocks; that is, $\mathbf{M} = \{m_1, m_2, \cdots, m_n\}$. Next, for $1 \le i \le n$, it calculates the hash value $H(m_i)$ for a block m_i and then computes the HVT as following equation:

$$\delta_i = \frac{1}{H(m_i) + m_i + \alpha} g. \tag{2}$$

Finally, it outputs a HVTs sequence, $\mathbf{\Phi} = \{\delta_1, \delta_2, \cdots, \delta_n\}$, and a hash sequence $\mathbf{HS} = \{H(m_1), H(m_2), \cdots, H(m_n)\}$.

(3) $Challenge(seed) \longrightarrow chal$. Based on the $seed = (c, \kappa)$, the algorithm chooses two random numbers (k_1, k_2), where $1 \le k_1 \le \kappa$, $1 \le k_2 \le \kappa$. For $1 \le j \le c$, it computes $i_j = \pi_{k_1}(j)$ and $v_j = f_{k_2}(j)$ by the PRR function and the PRF function. The output is $chal = \{(i_j, v_j)\}_{1<j<c}$

(4) $Response(pk, chal, \mathbf{\Phi}, \mathbf{M}) \longrightarrow P$. According to the challenge, the algorithm's calculation is as follows:

$$\begin{aligned} \theta &= \sum_{j=1}^{c} v_j Y, \\ u &= \sum_{j=1}^{c} v_j m_{i_j}, \\ \eta &= g - g^2 \sum_{j=1}^{c} \frac{v_j}{\delta_{i_j}}. \end{aligned} \tag{3}$$

Lastly, it outputs $P = \{\theta, u, \eta\}$.

(5) $VerifyProof(pk, chal, P, \mathbf{HS}) \longrightarrow \{1, 0\}$. The algorithm accepts pk, P, $chal = \{(i_j, v_j)\}_{1<j<c}$ and **HS** as input and then calculates:

$$\begin{aligned} \nu &= \sum_{j=1}^{c} v_j H\left(m_{i_j}\right) g, \\ \mu &= \nu + ug. \end{aligned} \tag{4}$$

Finally, the output is depending on whether the following equation holds:

$$e(\eta, g) \cdot e(\mu + \theta, g) = e(g, g). \tag{5}$$

If the equation holds, the algorithm outputs 1; otherwise, it outputs 0.

5.3. Dynamic. Because the HVT built in the scheme (as shown in Equation (2)) is based solely on the block and excludes a fixed numerical index, it can enable completely dynamic operations such as modification, insertion, and deletion. The technique generates a hash sequence that records the location of each block. As a consequence, the following procedure is used to update the data:

Step 1: DOD delivers an request to A-SC, $Request = (ope, pos, con)$, where *ope* represents the updated operations, *pos* denotes the updated position, and *con* represents the updated content. Note that when a delete operation is performed, *con* is empty

Step2: According to the request, A-SC performs the corresponding update operation and records the modification on the blockchain

Step3: When the blockchain is recorded successfully, DOD sends the *Request* to CSP to complete the update

5.4. Implementation. In our scheme, we encapsulate the *Challenge* algorithm and the *VerifyProof* algorithm into smart contracts to perform the task of auditing instead of TPA. Users can trigger the execution of smart contracts by sending *seed*. This *Seed* not only includes the number of audit blocks and security parameters, but users can also set parameters such as the audit cycle time and the number of audits performed according to their needs. The parameter c is the number of randomly sampled blocks in one audit. The parameter c is larger, the higher the audit confidence and the higher the computational overhead. Therefore, users set different c according to their needs to make a trade-off between different confidence levels and computation overhead. The tamper-proof nature of smart contracts eliminates the possibility of privacy leakage and collusion attacks. Because to the collision resistance and one-way nature of the hash function, an attacker cannot access the data through the hash value, despite the fact that we have put the hash sequence on the public smart contract.

6. Scheme Analysis

6.1. Correctness. The PRP and PRF functions in the *Challenge* algorithm of the SBB-IV scheme ensure that the blocks are randomly picked for each audit, making it impossible for a malicious CSP to prepare proofs ahead of time. If the remote data is preserved, the proof P generated by the *Response* algorithm will always pass the *VerifyProof* algorithm's verification. The scheme is correct in the following ways:

$$\begin{aligned} &e(\eta, g) \cdot e(\mu + \theta, g) \\ &\quad = e\left(\sum_{j=1}^{c} v_j H\left(m_{i_j}\right) \cdot g + \sum_{j=1}^{c} v_j m_{i_j} g + \sum_{i=1}^{c} v_j Y, g\right) \cdot \\ &e\left(g - g^2 \sum_{j=1}^{c} \frac{v_j}{\delta_{i_j}}, g\right) \\ &\quad = e(g, g) \cdot e\left(-g \sum_{j=1}^{c} v_j \left(H\left(m_{i_j}\right) + m_{i_j} + \alpha\right), g\right) \cdot \\ &e\left(g \sum_{j=1}^{c} v_j \left(H\left(m_{i_j}\right) + m_{i_j} + \alpha\right), g\right) \\ &\quad = e(g, g) \cdot e(g, g)^{-\sum_{j=1}^{c} v_j \left(H\left(m_{i_j}\right) + m_{i_j} + \alpha\right)} \\ &\qquad \cdot e(g, g)^{\sum_{j=1}^{c} v_j \left(H\left(m_{i_j}\right) + m_{i_j} + \alpha\right)} = e(g, g). \end{aligned} \tag{6}$$

6.2. Security. We treat the hash function $H(\cdot)$ as a random oracle and reduce the security of the SBB-IV scheme to the q-CAA problem [21].

Definition 2 (q-CAA problem). For an integer q, and $\alpha \in_R \mathbb{Z}_p$, $g \in \mathbb{G}$, given

$$\left\{ g, Y = \alpha g, \frac{1}{w_1 + \alpha} g, \cdots, \frac{1}{w_q + \alpha} g \right\}, \tag{7}$$

where $w_1, \cdots, w_q \in_R \mathbb{Z}_p$, to compute $(1/w + \alpha)g$ for some $w \notin \{w_1, \cdots, w_q\}$.

q-CAA assumption. The q-CAA problem is (t, ε)-hard if for a t-time adversary $\mathscr{A}$, the advantage of $\mathscr{A}$ to solve the problem is negligible:

$$Adv_{\mathscr{A}} = \Pr\left[\mathscr{A}\left(g, \alpha g, \frac{1}{w_1 + \alpha} g, \cdots, \frac{1}{w_q + \alpha} g \right) = \frac{1}{w + \alpha} g \right] \leq \varepsilon, \tag{8}$$

where ε is a negligible probability, and $w_1, \cdots, w_q \in_R Z_p$.

Theorem 3. *Suppose the (t, ε)-q-CAA assumption holds in the group $\mathbb{G}$, our scheme is (t, ε)-secure against adaptive chosen message attack under the random oracle model.*

Proof. If an adversary $\mathscr{A}$ can break the security of the SBB-IV scheme, we will show a challenger $\mathscr{C}$ how to use $\mathscr{A}$ to solve the q-CAA problem. The challenger $\mathscr{C}$ has known that $g \in \mathbb{G}, Y = \alpha g, w_1, w_2, \cdots, w_q$ and $\delta_1 = (1/w_1 + \alpha)g, \delta_2 = (1/w_2 + \alpha)g, \cdots, \delta_q = (1/w_q + \alpha)g$, and her goal is to calculate $(1/w + \alpha)g$ for some $w \notin \{w_1, w_2, \cdots, w_q\}$. Therefore, an interactive game between a challenger $\mathscr{C}$, a smart contract $\mathscr{S}$, and an adversary $\mathscr{A}$ as follows:

(i) Setup: The challenger $\mathscr{C}$ generates the public key pk, $Y = \alpha g$ and sends it to $\mathscr{A}$. Then, $\mathscr{C}$ selects a data $M = \{m_1, m_2, \cdots, m_n\}$ and constructs its HVTs sequence Φ and its hash sequence HS as follows. $\mathscr{C}$ maintains a list of tuples $<w_i, H_i, m_i>$. The list is initially empty. For a block m_i, $\mathscr{C}$ selects a $w_i \in \{w_1, \cdots, w_n\}$ and computes $H_i = w_i - m_i$; its HVT is $\delta_i = (1/w_i + \alpha)g = (1/H_i + m_i + \alpha)g$. Then $\mathscr{C}$ adds the tuple $<w_i, H_i, m_i>$ to the list and removes w_i from w-parameters $(w_1, \cdots, w_q)$. Finally, $\mathscr{C}$ sends the HVTs sequence to $\mathscr{A}$ and the hash sequence HS to $\mathscr{S}$

(ii) Query: The adversary $\mathscr{A}$ adaptively selects $q - n$ different blocks $m_1, m_2, \cdots, m_{q-n}$ and sends them to $\mathscr{C}$ for HVTs queries. Note that m_j $(1 \leq j \leq (q - n))$ can be a block of the data $M = \{m_1, m_2, \cdots, m_n\}$. At any time $\mathscr{A}$ can query hash value. When $\mathscr{A}$ queries at m_j, $\mathscr{C}$ responds as follows: [1)]

(1) Hash query. $\mathscr{C}$ firstly checks if the query m_j already exists in the list $<w_i, H_i, m_i>$. If so, $\mathscr{C}$ responds with H_i; otherwise, $\mathscr{C}$ randomly selects a w_j in the remaining w-parameters and responds with $H_j = w_j - m_j$ and adds the tuple $<w_j, H_j, m_j>$ to the list and removes w_j from w-parameters

(2) HVT query. $\mathscr{C}$ firstly checks if the query m_j already exists in the list $<w_i, H_i, m_i>$. If so, $\mathscr{C}$ responds with corresponding $\delta_j = (1/w_j + \alpha)g = (1/H_j + m_j + \alpha)g$; otherwise, $\mathscr{C}$ randomly selects a w_j in the remaining w-parameters and computes $H_j = w_j - m_j$ and responds with corresponding δ_j. Then, $\mathscr{C}$ adds the tuple $<w_j, H_j, m_j>$ to the list and removes w_j from w-parameters

(iii) Challenge: $\mathscr{S}$ generates *chal* for M by running the *Challenge* algorithm

(iv) Forge: According to the challenge *chal*, $\mathscr{A}$ calculates a proof P for M and delivers it to $\mathscr{S}$

(v) Verify: $\mathscr{S}$ verifies the proof P by executing the algorithm $VerifyProof$. If $VerifyProof$ outputs 1, $\mathscr{A}$ wins the game

When the block audited is corrupted, suppose that there is a block m_j corrupted and $\mathscr{A}$ can forge a fake proof that passes the verification with a nonnegligible probability.

We assume that the fake proof is $P^* = \{\theta^*, u^*, \eta^*\}$, where

$$\begin{aligned} \theta^* &= \sum_{i=1}^{c} v_i Y, \\ u^* &= \sum_{i=1, i \neq j}^{c} v_i m_i + v_j m_j^*, \\ \eta^* &= g - g^2 \sum_{i=1, i \neq j}^{c} \frac{v_i}{\delta_i} - g^2 \frac{v_j}{\delta_j^*}. \end{aligned} \tag{9}$$

When $\mathscr{S}$ verifies the proof P^* by executing the algorithm $VerifyProof$, it computes

$$\begin{aligned} \nu &= \sum_{i=1}^{c} v_i H(m_i) g, \\ \mu^* &= \nu + u^* g. \end{aligned} \tag{10}$$

Therefore, the process of verification is as follows:

$$\begin{aligned}
&e(\eta^*, g)\cdot e(\mu^*+\theta^*, g)\\
&= e\left(g\sum_{i=1}^{c} v_i H(m_i) + \sum_{i=1 i\neq j}^{c} v_i m_i g + v_j m_j^* g + \sum_{i=1}^{c} v_i Y, g\right)\\
&\quad \cdot e\left(g - g^2 \sum_{i=1 i\neq j}^{c} \frac{v_i}{\delta_i} - g^2 \frac{v_j}{\delta_j^*}, g\right)\\
&= e(g,g)\cdot e(g,g)^{-\sum_{i=1 j\neq j}^{c} v_i (H(m_i)+m_i+\alpha)} \cdot e(g,g)^{-g\left(v_j/\delta_j^*\right)}\\
&\quad \cdot e(g,g)^{\sum_{i=1 j\neq j}^{c} v_i (H(m_i)+m_i+\alpha)} \cdot e(g,g)^{v_j\left(H\left(m_j\right)+m_j^*+\alpha\right)}\\
&= e(g,g)\cdot e\left(-g^2\frac{v_j}{\delta_j^*}, g\right)\cdot e(g,g)^{v_j\left(H\left(m_j\right)+m_j^*+\alpha\right)}\\
&= e(g,g)\cdot e(g,g)^{-g\left(v_j/\delta_j^*\right)}\cdot e(g,g)^{v_j\left(H\left(m_j\right)+m_j^*+\alpha\right)}\\
&= e(g,g)\cdot e(g,g)^{-g\left(v_j/\delta_j^*\right)+v_j\left(H\left(m_j\right)+m_j^*+\alpha\right)}.
\end{aligned} \tag{11}$$

If the fake proof passes the verification, we get $e(\eta^*, g)\cdot e(\mu^*+\theta^*, g) = e(g,g)$. Hence, from the above derivation, we get the following equation:

$$e(g,g)\cdot e(g,g)^{-g\left(v_j/\delta_j^*\right)+v_j\left(H\left(m_j\right)+m_j^*+\alpha\right)} = e(g,g). \tag{12}$$

where $-g(v_j/\delta_j^*) + v_j(H(m_j) + m_j^* + \alpha) = 0$. That is, $g(1/\delta_j^*) = H(m_j) + m_j^* + \alpha$. As a result, we get $\delta_j^* = (1/H(m_j) + m_j^* + \alpha)g$. Since we have assumed that $m_j \neq m_j^*$, we will discuss it in two cases.

(i) Case 1: $H(m_j) + m_j^* = H(m_j) + m_j$.

In this case, we get $m_j^* = m_j$, which contradicts our hypothesis. Therefore, this case proves that the block m_j is not corrupted when the adversary $\mathcal{A}$ wins the game.

(ii) Case 2: $H(m_j) + m_j^* \neq H(m_j) + m_j$.

This case shows that when the adversary $\mathcal{A}$ finds a fake HVT δ_j^* with a nonnegligible probability in a time t, the challenger $\mathcal{C}$ finds a $(1/w + \alpha)g$ for some $w \notin \{w_1, \cdots, w_q\}$ with same nonnegligible probability in a time t, which means $\mathcal{C}$ breaks the q-CAA problem.

In summary, suppose the (t, ε)-q-CAA assumption holds in the group $\mathbb{G}$, our scheme is (t, ε)-secure against adaptive chosen message attack under the random oracle model. □

6.3. Scalability. In an IoT data storage system, with the continuous increase of IoT data, cloud storage needs to have scalability. In the SBB-IV scheme, we divide large data into smaller blocks, which is beneficial to the fine-grained control of the data and enhances the scalability of the cloud storage system. In the meanwhile, the proposed scheme is fully dynamic which means node devices can insert, modify, and delete uploaded data according to their needs. Furthermore, the scheme can be compatible with more systems without compromising efficiency, since HVTs are computed using general cryptographic hash functions rather than expensive elliptic curve hash functions [30]. As a result, the scheme is suitable for the integrity verification of large-scale IoT data.

In addition to IoT systems, our scheme can also be applied to a blockchain-based P2P (peer-to-peer) file system. In this system, an edge device is a peer node, and each peer node can become a client or server. Our scheme solves the bandwidth problem of sharing files from a central server to clients. Files can be shared through different nodes without requesting all files from a central server. At the same time, due to the homomorphism of HVTs, the speed of nodes verifying file integrity is greatly improved. Therefore, the SSB-IV scheme greatly improves the scalability and efficiency of file sharing

7. Evaluation

To justify the performance of the SBB-IV scheme, we conduct mathematical analysis and a series of experiments in this part. The pairing-based cryptography library (PBC, http://crypto.stanford.edu/pbc/) is used in our experiments. The experiments are implemented in the GoLang programming language and run on an Intel(R) Core(TM) i7-10700 CPU with 16 GB of RAM. The blockchain platform is Hyperledger Fabric 2.2.0. The security level has been set to 80 bits, implying that the $|p| = 160$. We set each block's size to 8 KB and produce 1000, 5000, 10000, 50000, and 100000 blocks for the test. We present the average values across these 10 trials throughout the examination.

7.1. Mathematical Analysis. We calculate the computation complexity of the SBB-IV scheme. Users execute the algorithms *KeyGen* and *HVTGen*; smart contracts run the algorithms *Challenge* and *VerifyProof*; CSPs runs the algorithm *Response*. The complexity of each algorithm is shown in Table 2.

(i) *KeyGen*: This algorithm performs only one multiplication

(ii) *HVTGen*: In this algorithm, since each block is needed to compute a HVT, the overhead of the algorithm is $O(n)$

(iii) *Challenge*: According to the parameter c, C-SC needs to use the PRF function and the PRP function to calculate two groups of random numbers. Hence, the computation overhead is $O(c)$

(iv) *Response*: This algorithm needs to calculate $\theta = \sum_{j=1}^{c} v_j Y$, $u = \sum_{j=1}^{c} v_j m_{i_j}$, and $\eta = g - g^2 \sum_{j=1}^{c} (v_j/\delta_{i_j})$. The computation cost is $O(c)$

Table 2: Complexity analysis.

(a)

Algorithm	Computation	Complexity
KeyGen	$Mult_{\mathbb{G}}^{1}$	$O(1)$
HVTGen	For $m_1, m_2, \cdots, m_n$ computes $\delta_1, \delta_2, \cdots, \delta_n$	$O(n)$
Challenge	$PRF^c + PRP^c$	$O(c)$
Response	$2Mult_{Z_p}^{c} + 2Add_{Z_p}^{c-1} + Mult_{\mathbb{G}}^{c+2} + Add_{\mathbb{G}}^{c}$	$O(c)$
VerifyProof	$Mult_{\mathbb{G}}^{c+2} + Add_{\mathbb{G}}^{c} + BM_{\mathbb{G}}^{3}$	$O(c)$

(b)

Phase	Communication	Complexity
Setup	User sends pk, **M**, **Φ**, **HS**	$O(n)$
Challenge	A-SC sends $chal = \{(i, v_i)\}_{s_1 \le i \le s_c}$	$O(c)$
Verify	CSP sends $P = \{\theta, u, \eta\}$	$O(1)$

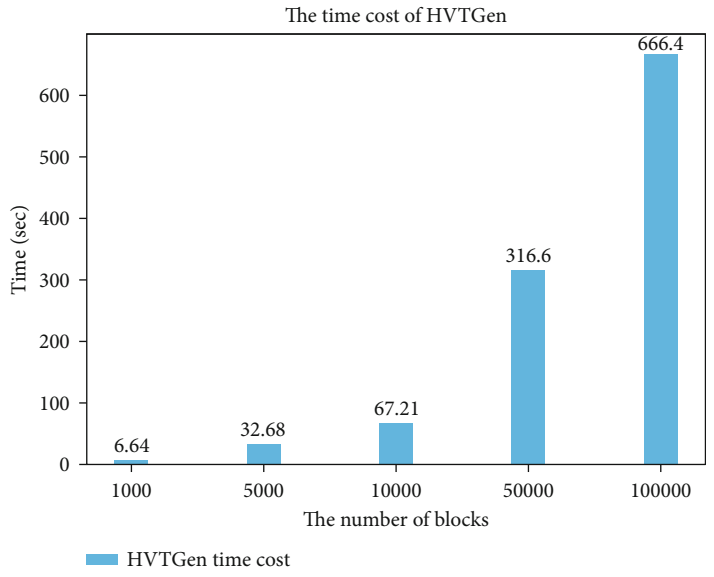


Figure 4: Comparison of computational overhead.

(v) *VerifyProof*: This algorithm needs to verify the equation $e(\eta, g) \cdot e(\mu + \theta, g) = e(g, g)$, where $\mu = \nu + ug$, $\nu = \sum_{j=1}^{c} v_j H(m_{i_j}) g$. The computation cost is $O(c)$

In the Setup phase, user sends pk, M, Φ, and HS, in which the communication complexity is $O(n)$. In the Challenge phase, C-SC sends challenge $chal = \{(i, v_i)\}_{s_1 \le i \le s_c}$ which is $2c|p|$ bits. In the Verify phase, CSP sends proof $P = \{\theta, u, \eta\}$ which is $3|p|$ bits. Therefore, the communication complexity of an audit is $O(c)$.

7.2. Experiments. In this section, we evaluate the actual performance of the scheme with a series of experiments.

7.2.1. Setup. In the Setup stage, the user's main computation overhead comes from the *HVTGen* algorithm. At the same time, the smart contract needs to store a hash sequence **HS** . In our experiments, we set the number of blocks to 1,000, 5,000, 10,000, 50,000, and 100,000, respectively. Because each block is 8 KB in size, 100,000 blocks represent 780 MB of data. As shown in Figure 4, the time consumption of the *HVTGen* algorithm grows linearly, but the algorithm only needs to be executed once. For 780 MB of data, the smart contract's storage consumption is only 15.04 MB, which is easily achievable for a distributed ledger (Figure 5).

7.2.2. Audit. As we discussed in Section 5, in the *Challenge* algorithm, the parameter c is larger, the higher the audit

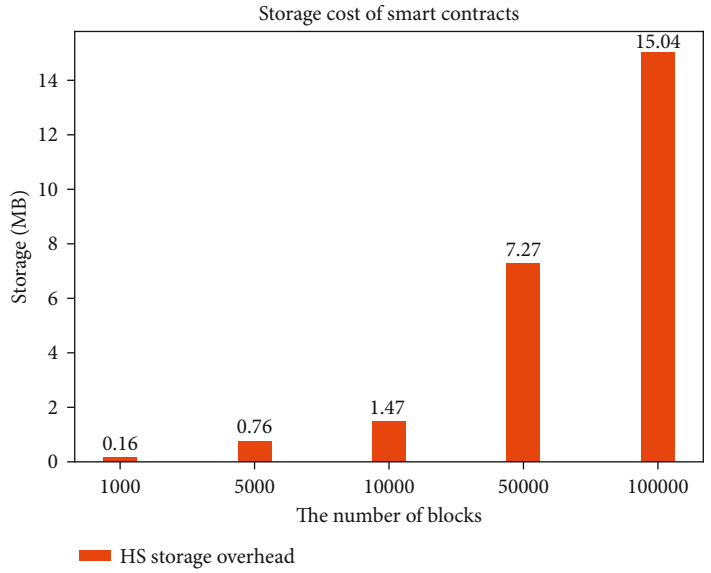


Figure 5: Storage overhead.

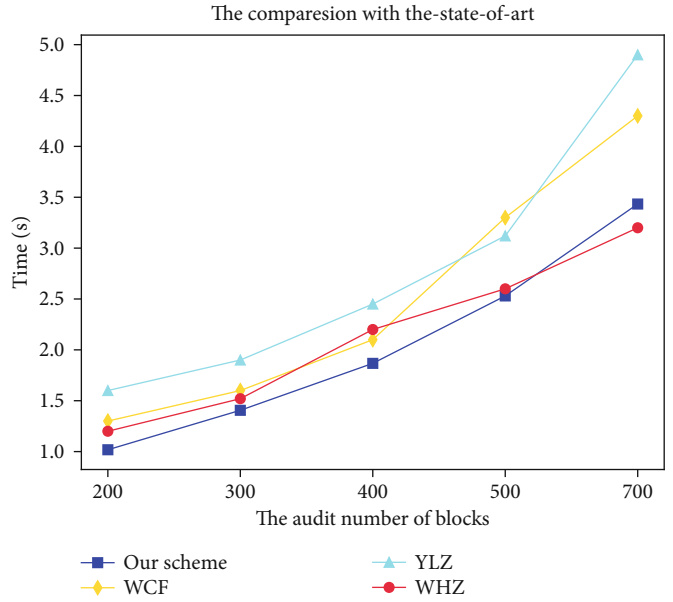


Figure 6: The time cost of an audit.

confidence, and the higher the computational overhead. Therefore, users set different c according to their needs to make a trade-off between different confidence levels and computation overhead.

To fully understand the time consumption of an audit, we locally tested the overall time consumption of three algorithms which include the *Challenge* algorithm, the *Response* algorithm, and the *VerifyProof* algorithm. We select other three blockchain-based schemes (YLZ-[22], WCF-[25], and WHZ-[11]) for comparison. In our experiments, we audit numbers c to 200, 300, 400, 500, and 700, respectively. The experimental results (as presented in Figure 6) show that our scheme has the lowest overall time consumption for one audit.

In addition, we test the time consumption of the *Response* algorithm running locally and the time consumption of the *VerifyProof* algorithm running in the encapsulated smart contract. Our blockchain platform uses Hyperledger Fabric 2.2.0, and we build a test network on a virtual machine (Ubuntu 20.04). Let P_x indicates a probability and t means the number of corrupted blocks, we get

$$P_x \geq 1 - \left(\frac{n-t}{n}\right)^c, \tag{13}$$

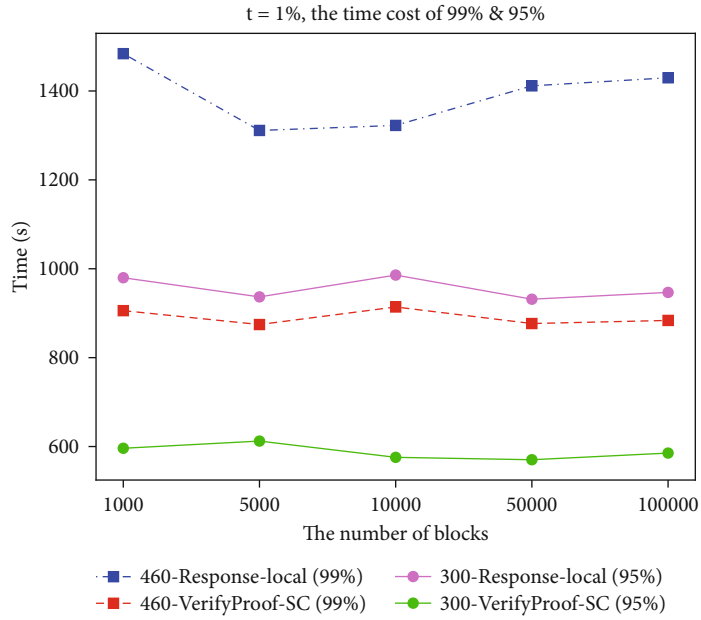


Figure 7: The time cost of Response and VerifyProof algorithms (99% and 95%).

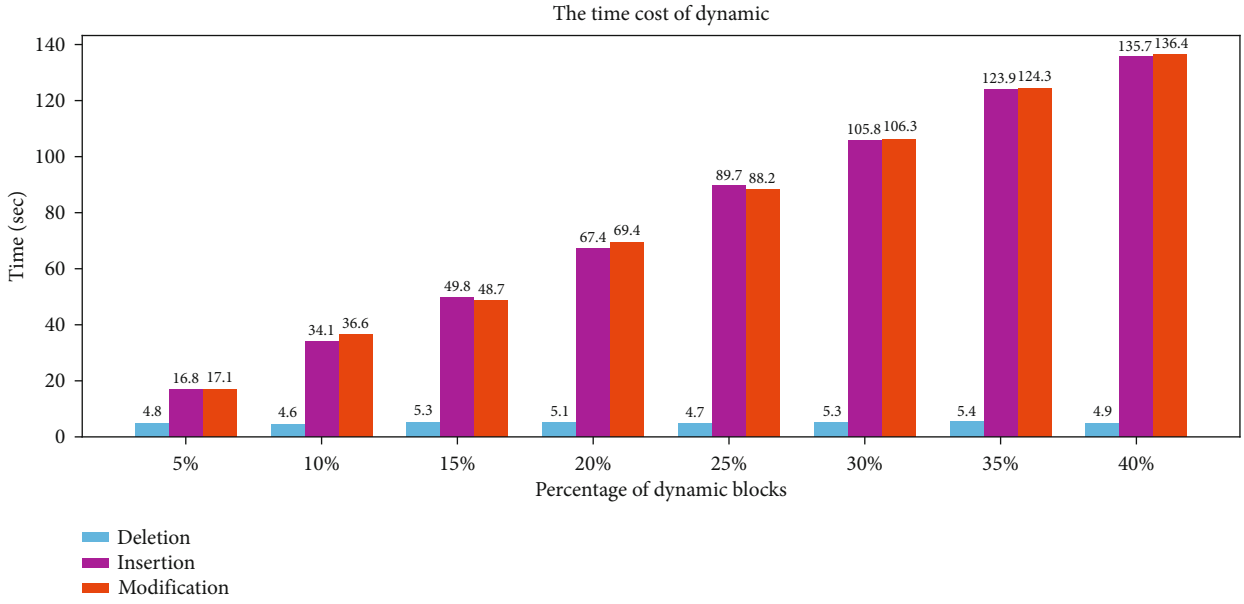


Figure 8: Time cost of dynamicoOperations.

Equation (13) shows that when t blocks are corrupted, different values of c will produce different confidence levels. Therefore, we assume that when 1% blocks corrupted, we set $c = 300$ and $c = 460$ to get 95% and 99% confidence, respectively, As presented in Figure 7, the time cost of *Response* and *VerifyProof* algorithm will remain even with the increase of the number of blocks.

7.2.3. Dynamic. For dynamic simulation tests, we use $n = 50000$. The time it takes to edit a block in our situation is determined by the time it takes for the blockchain to write the record and the time it takes to produce HVTs. We configured an endorsement node in our test network to write operation records to the Hyperledger (https://hyperledger-fabric.readthedocs.io/en/latest/index.html). The time consumption of insertion and modification is linear as the number of dynamic blocks rises, but the time consumption of deletion remains constant, as shown in Figure 8. Because deletion does not involve the creation of new HVTs, insertion and modification take longer than deletion. The time spent on deletion is primarily due to the time spent writing records by the endorsing node. Because the method for both operations is the same, insertion and modification take almost the same amount of time.

8. Conclusion

This paper mainly solves three problems, including the problem of TPA's privacy leakage and collusion attack, the problem of poor blockchain scalability, and the security problem of blockchain-based integrity verification schemes. To address the problems above, we propose a scalable blockchain-based integrity verification scheme that implements fully dynamic operations and blockless verification. The scheme builds scalable homomorphic verification tags based on ZSS short signatures. We exploit smart contract technology to replace TPA for integrity verification tasks, which not only eliminates the risk of privacy leakage but also resists collusion attacks. Furthermore, we formally define a blockchain-based security model that captures the semantic security of adaptive chosen message attacks. We show that our scheme is secure under the security assumption of cryptographic primitives. Finally, the mathematical analysis of our scheme shows that both the communication complexity and the communication complexity of an audit are $O(c)$, in which c is the number of challenge blocks. We compare our scheme with other schemes, and the results show that our scheme has the lowest time consumption to complete an audit.

Data Availability

The data that support the findings of this study are not openly available due to reasons of sensitivity and are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Key R&D Program of Zhejiang Province (Grant No. 2022C01055), the Key R&D Program of Zhejiang Province (Grant No. 2020C05005), the Hangzhou Innovation Institute, Beihang University, under Grant 2020-Y5-A-022, and the Beijing Natural Science Foundation (No.4202036). An earlier version of this paper has been presented at conference in 2021 IEEE SmartWorld Ubiquitous Intelligence and Computing Advanced and Trusted Computing Scalable Computing and Communications Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/IOP/SCI) .

References

[1] Z. Cai, X. Zheng, and J. Yu, "A differential-private framework for urban traffic flows estimation via taxi companies," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 6492–6499, 2019.

[2] C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring data storage security in cloud computing," *2009 17th International Workshop on Quality of Service*, pp. 1–9, 2009.

[3] Y. Deswarte, J. J. Quisquater, and A. Saïdane, "Remote integrity checking," in *Working conference on integrity and internal control in information systems*, pp. 1–11, Boston, MA, 2004.

[4] G. Ateniese, R. Burns, R. Curtmola et al., "Remote data checking using provable data possession," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 1, pp. 1–34, 2011.

[5] X. Luo, Z. Zhou, L. Zhong, J. Mao, and C. Chen, "An effective integrity verification scheme of cloud data based on bls signature," *Security and Communication Networks*, vol. 2018, 11 pages, 2018.

[6] M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan, "Auditing to keep online storage services honest," *HotOS*, 2007.

[7] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in *IEEE infocom*, vol. 2010, pp. 1–9, IEEE, 2010.

[8] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 5, pp. 847–859, 2011.

[9] S. G. Worku, C. Xu, J. Zhao, and X. He, "Secure and efficient privacy-preserving public auditing scheme for cloud storage," *Computers & Electrical Engineering*, vol. 40, no. 5, pp. 1703–1713, 2014.

[10] B. Liu, X. L. Yu, S. Chen, X. Xu, and L. Zhu, "Blockchain based data integrity service framework for iot data," in *2017 IEEE International Conference on Web Services (ICWS)*, pp. 468–475, Honolulu, HI, USA, 2017.

[11] H. Wang and J. Zhang, "Blockchain based data integrity verification for large-scale iot data," *IEEE Access*, vol. 7, pp. 164996–165006, 2019.

[12] G. Ateniese, R. Burns, R. Curtmola et al., "Provable data possession at untrusted stores," in *Proceedings of the 14th ACM conference on Computer and communications security*, pp. 598–609, 2007.

[13] A. Juels and B. S. Kaliski Jr., "Pors: proofs of retrievability for large files," in *Proceedings of the 14th ACM conference on Computer and communications security*, pp. 584–597, 2007.

[14] H. Shacham and B. Waters, "Compact proofs of retrievability," in *International conference on the theory and application of cryptology and information security*, pp. 90–107, Berlin, Heidelberg, 2008.

[15] C. C. Erway, A. Küpçü, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," *ACM Transactions on Information and System Security (TISSEC)*, vol. 17, no. 4, pp. 1–29, 2015.

[16] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective datasanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[17] M. A. Shah, R. Swaminathan, and M. Baker, "Privacypreserving audit and extraction of digital contents," *pasos revista de turismo y patrimonio cultural*, 2008.

[18] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial Networks," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–38, 2021.

[19] Z. Cai and Z. He, "Trading private range counting over big iot data," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 144–153, Dallas, TX, USA, 2019.

[20] S. Mitsunari, R. Sakai, and M. Kasahara, "A new traitor tracing," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 85, no. 2, pp. 481–484, 2002.

[21] Z. Hao, S. Zhong, and N. Yu, "A privacy-preserving remote data integrity checking protocol with data dynamics and public verifiability," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 9, pp. 1432–1437, 2011.

[22] D. Yue, R. Li, Y. Zhang, W. Tian, and C. Peng, "Blockchain based data integrity verification in p2p cloud storage," in *2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 561–568, Singapore, 2018.

[23] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla, "Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability," in *2017 17th IEEE/ACM international symposium on cluster, cloud and grid computing (CCGRID)*, pp. 468–477, Madrid, Spain, 2017.

[24] X. Zheng and Z. Cai, "Privacy-Preserved data sharing towards multiple parties in industrial iots," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[25] C. Wang, S. Chen, Z. Feng, Y. Jiang, and X. Xue, "Block chain-based data audit and access control mechanism in service collaboration," in *2019 IEEE International Conference on Web Services (ICWS)*, pp. 214–218, Milan, Italy, 2019.

[26] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," *Decentralized Business Review*, 2008.

[27] M. Ali, J. Nelson, R. Shea, and M. J. Freedman, "Blockstack: A global naming and storage system secured by blockchains," 2016.

[28] R. Almadhoun, M. Kadadha, M. Alhemeiri, M. Alshehhi, and K. Salah, "A user authentication scheme of iot devices using blockchain-enabled fog nodes," in *2018 IEEE/ACS 15th international conference on computer systems and applications (AICCSA)*, pp. 1–8, Aqaba, Jordan, 2018.

[29] F. Zhang, R. Safavi-Naini, and W. Susilo, "An efficient signature scheme from bilinear pairings and its applications," in *International Workshop on Public Key Cryptography*, pp. 277–290, Springer, 2004.

[30] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in *International conference on the theory and application of cryptology and information security*, pp. 514–532, Berlin, Heidelberg, 2001.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 8045535, 11 pages
https://doi.org/10.1155/2022/8045535

Research Article

Few-Shot Multihop Question Answering over Knowledge Base

Meihao Fan,[1] **Lei Zhang**,[2] **Siyao Xiao**,[1] **and Yuru Liang**[3]

[1]*School of Information Science and Engineering, Chongqing Jiaotong University, Chongqing 400074, China*
[2]*School of Mathematics and Statistics, Chongqing Jiaotong University, Chongqing 400074, China*
[3]*School of Economics and Management, Chongqing Normal University, Chongqing 400074, China*

Correspondence should be addressed to Lei Zhang; zhangleicqjtu@163.com

Received 20 December 2021; Accepted 16 March 2022; Published 6 May 2022

Academic Editor: Yingjie Wang

KBQA is a task that requires to answer questions by using semantic structured information in knowledge base. Previous work in this area has been restricted due to the lack of large semantic parsing dataset and the exponential growth of searching space with the increasing hops of relation paths. In this paper, we propose an efficient pipeline method equipped with a pretrained language model. By adopting beam search algorithm, the searching space will not be restricted in subgraph of 3 hops. Besides, we propose a data generation strategy, which enables our model to generalize well from few training samples. We evaluate our model on an open-domain complex Chinese question answering task CCKS2019 and achieve F1-score of 62.55% on the test dataset. In addition, in order to test the few-shot learning capability of our model, we randomly select 10% of the primary data to train our model, and the result shows that our model can still achieves F1-score of 58.54%, which verifies the capability of our model to process KBQA task and the advantage in few-shot learning.

1. Introduction

Due to the proliferation of artificial intelligence (AI), smart systems have made significant achievements in communication and information extraction [1–8]. Since a sophisticated smart system can bring much convenience and efficiency, the research in this field has attracted extensive attention from academic and industrial circles.

A KBQA system aims to answer questions (QA) by understanding the semantic structure and extract the answers in large knowledge base (KB).Recently, tremendous KBQA models are proposed to effectively utilize KB to answer "simple" questions. Here, "simple" refers to questions that can be answered with a single predicate or a predicate sequence in the KB. For instance, "Who directed Avatar?" is a simple question due to its answer can be obtained by a single triplet fact query (?, director_of, Avatar). To answer such questions, plenty of rule-based [9], keyword-based [10], and synonym-based methods [11–14] have been proposed. However, questions in real life are usually more complex which can only be answered correctly by a multihop query path with constraints. As is shown in Figure 1, for answering a complex question, a sequence of operations needs to be generated, including multihop query and answers combination. Recently, the use of KB to answer such complex questions (KBCQA) has attracted growing interests prodigiously [15]. Previous state-of-art KBCQA models can be categorized into a taxonomy that contains two main branches, namely, information retrieval-based (IR-based) and neural semantic parsing-based (SP-based) model. The IR-based model first recognizes topic entities in the natural language and links them to node entities in knowledge base [16–19]. Then, all nodes surrounding around the topic nodes are regarded as candidate answers, and a score function is used to model their semantic relevance and predict the final answers. Methods based on

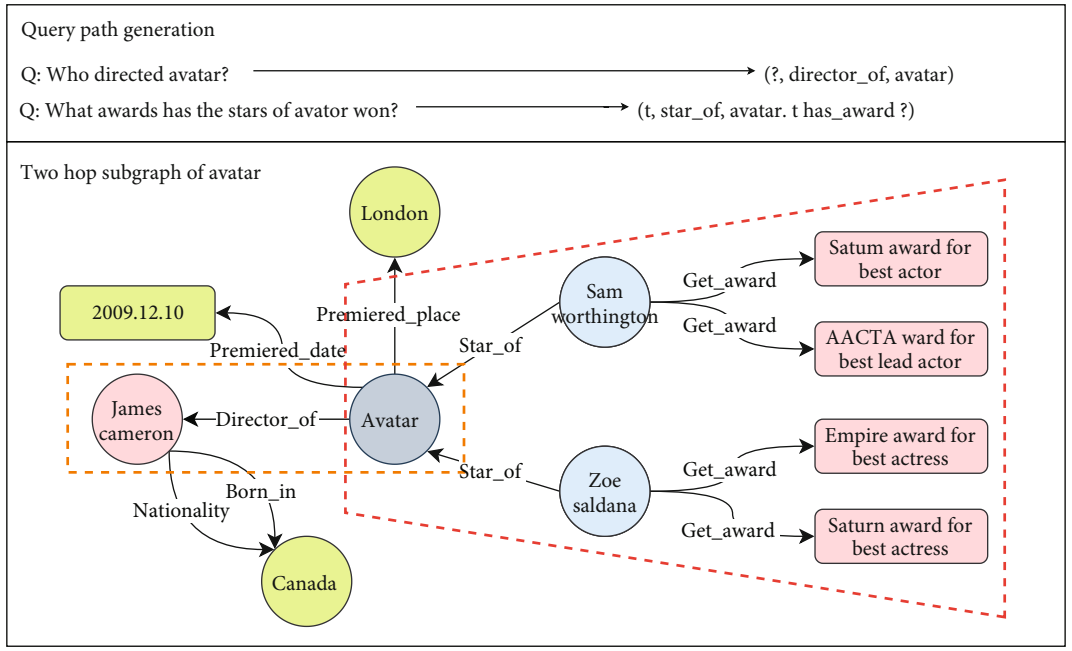


Figure 1: The subgraph of KB and two query paths.

semantic parsing usually includes a Seq2Seq module which converts natural languages into executable query languages and an executor module which executes the generated logical sequence on KB to obtain the final answers [20–24].

However, although the state-of-art models have made great achievements, several challenges still exist. Firstly, the dependency of annotated data is a thorny problem for SP-based models, which is usually settled by using a breadth-first search (BFS) to produce pseudo-gold action sequences and adopting the reinforce learning (RL) algorithm [25, 26]. Yet since BFS will inevitably ignore many other plausible annotations and RL usually suffers from several challenges, such as sparse reward and data inefficiency, the research of SP-based models is immensely hindered. Secondly, both IR-based and SP-based methods suffer from the large searching space. For better performance on KBCQA task, large KBs, such as Wikidata or FreeBase, are usually needed [27, 28]. Although these KBs contain comprehensive knowledge, they also bring vast search space when searching a query path with more than 3 hops. We record the average number of relations in one-hop and multihop subgraphs of a topic entity in our training dataset. It is shown that in one-hop subgraphs, the average number is 515, while in 2-hop and 3-hop subgraphs, it grows to 1920 and 6408, respectively. This exponential growth of generated candidate tuples makes it expensive and difficult for calculation. Thirdly, most previous work requires large KBCQA datasets to train their model, such as complex web questions and QALD [29, 30]. However, these large datasets are usually in English, hindering research in more realistic settings and in languages other than English.

To solve the three problems above, we propose a template-based model consisting of question classification, named entity recognition, query path generation, and path ranking module. Our contribution can be categorized into three fields:

(1) We propose a data-efficient model equipped with a pretrained language model BERT which can achieve high performance but only use tiny amount of data. Thus, our model can be utilized to process KBQA task in some languages without large KBQA datasets

(2) By adopting beam search algorithm and using ERNIE [31] to score for each searching branch, the spatial complexity and time complexity have been greatly dropped, but the generating accuracy still remains competitive

(3) We put forward a method to construct artificial data on predefined schemas of query graphs, allowing our model to process questions with novel categories which are excluded by training set

With the utilize of pretrained language model BERT and predefined schemas of query graphs, our model can effectively extract and filter the query tuples for a complex question. Also, we adopt beam search algorithm to relieve the exponential growth with increasing hops, which make it possible to handle multihop questions.

This paper is organized as follows: In Section 2, we review works on NER and beam search, which are the basis of our experiments. In Section 3 we present the overall architecture and then introduce each key component in detail. In Section 3, we demonstrate the evaluated models and the methodology used to generate the sentence embeddings. In Section 4, we describe the experimental setup and evaluation of the proposed model. Finally, we summarize the contribution of this work in the Section 5.

2. Related Work

Recently, with the rapid development and increasing attention of deep learning, the research on natural language processing has made great process. Especially when supported by emerging word embedding technologies and pretrained language models, the effectiveness of knowledge base question answering has been greatly improved. In this section, we will introduce some previous work related to the submodules of our model including named entity recognition (NER) and beam search algorithm. Besides, some few-shot KBQA models and a template-based model will also be introduced.

Named entity recognition is a key component in NLP systems for question answering, information retrieval, and relation extraction. Early NER models are mainly based on unsupervised and bootstrapped systems [32, 33] or feature-engineering supervised task [34, 35]. Nowadays, researchers tend to use neural network for NER task. NER is often solved as a sequence labeling problem by using the conditional random field (CRF) which requires a set of predefined features. Recently, some effective neural network approaches, especially for bidirectional long short-term memory, significantly improve the performance of CRF for NER task. Huang et al. use two LSTMs to capture past features and future features in sequence tagging task [36]. Then, a CRF layer is used to efficiently grasp the sentence level tag information of the sentence. The BiLSTM CRF is usually employed as the cornerstone of many subsequent improved NER models. BERT BiLSTM CRF uses BERT to embed extract rich semantic features into vectors and sends them to the BiLSTM CRF [37]. This model has achieved state-of-art performance in many NER tasks [38].

Beam search is a common heuristic algorithm for decoding structured predictors. When generating query paths for complex multihop questions, we need to consider longer relation path in order to reach the correct answers. However, the search space grows exponentially with the length of relation paths, bringing expensiveness for calculation and storage. The core idea of beam search is to use a score function to keep Top-K candidate relations instead of considering all relations when extending a relation path. Thus, the definition of score function determines the performance of Bean Search. Chen et al. (2019) proposed to keep only the best matching relation with a path ranking module that considers features extracted from topic entities and semantic information of the generated query paths [20]. Lan et al. (2019) also keep only one candidate relation using a traditional Siamese architecture where both the question and the candidate paths are each separately encoded into a single vector before the two vectors are matched [39]. The experimental results of these two models show little performance dropped but with significant reduction in spatial complexity and time complexity.

Since the expensiveness of constructing the annotated datasets, several works have been focused on few-shot learning for KBQA task. Chada et al. (2021) proposed a simple fine-tuning framework that regards the query path generation as a text-to-text task [40]. By leveraging a pretrained sequence-to-sequence models, their method outperforms many state-of-art models with an average margin of 34.2 F1 points on various few-shot settings of multiple QA benchmarks. Hua et al.(2020) proposed a semantic parsing based method using BFS to find the pseudo-gold annotation of a question and learning a reinforcement learning (RL) policy to generate a query sequence for obtaining the final answer [41].

Our model is most inspired by a template-based Chinese KBCQA model proposed by Wang et al. [42]. They use a pipeline method including a NER module, a query path generation module, and candidate tuple ranking module and process the question step by step. In NER module, they attach the BiLSTM CRF layer with a BERT layer to better understand the semantic information in the question, which gets quite high accuracy in topic entities recognition. Then, they extend one or two relations from the topic entity to generate the query paths and adopt bridging technology to process questions with multiple entities. Finally, a candidate query path ranking module is carefully designed to select the final query path. The differences between their work and our model are that we process the one-entity and multientity questions separately with a question classification module and predefine a set of query schema to restrict the searching space. On the predefined query pattern, we use a strategy to construct artificial questions which improve the ability of the classification model for few-shot learning. Moreover, we adopt beam search algorithm when generating query paths, which helps us achieve comparable performance but only using 10% resource of calculation and storage.

3. Our Method

In this section, we will present the overall architecture (shown in Figure 2) and then introduce each key component of the proposed model in detail.

3.1. Method Overview. The general idea behind our method is to process the question step by step. Given a question, we first encode it with a BERT layer, and then, the representations will be passed to an entity linking module (Section 3.2) of BERT-BiLSTM-CRF layer and a question classification module (Section 3.3) trained with extra manually constructed samples (Section 3.5). With the recognized topic entities and a specific category the question belongs to, we can refer to a more precise schema (Section 3.4) to generate the query path in a narrower searching space. However, since the query graph of a complex question may involve multiple relations, such simple generating program will bring intolerable time complexity and spatial complexity and bring calculating burden to the candidate tuple ranking module. To solve this, we adopt a heuristic algorithm for graph search (Section 3.6) based on a pretrained text-match model, which greatly decreases the number of candidate query paths. Afterwards, a candidate tuple ranking module is designed to sift out the final path using the above PTM-TextMatch model. By executing the golden query tuple, we can retrieve the answer in knowledge base.

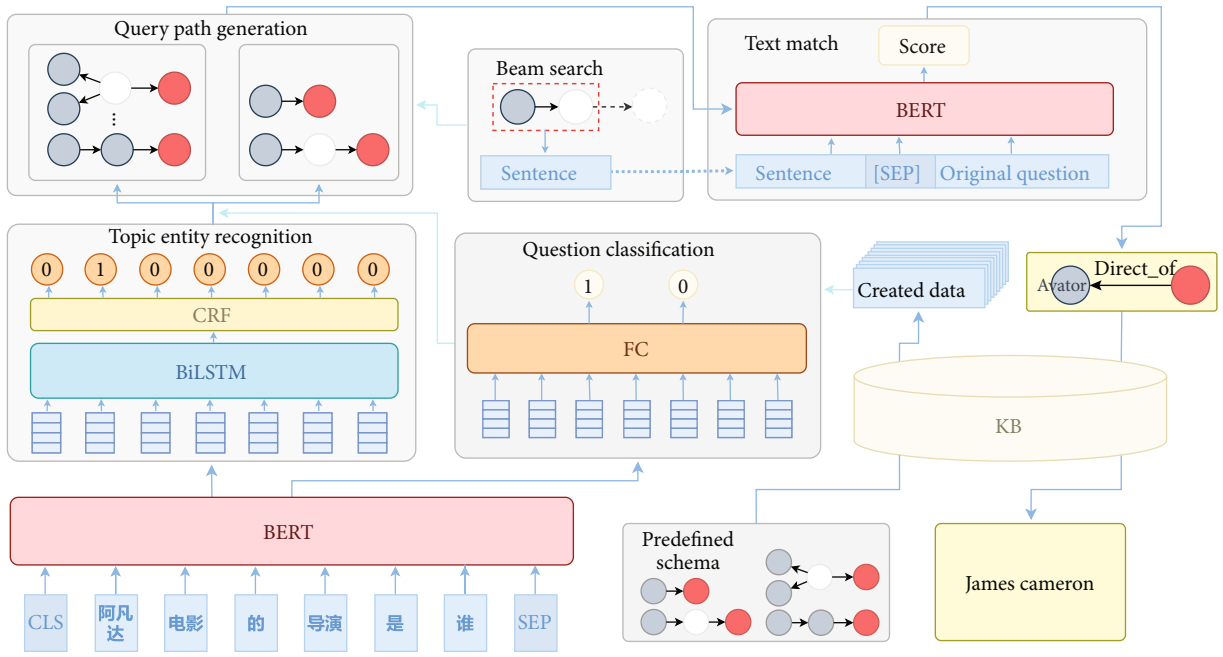


Figure 2: Basic framework of our model.

Besides, we are not to search aimlessly in KB when generating query subgraph. Instead, we refer to a set of predefined schemas of all possible query graphs in complex question answering. This policy will not only narrow the searching space significantly but also provide a semantic framework for reference when constructing artificial questions.

3.2. Node Extractor. The main goal of this module is to identify topic entities in the question. This module includes tokenization with dictionaries, named entity recognition (NER), and entity linking.

3.2.1. Tokenize. Different from English tokenize, Chinese tokenizing usually uses dictionaries as a supplementary to tokenize Chinese question text into Chinese words. In this paper, we use a dictionary provided by CCKS consisting of all subjects in KB, all entities, and their mentions in mention dictionary.

3.2.2. Named Entity Recognition. In the NER module, we encode the question with BERT layer and then pass it through a BiLSTM to capture the information of context and a CRF layer to predict label of each token. Let us use $Q = (t_1, t_1, t_1, \cdots, t_n)$ to represent a tokenized question. We put Q into a BERT layer to encode representations with semantic knowledge. Next, the representations $X_{i=1}^{|Q|}$ are passed through a BiLSTM layer and CRF layer [28].

For each input token, the context information is captured by two LSTMs, where one capture information from left to right and the other from right to left. At each time step t, a hidden vector $\overrightarrow{h_t}$ (from left to right) is computed based on the previous hidden state $\overrightarrow{h_{t-1}}$ and the input at the current step x_t. Then, the forward and backward context representations, generated by $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$, are concatenated into a long vector which we represent as $h_t = [\overrightarrow{h_t} : \overleftarrow{h_t}]$. The basic LSTM function is defined as follows:

$$\begin{bmatrix} \widetilde{c_t} \\ f_t \\ o_t \\ i_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} \left(W^T \begin{bmatrix} x_t \\ h_{t-1} \end{bmatrix} + b \right), \tag{1}$$

$$c_t = i_t \odot \widetilde{c_t} + f_t \odot c_{t-1},$$

$$h_t = o_t \odot \tanh(c_t),$$

where W^T and b are trainable parameters; $\sigma()$is the sigmoid function; i_t, o_t, and f_t indicate input, output, and forget gates, respectively; $\odot$ represents the dot product function; and x_t is the input vector of the current time step.

The output vectors of the BiLSTM contain the bidirectional relation information of the words in a question. Then, we adopt CRF to predict labels for each word, considering the dependencies of adjacent labels. The CRF is the Markov random field of Y given a random variable X condition and included an undirected graph G, where Y are connected by undirected edges indicating dependencies. Formally, given the observation variables $H = h_{i=1}^{|Q|}$, and a set of output values $y \in \{0, 1\}$, where $y = 1$ means, the corresponding token is a topic entity, and $y = 0$ is not. CRF defines potential function as

$$p(y|h) = \frac{1}{Z_h} \prod_{s \in S(y,h)} \phi_s(y_s, h_s), \tag{2}$$

where Z_h is a normalization factor overall output values, $S(y, h)$ is the set of cliques of G, and $\phi_s(y_s, h_s)$ is the clique potential on clique s.

Afterwards, in the BiLSTM-CRF model, a softmax over all possible tag sequences yields a probability for the sequency. The prediction of the output sequence is computed as follows:

$$y_* = \arg \max_{y \in \{0,1\}} \sigma(H, y),$$
$$\sigma(H, y) = \sum_{i=0}^{n} A_{y_i, y_{i+1}} + \sum_{i=0}^{n} P_{i, y_i}, \tag{3}$$

where A is a matrix of transition scores, $A_{y_i, y_{i+1}}$ represents the score of a transition from the tag y_i to y_{i+1}, n is the length of a sentence, P is the matrix of scores output by the BiLSTM network, and P_{i, y_i} is the score of the y_i^{th} tag of the i^{th} word in a sentence.

3.2.3. Entity Linking. In this module, we link the recognized named entity to the entity in KB and select a set of candidate topic entities with a mention dictionary. The mention dictionary is provided by CCKS sponsors describing mapping relations from mentions to node entities. After obtaining mentions of entities in a question, we correspond them to relevant node entities. Then, we need to extract helpful features from the mentions and entities to select the potential candidate entities. In this work, we extract six features as follows: the length of entity mention (f_1), the TF value of entity mention (f_2), the distance between the entity mention and interrogative word (f_3), word overlap between question and triplet paths (f_4), and popularity of candidate entities (f_5). The popularity is calculated as $\sqrt{k}$, where k represents the number of relation path the candidate entity has within 2-hop graph. We assume that an entity with larger f_1, f_2, f_4, and f_5 and smaller f_3 is more likely to be a topic entity.

These six features will be calculated and put into a linear weighing layer to output relative scores. Entities with Topk score build the candidate entities set.

The score is calculated using the following function:

$$s = w_1 \cdot f_1 + w_2 \cdot f_2 + w_3 \cdot f_3 + w_4 \cdot f_4 + w_5 \cdot f_5 \tag{4}$$

where f_i represents the i^{th} feature and w_i represents the corresponding weight.

3.3. Question Classification. In order to improve the efficiency of our model, we use a pretrained language model BERT to classify the complex questions into two categories, one topic entity question and multientity question, and process each of them separately. In one entity question, predicted paths usually extend from the topic entity with one relation or a sequence of relation hops. While in multientity questions, correct answers can only be obtained accurately by executing the query paths extended from several topic entities in the question. For instance, the question "Whose husband is the director of Avatar?" is one-entity question because its query paths (?, wife_of, t. t, director_of, Avatar) can be extracted from the "Avatar" through the relations "director_of" and "wife_of" and the transitional entity t. Meanwhile, "Which actors in Avatar born in British?" is a complex question because the correct query paths can only be generated from the entity "Avatar" and "British," respectively, through the relations "actor_of" and "born_in". In addition, we generate artificial questions in a semantic structured form to improve the performance of our classification model. The detailed implementation will be represented in Subsection 3.5.

Given a question, we encode it with words encoding, position encoding, and segment encoding and attach a special token [CLS] at the beginning of a question to separate different sentences. Then, the semantic information will be captured with a multihead attention system, and a dense layer will be attached to obtain the prediction.

3.4. Predefine the Query Schema. The golden key to solving the KBCQA task is to map entities of a question into a specific query graph. A semantic parsing-based model transfers the KBQA task into a Seq2Seq task. By feeding the model with numerous annotated data, SP-based model can understand the semantic framework of a question and refine corresponding query graph. An information retrieval-based model adopts a different method that searches all query graphs surrounding the extracted topic entities and then uses a candidate tuple ranking module to sift the final query graphs. However, with limited data, it is challenging to learn the query structure of questions, let alone changing it to an executable action sequence. In this work, we relieve this problem by predefining the schema of query graph and adopt beam search to pruning the searching space of multihop query paths.

Inspired by Aqqu [43], we propose an inverse solution that we first take a deep insight into numerous Chinese multihop questions and propose eight searching schemas for complex questions as shown in Figure 3. By predefining the schema of query graph, our model can benefit from three aspects:

(a) Predefining the schema introduces prior knowledge, which stipulates the semantic structure of the queried question and greatly prunes the search space

(b) Since the patterns of query tuples are specified, we can easily turn each query tuples into its semantic form and calculate the similarity between the artificial question and real question with a pretrained language model, which we define as the score of the query path we generate

(c) Extra data can be constructed on the enumerated query schema to train the classification model, which allows the model to learn the basic semantic knowledge of classifying questions

We assume that the diversity of candidate tuples will lead to poor performance of candidate query path ranking module. Thus, we divide the query schema into two modules according to number of topic entities the query pattern has.

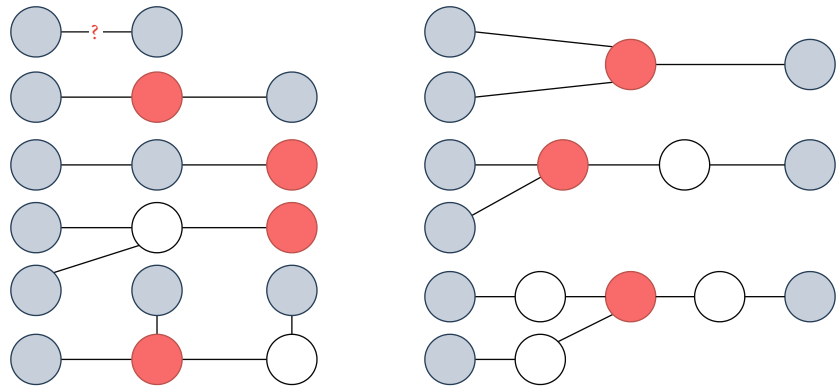


Figure 3: Predefined schema of query path.

Input: KB, question q, topic entity set E, number of hops T
Output: P^T
Initialize: $P^0 \longleftarrow \{e_0\} \in E$
for t=1,2,...,T **do**
  $\tilde{P}^{(t)} \longleftarrow \phi$
  $\tilde{S}^{(t)} \longleftarrow \phi$
  for each $p \in P^{(t-1)}$ **do**
    $e_{t-1} \longleftarrow tail(p)$
    for each $(e_{t-1}, r, e_t) \in KB$ **do**
      if $e_t \in E$ **then**
        $p' \longleftarrow p \oplus (r, e_t)$
      else
        $p' \longleftarrow p \oplus (r)$
      end if
      $\tilde{P}^{(t)} \longleftarrow \tilde{P}^{(t)} \cup \{p'\}$
      $\tilde{S}^{(t)} \longleftarrow \tilde{S}^{(t)} \cup \{Sentence(p')\}$
    end for
  end for
  score all elements in $\tilde{S}^{(t)}$ and rank all corresponding elements in $\tilde{P}^{(t)}$
end for

Algorithm 1: Multihop relation extraction. For each query schema, we generate a set of candidate query paths P^T, where T represents the hop number of the schema.

When generating query paths, we use two separate modules to generate candidate query paths. For one-entity question, we simply search the subgraph of the topic entity within two relation hops. While for questions of multiple entities, we generate query paths on the searching schemas shown in Figure 3. The gray ones represent topic entities we already know. The white one represents transitional entity we need not record, and the red one represents the answer we query. Let n represents the number of candidate topic entities, and m represents the number of true topic entities in a given question. Since combinatorial number C_n^m grows too large when m is greater than 3, we only consider questions containing three or fewer topic entities.

3.5. Artificial Data Construction. For better predicting which class a question belonging to and alleviating the need of labeled training data, we generate substantial artificial questions on the predefined query schemas. In our method, we randomly select a node entity in KB and extend a query path from the entity. When generating a query path, we are not to consider all branches in a random searching schema. Instead, we conduct the algorithm on the predefined query schema which has been introduced in Subsection 3.4. For instance, as for the above question "Whose husband is the director of Avatar?," the corresponding query schema is $(x, r_1, t.t, r_2, e.)$, where x represents the answer and r_1 and r_2 represent any relations in two-hop query path extended from the topic entity e through an intermediate entity t. We generate the artificial question by replacing mentions of topic entities (in this example is "Avatar") and relations ("wife_of", "director_of") with mentions of randomly selected node entities and correlated relations. In addition,

if the query schema is excluded in training samples, we only need to manually construct a fake question corresponding to the query schema and then execute the above steps.

Since our predefined query schema contains semantic structure for both one-entity and multientity questions, our constructed samples can lead the pretrained language model to converge in a direction which is more compatible with our specific classification task. Besides, the ratio of questions of different query patterns should be carefully controlled in order to improve the generalization of created data.

Although our constructed questions have some differences from the real questions in semantic expression, our model can still learn extra semantic structure of questions in two classes. In our experiment, we constructed 5 k artificial questions and use them to train our classification model. With the help of pretrained language model, our model can handle some questions that have never shown in training set. As the results in Section 4 shown, given only 10% of training data, our model can achieve good performance in classifying the questions.

3.6. Beam Search. It is worth to note that when extending multihop relations of the two type questions above, query path generation module often suffers from the vast searching space. To solve this, we adopt a heuristic algorithm beam search algorithm equipped with a pretrained language model BERT to score for each breach of relations; thus, we avoid exhaustive search on irrelevant relations. When extending a new relation path at n-step, we try to add the relation r_n to the previous generated query path R_{n-1} and use the strategy introduced in Artificial Data Construction to transfer the graph into a semantic form S_n. Then, S_n and original question Q are tokenized and concatenated with a special token [SEP] as

$$\text{input} = [\text{CLS}]S_n[\text{SEP}]Q. \tag{5}$$

This two sentences are fed into a pretrained language model of downstream task to calculate the semantic similarity which represents the score for r_n given a subquery path R_{n-1}. The formulation is defined as

$$Sco(r_n|R_{n-1}) = \text{BERTLayer}(\text{input}). \tag{6}$$

At each extending step, we only consider relations with Topk score for further search, which significantly excluded some irrelevant query branches. The result in Section 4.3.1 shows that by adopting the beam search algorithm, the accuracy of query path generation remains competitive, but the number of candidate paths decreases above 80%. The detailed description is seen in Algorithm 1.

4. Experiments

In this section, we study the performance our model achieves on complex question answering with limited training data. We take an insight into each module and conduct ablation experiments to better understand our model.

Table 1: Number of triples, entity type, and entity linking in PKU-Base.

Type	Triples	Entity type	Entity linking
Number of data	61,006,527	25,182,627	13,930,117

Table 2: Results of ablation experiments in entity linking module.

Type	One entity	Multientity
Baseline	0.848	0.726
w/o f_1	0.841	0.733
w/o f_2	0.848	0.721
w/o f_3	0.843	*0.744*
w/o f_4	0.838	0.706
w/o f_5	*0.849*	0.637

Table 3: We evaluate our model on primary training datasets, where created samples are excluded.

Data	Train	Valid	Test
10%	82.90	84.31	80.13
10% + created data	*87.51*	*89.54*	*82.75*
50%	94.95	*93.99*	88.50
50% + created data	*95.12*	93.72	*89.41*
100%	97.39	95.42	88.76
100% + created data	*99.09*	*95.45*	*91.11*

4.1. KB and Datasets. Our model uses an open-domain KB PKU-Base, which adopts resource description framework (RDF) as their data format and contains billions of SPO (subject, predicate, and object) triples [30], as shown in Table 1. We train and evaluate our model on CCKS datasets, which contain 2298, 766, and 766 pairs of questions.

4.2. Entity Linking. In entity linking module, we remove each feature of candidate entities to observe the influence on the performance of entity linking models. The left column is disassembled model, and the right is its recall of recognizing topic entities.

As is shown in Table 2, without f_3, the recall of multientity questions surprisingly increased while accompanied with a sacrifice of accuracy for one-entity questions. Similarly, without f_5, the topic entity extracting accuracy for questions of one topic entity increases, but the accuracy for multientity question drops. Moreover, excluding any of other features, the performance of entity linking model drops, which verifies their contribution for this module. Based on the results, we can modify the entity linking module by discarding feature f_5 in one-entity question's entity linking stage while only considering f_1, f_2, f_4, and f_5 when processing multientity questions. This will be included in our further study.

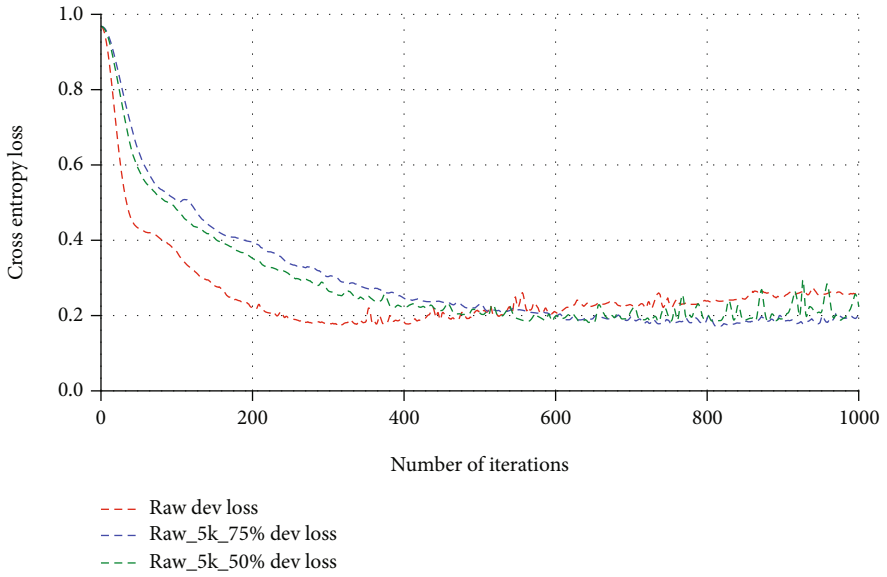


Figure 4: Loss of classification model.

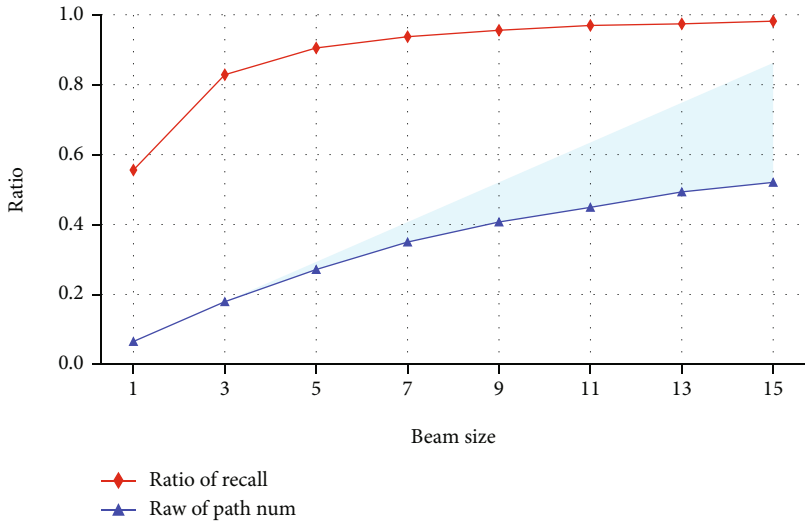


Figure 5: Ratio of recall and path numbers.

4.3. Question Classification. In this module, we construct 5 k artificial data based on the predefined query graphs and attached them to the training datasets. In order to evaluate the learning capability of our model on small amount of data, we train our model on 10%, 50%, and 100% randomly selected samples of primary training datasets and compare their performance with those additionally attached with certain number of created training samples.

Notably, when adding the constructed samples, we should carefully control the quantity according to the number of primary training samples. For one thing, negligible improvement of the learning ability can be brought, if the quantities of the added samples are too small. For the other, adding too many constructed data will bring knowledge noise, which leads the model to learn a distribution far away from the primary datasets. In our experiment, for 10%, 50%, and 100% primary training data, we add 0.05 k, 0.5 k, and 3.75 k manually constructed samples, respectively. The result is illustrated in Table 3.

From the above table, we find that when attached with manually constructed samples, our model's performance has improved on both partial and whole primary data. Our strategy can bring more significant improvement especially when given a small amount of training data. Moreover, we can see an obvious improvement of the prediction on training datasets, which indicates that appropriate number of created samples can make the model better fit the distribution of training data.

We owe the model's out performance to the introduction of prior knowledge. Due to the diversity of the samples in datasets, the test set may contain questions whose semantic structures have not appeared in training set. In this zero-

shot or few-shot situation, the model may have difficulty predicting the correct class. However, with additional created samples, our model can learn the predefined semantic structures. If these structures appear in test sets while not included by training set, the performance of our model will be improved. Thus, our model may need more steps to converge.

To verify the idea, we record the loss of each iteration when training with total primary data attached with 0%, 50%, and 75% created data, as shown in Figure 4.

We find that when training primary data attached with 0%, 50%, and 75% created data, our model converges at about 280, 550, and 760 steps, respectively, which indicates that with more created data, the model needs more iterations to converge.

4.4. Beam Search. For better exhibiting the effect of beam search (BS), we select 653 questions whose query path containing 2 hops of relations to test our methods. In the experiment, we design the benchmark by enumerating all the query paths within two-hop relations of the topic entity and recording the average number of query paths N. Notably, we only use BS algorithm at first hop, while searching for the second hop, we only extend from the reserved Top-K subquery path filtered by the BS algorithm and keep all the two hops query paths. By setting different beam size, we can observe the influence on the recall and number of generated query paths.

Figure 5 shows that a larger beam size will bring an increase in both recall and number of candidate query paths. Through further observation, we notice that the growth of both indexes slow down. The retarded growth of recall is intelligible. Due to the existence of upper bound, if the beam size is large enough, the recall will approach to and finally reach 1.0. However, the retarded increasing speed of the number of candidate tuples can illustrate something. When designing the score function for BS, we use a PTLM model to calculate the similarity of generated query paths and primary questions. Thus, the remaining one-hop relations are usually more relevant to the semantic information in the primary question. As the Figure 5 shows, extended from a relation with lower semantic score, the second hop tends to generate fewer query paths. Since the relation whose tail has more triples may have more probability to be the component of golden query path, we assume that the language model can be interpreted as a probabilistic model not only in the dimension of words but also in the dimension of query paths.

4.5. Final Result. We evaluate our model in the CCKS2019 datasets and compare our performance with a start-of-art model proposed by Wang et al. [42]. Their model first generated all query paths within 2 hops and adopted bridging technologies to handle questions with multiple topic entities. In candidate tuple ranking module, Lan [5] uses a PTLM model to calculate scores for generated query paths. Notably, their model introduces negative samples when training the semantic match model. Besides, since introducing bridging technology may harm the predicting performance of one-entity questions, they adopt a literal match technology to rerank the generated query path.

Table 4: Comparative results between our best model with other models.

Method	Negative sample	Avg F_1
Wang (baseline)	3	56.70
Wang (bridging)	3	58.60
Wang (bridging+literal match)	3	61.50
Wang (bridging+literal match)	1	61.10
Wang (bridging+literal match)	5	59.40
Our model (with 10% data)		58.54
Our model (with 100% data)		*62.55*

We implement their model and run it on a RTX 2080. It must to be pointed out that due to the difference of experimental equipment and subtle distinction of our datasets, the performance we obtain has some discrepancy with Wang proposed. However, since both our model and theirs are trained in the same experiment environment, the comparison is still persuasive in Table 4.

The result shows that our method is data-efficient and high-performed. Only using 10% data, our model can achieve competitive result. Moreover, when using 100% data, our model outperforms at over 1.0 point.

5. Conclusion

This paper proposes a KBQA system equipped with pretrained language model to handle multihop questions. We have shown that our model has the capability of answering multihop questions given small amount of data. Besides, experiments have been conducted to demonstrate that, by adopting beam search algorithm, we can achieve competitive results with much smaller cost of calculation and storage, which shows the superiority of our model for few-shot KBCQA task.

Data Availability

The dataset we use to train our model is CCKS2019 dataset, which can be accessed by the URL "https://github.com/pkumod/CKBQA." The knowledge base we used is also available by the URL "https://github.com/pkumod/gAnswer/tree/pkubase."

Conflicts of Interest

No competing interests exist within this work.

Acknowledgments

This work was partially supported by the Group Building Scientific Innovation Project for Universities in Chongqing (CXQT21021), the Innovation and Entrepreneurship Training Program for College Students (202110618001), and the Joint Training Base Construction Project for Graduate Students in Chongqing (JDLHPYJD2021016). We are also

grateful to Ren Li for providing us experimental equipment. Besides, it is worth noting that this work is preprinted at http://arxiv.org [44].

References

[1] Z. Sun, Y. Wang, Z. Cai, T. Liu, X. Tong, and N. Jiang, "A two-stage privacy protection mechanism based on blockchain in mobile crowdsourcing," *International Journal of Intelligent Systems*, vol. 36, no. 5, pp. 2058–2080, 2021.

[2] Y. Wang, Z. Cai, Z. H. Zhan, B. Zhao, X. Tong, and L. Qi, "Walrasian equilibrium-based multiobjective optimization for task allocation in mobile crowdsourcing," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 4, pp. 1033–1046, 2020.

[3] Z. P. Cai and Z. B. He, "Trading private range counting over big IoT data," in *2019 IEEE 39th International Conference on Distributed Computing Systems(ICDCS)*, pp. 144–153, Dallas, TX, USA, 2019.

[4] Y. Wang, Y. Gao, Y. Li, and X. Tong, "A worker-selection incentive mechanism for optimizing platform-centric mobile crowdsourcing systems," *Computer Networks*, vol. 171, article 107144, 2020.

[5] Y. Lan and J. Jiang, "Query graph generation for answering multi-hop complex questions from knowledge bases," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 969974, 2020.

[6] Z. Lu, Y. Wang, Y. Li, X. Tong, C. Mu, and C. Yu, "Data-driven many-objective crowd user selection for mobile crowdsourcing in industrial IoT," *IEEE Transactions on Industrial Informatics*, p. 1, 2021.

[7] C. Liang, J. Berant, Q. Le, K. D. Forbus, and N. Lao, "Neural symbolic machines: Learning semantic parsers on freebase with weak supervision," 2016, https://arxiv.org/abs/1611.00020.

[8] J. S. Sharath and R. Banafsheh, "Question answering over knowledge base using language model embeddings," in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, Glasgow, United Kingdom, 2020.

[9] S. Ou, C. Orasan, D. Mekhaldi, and L. Hasler, "Automatic question pattern generation for ontology-based question answering," in *Flairs Conference*, pp. 183–188, Menlo Park, 2008.

[10] C. Unger and P. Cimiano, "Pythia: compositional meaning construction for ontologybased question answering on the semantic web," in *International conference on application of natural language to information systems*, pp. 153–160, Springer, Berlin, Heidelberg, 2011.

[11] C. Unger, L. Bühmann, J. Lehmann, A. C. Ngonga Ngomo, D. Gerber, and P. Cimiano, "Template-based question answering over RDF data," in *Proceedings of the 21st international conference on World Wide Web*, pp. 639–648, Lyon, France, 2012.

[12] M. Yahya, K. Berberich, S. Elbassuoni, M. Ramanath, V. Tresp, and G. Weikum, "Natural language questions for the web of data," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 379–390, Jeju Island, Korea, 2012.

[13] C. Unger and C. P. Pythia, "Natural language question answering over RDF: a graph data driven approach," in *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pp. 313–324, New York, United States, 2014.

[14] W. Zheng, L. Zou, X. Lian, J. X. Yu, S. Song, and D. Zhao, "How to build templates for rdf question/answering: an uncertain graph similarity join approach," in *Proceedings of the 2015 ACM SIGMOD international conference on management of data*, pp. 1809–1824, New York, United States, 2015.

[15] B. Fu, Y. Qiu, C. Tang, Y. Li, H. Yu, and J. Sun, "A survey on complex question answering over knowledge base: recent advances and challenges," 2020, https://arxiv.org/abs/2007.13069.

[16] L. Dong, F. Wei, M. Zhou, and K. Xu, "Question answering over freebase with multi-column convolutional neural networks," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 260–269, Beijing, China, 2015.

[17] Y. Hao, Y. Zhang, K. Liu et al., "An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 221–231, Vancouver, Canada, 2017.

[18] H. Sun, B. Dhingra, M. Zaheer, K. Mazaitis, R. Salakhutdinov, and W. W. Cohen, "Open domain question answering using early fusion of knowledge bases and text," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4231–4242, Brussels, Belgium, 2018.

[19] H. Sun, T. Bedrax-Weiss, and W. W. Cohen, "Pullnet: open domain question answering with iterative retrieval on knowledge bases and text," 2019, https://arxiv.org/abs/1904.09537.

[20] Z. Y. Chen, C. H. Chang, Y. P. Chen, J. Nayak, and L. W. Ku, "UHop: An unrestricted-hop relation extraction framework for knowledge-based question answering," 2019, https://arxiv.org/abs/1904.01246.

[21] K. Luo, F. Lin, X. Luo, and K. Zhu, "Knowledge base question answering via encoding of complex query graphs," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2185–2194, Brussels, Belgium, 2018.

[22] G. Maheshwari, P. Trivedi, D. Lukovnikov, N. Chakraborty, A. Fischer, and J. Lehmann, "Learning to rank query graphs for complex question answering over knowledge graphs," in *International semantic web conference*, pp. 487–504, Auckland, New Zealand, 2019.

[23] S. Zhu, X. Cheng, and S. Su, "Knowledge-based question answering by tree-to-sequence learning," *Neurocomputing*, vol. 372, pp. 64–72, 2020.

[24] Y. Sun, L. Zhang, G. Cheng, and Y. Qu, "SPARQA: skeleton-based semantic parsing for complex questions over knowledge bases," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 8952–8959, 2020.

[25] Y. Hua, Y. F. Li, G. Qi, W. Wu, J. Zhang, and D. Qi, "Less is more: data-efficient complex question answering over knowledge bases," 2020, https://arxiv.org/abs/2010.15881.

[26] G. A. Ansari, A. Saha, V. Kumar, M. Bhambhani, K. Sankaranarayanan, and S. Chakrabarti, "Neural program induction for KBQA without gold programs or query annotations," *IJCAI*, pp. 4890–4896, Macao, China, 2019.

[27] D. Vrandečić and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Communications of the ACM*, vol. 27, pp. 78–85, 2014.

[28] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, Vancouver, Canada, 2008.

[29] A. Talmor and J. Berant, "TheWeb as a knowledge-base for answering complex questions," 2018, https://arxiv.org/abs/1803.06643.

[30] C. Unger, C. Forascu, V. Lopez et al., "Question answering over linked data (QALD-4)," *Working Notes for CLEF 2015-Conference and Labs of the Evaluation forum*, Toulouse France, 2014.

[31] Y. Sun, S. Wang, Y. Li et al., "Ernie: enhanced representation through knowledge integration," 2019, https://arxiv.org/abs/1904.09223.

[32] O. Etzioni, M. Cafarella, D. Downey et al., "Unsupervised named-entity extraction from the web: an experimental study," *Artificial Intelligence*, vol. 165, no. 1, pp. 91–134, 2005.

[33] M. Collins and Y. Singer, "Unsupervised models for named entity classification," in *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.

[34] G. D. Zhou and J. Su, "Named entity recognition using an HMM-based chunk tagger," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 473–480, Philadelphia, Pennsylvania, United State, 2002.

[35] R. Malouf, "Markov models for language-independent named entity recognition," in *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*, 2002.

[36] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," 2015, https://arxiv.org/abs/1508.01991.

[37] Z. Dai, X. Wang, P. Ni, Y. Li, G. Li, and X. Bai, "Named entity recognition using BERT BiLSTM CRF for Chinese electronic health records," in *2019 12th international congress on image and signal processing, biomedical engineering and informatics (cisp-bmei)*, pp. 1–5, Suzhou, China, 2019.

[38] W. Liu, X. Fu, Y. Zhang, and W. Xiao, "Lexicon enhanced Chinese sequence labelling using BERT adapter," 2021, https://arxiv.org/abs/2105.07148.

[39] Y. Lan, S. Wang, and J. Jiang, "Multi-hop knowledge base question answering with an iterative sequence matching model," in *IEEE International Conference on Data Mining (ICDM)*, pp. 359–368, Beijing, China, 2019.

[40] R. Chada and P. Natarajan, "FewshotQA: a simple framework for few-shot learning of question answering tasks using pre-trained text-to-text models," 2021, https://arxiv.org/abs/2109.01951.

[41] Y. Hua, Y. F. Li, G. Haffari, G. Qi, and T. Wu, "Few-shot complex knowledge base question answering via meta reinforcement learning," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5827–5837, 2002.

[42] X. L. Wang, S. C. Li, Z. H. Yang et al., "A Chinese KBQA system based on pre-trained language model," *Journal of Shanxi University(Natural Science Edition)*., vol. 43, pp. 955–962, 2020, (in Chinese).

[43] H. Bast and E. Haussmann, "More accurate question answering on freebase," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pp. 1431–1440, New York, United States, 2015.

[44] F. Meihao, "Few-shot multi-hop question answering over knowledge base," 2021, https://arxiv.org/abs/2112.11909.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 6902348, 10 pages
https://doi.org/10.1155/2022/6902348

Research Article

OTCS: An Online Target Close-Up Shooting Method Based on the UAV Image System

Wentao Wang,[1] **Huibin Wang**,[2] **Xuzhou Shi**,[1] **and Ming Chen**[1]

[1]*College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China*
[2]*College of Computer Science, Chuzhou University, Chuzhou 239099, China*

Correspondence should be addressed to Ming Chen; mingchennj@163.com

Received 27 October 2021; Revised 25 February 2022; Accepted 30 March 2022; Published 2 May 2022

Academic Editor: Yingjie Wang

Unmanned aerial vehicles (UAV) equipped with intelligent gimbaled cameras can take images and identify specific targets from them in real-time. However, the targets in the images are generally small and difficult to be seen. This paper proposes an Online Target Close-up Shooting (OTCS) method to solve the problem of taking high-definition (HD) close-up images of the targets. Firstly, we build an online target close-up shooting model, which uses deep learning (DL) algorithm to identify the targets from the images taken online. Then, an intelligent control algorithm is presented to control the gimbaled camera to take a close-up shot of the target, which considered the target being fixed on the ground statically and moving at an ununiform speed. Finally, we build a UAV-based prototype system and conduct a series of experiments to verify our proposal. Experimental results show that our proposed method is feasible and outperforms traditional methods in effectiveness.

1. Introduction

Aerial photography is an important function of unmanned aerial vehicles (UAV), which plays a crucial role in many applications, such as news reporting, rescuing, forest fire monitoring, transmission pipeline inspection, and traffic monitoring. Thanks to advantages including flexibility and high effectiveness, this image system technology enabled by UAV could expand the task coverage of the base stations on the ground, where UAVs are equipped with photography equipment [1, 2].

There is increasingly more research work on UAV image systems (e.g., [3–8]). The systems could currently use object detection algorithms to accurately identify specific targets and monitor their status. For example, [3] develops an advanced vehicle detection method to improve the original Viola-Jones object detection scheme for better vehicle detections from low altitude unmanned aerial vehicle imagery. [4] efficiently interleaves a fast keypoint tracker and presents a real-time approach for image-based localization within large scenes. [5] proposes a long-term tracking method on the basis of a multifeature coded correlation filter for vision-based UAV flocking control. However, the existing works on UAV image system overlook the target close-up shooting application to detect and take images of the specific targets. How to further obtain more pixel information of objects after detection is still a problem. Unlike UAV-based searching (e.g., [9, 10]), the purpose of this type of task is not only to check whether there are specific targets but also to take close-up images for more details of them.

On the other hand, traditional UAV aerial photography methods generally adopt the offline processing mode, i.e., the images taken by UAV are processed after the UAV landed. However, offline processing mode cannot meet the requirements of some urgent or real-time tasks, such as border patrol, surveillance, and power line inspection. These tasks usually adopt the online processing mode; that is, the images taken by UAVs during flights are processed in real-time. Online methods can be processed in ground stations, edge nodes, and on-board computers. Each of them has its own advantages and disadvantages. Ground stations and edge nodes could provide higher computing efficiency, but

the data links with the UAVs may be unstable in harsh environments. On-board computing, a form of computing that is done on site could minimize the data transmission delay at the expense of computing efficiency [11, 12].

Given the aforementioned requirements, we advance the research on the control method of UAV-based Online Target Object Close-up Shooting (OTCS). In addition to real-time object detection, the system is required to automatically adjust parameters such as the focal length of the camera, so as to take a high-definition (HD) of the target object with the largest possible area and as many pixels as possible (referred to as close-up images). We use YOLOv3 algorithm to detect the target object from the input image [13]. Gimbaled camera, an electronic camera supported by a three-axis pan-and-tilt, is mounted on the UAV to take images. People can control the direction of the camera by deflecting the pan-and-tilt and then adjusting the parameter of the camera such as focal length to change the camera screen [12, 14]. However, the gimbaled camera is mainly controlled by people, and there are still the following problems in intelligent object close-up: (1) lack of a system model that supports object close-up, (2) lack of pan-and-tilt deflection and camera zooming algorithm for static object close-up, and (3) lack of dynamic gimbaled camera control algorithm for dynamic object close-up.

In this backdrop, the technical route in this paper is as follows: first, the trained YOLOv3 algorithm is adopted to detect the target object from the taken images and build the relative positioning model of the object; the second is to study the intelligent control of the gimbaled camera under static scenarios and take HD images with more pixels of the object; the third is to study the close-up technology of intelligent control of the gimbaled camera under dynamic scenarios to further improve the efficiency of object searching. Moreover, a prototype system needs to be developed to verify the above theoretical results. Our main contributions in this paper are threefold: (1) A novel object close-up system model is put forward. (2) Intelligent gimbaled camera control algorithms in dynamic and static scenarios are presented to realize object close-up in a UAV-based system. (3) Extensive experiments are conducted to verify the effectiveness of our proposal. The remaining of this paper is organized as follows. Section 2 summarizes related work. Section 3 states the system on the basis of the YOLOv3 algorithm and introduces the gimbaled camera model. Section 4 presents our intelligent gimbaled camera control algorithms for objection detection and close-up in static and dynamic scenarios. Section 5 implements the prototype system and tests the gimbaled camera control algorithm. Section 6 concludes this paper briefly.

2. Related Work

In recent years, UAV technology has made great progress; there have been many studies on UAV image applications. The existing research on UAV image system focus on detection optimization, object tracking, and vision positioning. Firstly, many efforts for visual object detection have been conducted [15]. Rozantsev et al. investigated the problem of detecting flying objects with a single moving camera. And a regression-based approach for object-centric motion stabilization of image patches is proposed, which can achieve effective classification on spatio-temporal image cubes [9]. Dasgupta proposed a multiagent-based prototype system that uses swarming techniques inspired from insect colonies to perform automatic target recognition using UAVs. They presented algorithms for the different operations performed by the UAVs in the system and for different swarming strategies, which are embedded within software agents located on the UAVs [16]. Currently, deep learning (DL) technology has developed greatly. Compared with traditional object detection algorithms, Convolutional Neural Networks (CNN) have shown great advantages in accuracy and speed [17]. Therefore, CNN has been widely adopted in the field of computer vision [13]. Zhao et al. have adopted the on-board computing to meet real-time application requirements and used YOLOv3 algorithm to identify the vehicle in the images, but it does not consider further obtaining more object information [10]. Zhao et al. proposed a UAV inspection system, composed of a splicing module and a detection module. The splicing module obtains the video collected by the UAV camera, selects the frames in the video to splice into a panoramic image, and transmits it to the detection module. The detection module runs the Faster-RCNN algorithm to detect the object and returns a panoramic image with the detected object highlighted using the bounding box [18]. Furthermore, Zhang et al. presented SlimYOLOv3 with fewer trainable parameters and floating point operations (FLOPs) in comparison of original YOLOv3 as a promising solution for real-time object detection on UAVs [19]. To accomplish reliable pedestrian detection using unmanned aerial vehicles (UAVs) under night-time conditions, Wang et al. developed an image enhancement method to improve the low-illumination image quality [20].

Related to the real-time object tracking and vision positioning, there are only a few works. Tang et al. developed an integrated framework of tracking-learning-detection on the basis of multifeature coded correlation filter has been developed. To achieve long-term tracking, a redetector is trained online to adaptively reinitialize target for global sensing [5]. Liang et al. used the detection results to initialize the object tracker based on kernelized correlation filtering and go on to modify the tracking results. In the tracking process, a camera motion compensation strategy that adapts to the texture of the observation scenario is introduced to achieve target relocation, which enhances the robustness of the detection algorithm in complex scenarios [21]. Chen et al. put forward an object tracking control method, which divides the target tracking problem into three modes: object searching, object tracking, and object loss. For each mode, the corresponding control strategy is designed to realize the switch between different modes [22]. Zou et al. establish an on-board pan-tilt camera control system based on biomimetic eye, which can compensate the deflection caused by the UAV rotation and the movement of ground relative to the UAV [23]. However, the above methods about UAV image systems cannot apply to our target close-up problems, where the identification algorithm should be used to detect

the specific object and the gimbaled camera control should take into account object positioning and tracking. Compared with existing efforts listed above, our proposal in this paper could obtain close-up images with more information while detecting the target. Moreover, the computing modes of the UAV image system are compared and analyzed.

3. System Model

3.1. OTCS System. As shown in Figure 1, the OTCS system includes the following components: controller, pan-and-tilt, camera, and analyzer. The controller is the core of the OTCS, which coordinates the operation of various components of the system. Pan-and-tilt is used to carry the camera and control the direction of the camera. The camera is used to take images, and the analyzer is a computing device to detect whether an image contains specific objects. The OTCS system is scheduled by controller, which can call the controlling pan-and-tilt (PTC) and the controlling camera (CC) algorithm. The analyzer can run the object detection algorithm based on DL and output position parameters, which is used to control the pan-and-tilt camera. Gimbaled camera, i.e., pan-and-tilt and camera are connected with controller. Therefore, controller can control pan-and-tilt deflection to change the direction of the camera and adjust its focal length. The camera transmits the taken image to analyzer, which analyzes the image and the results are then be fed back to controller.

The basic principle of OTCS is as follows. Firstly, detect the target object O and compute relative position based on the YOLOv3 algorithm [5]. Secondly, control pan-and-tilt deflection and camera and take images with more pixels of object O_s to obtain more information. As shown in Figure 2, the process of OTCS is as follows.

(1) Controller controls camera to take pictures of P_0, where pan-and-tilt and camera are controlled based on specific parameters

(2) Controller sends P_0 to analyzer. Analyzer analyzes P_0 by DL algorithm. If P_0 contains suspected object O_s, analyzer will output the position coordinates in the image and confidence ε or it will return

(3) Controller calls controlling pan-and-tilt algorithm according to the position coordinates to control pan-and-tilt deflection so that the object can appear in the center of the camera screen

(4) Controller calls controlling camera algorithm to adjust parameters including focal length of camera to maximize the object in the image and take image P_{sm}

(5) Controller sends P_{sm} to analyzer. P_{sm} will be kept if the confidence is greater than ε_2. Then, the system restores the original parameters and waits for new instructions

3.2. Object Detection. We adopt the YOLOv3 algorithm to analyze the original image P_0 taken by the gimbaled camera. YOLOv3 has been trained with a large number of samples before processing and can quickly detect the object contained in P_0. The Darknet-53 network is used as the feature extractor, and it improves the inability of previous versions to accurately identify small objects [13].

YOLOv3 predicts bounding boxes using dimension clusters as anchor boxes. The output parameters include IsExist, ε and P_{size}, and Q_{all}. IsExist = true indicates that P_0 contains the suspected object O_s, and ε is the confidence of detection. As shown in Figure 3, P_{size}, i.e., $X_{\max} \times Y_{\max}$, is the size of P_0, and $Q_{lu}(x_{lu}, y_{lu})$, $Q_{ld}(x_{ld}, y_{ld})$, $Q_{rd}(x_{rd}, y_{rd})$, and $Q_{ru}(x_{ru}, y_{ru})$ indicate the vertices of the bounding box. The offset error of gimbaled camera is μ. Thus, we have:

$$\begin{cases} \text{IsExist} = \text{true}, P_0 \text{ contains } O_s, \\ \text{IsExist} = \text{false}, P_0 \text{ not contains } O_s. \end{cases} \tag{1}$$

If O_s lies in the center of P_0, we have

$$\begin{cases} \dfrac{X_{\max}}{2} - \mu < x_{lu} + \dfrac{x_{ru} - x_{lu}}{2} < \dfrac{X_{\max}}{2} + \mu \\ \dfrac{Y_{\max}}{2} - \mu < y_{ld} + \dfrac{y_{lu} - y_{ld}}{2} < \dfrac{Y_{\max}}{2} + \mu \end{cases}. \tag{2}$$

There is no need to zoom to enlarge O_s if it can fill the entire image, which is expressed as follows:

$$\begin{cases} \mu < x_{lu} < 2\mu, Y_{\max} - 2\mu < y_{lu} < Y_{\max} - \mu \\ \mu < x_{ld} < 2\mu, \mu < y_{ld} < 2\mu \\ X_{\max} - 2\mu < x_{rd} < X_{\max} - \mu, \mu < y_{rd} < 2\mu \\ X_{\max} - 2\mu < x_{ru} < X_{\max} - \mu, Y_{\max} - 2\mu < y_{ru} < Y_{\max} - \mu \end{cases}. \tag{3}$$

4. Gimbaled Camera Control Algorithm

4.1. Object Positioning Model. The gimbaled camera control of OTCS is divided into two parts, i.e., pan-and-tilt deflection and camera zooming control. The pan-and-tilt deflection determines the orientation of the camera, and the zooming determines the range of the camera [24]. Pan-and-tilt deflection is determined by course angle and pitch angle. As Figure 4 depicts, the course shaft is perpendicular to the ground and the rotation angle, i.e., course angle is in the range of 0 to 360 degrees. Correspondingly, the pitch shaft is perpendicular to the course shaft, and the pitch angle is in the range of -90 to 90 degrees.

After detecting the object, the relative position is computed according to the output. As shown in Figure 5, P_0's size is $X_{\max} \times Y_{\max}$, and its center coordinate m is $(X_{\max}/2, Y_{\max}/2)$. n is the center of detection box of YOLOv3, whose coordinate is $(x_{ld} + ((x_{ru} - x_{ld})/2), y_{ld} + ((y_{ru} - y_{ld})/2))$. The pixel distance between two center points is d_{mn}. The system obtains the relative position by computing the offset of the object in the camera screen. d_{mn} can be decomposed into

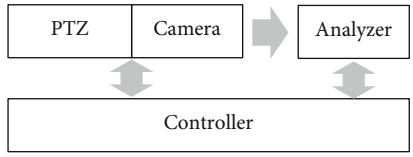


Figure 1: Main components of OTCS and relationships.

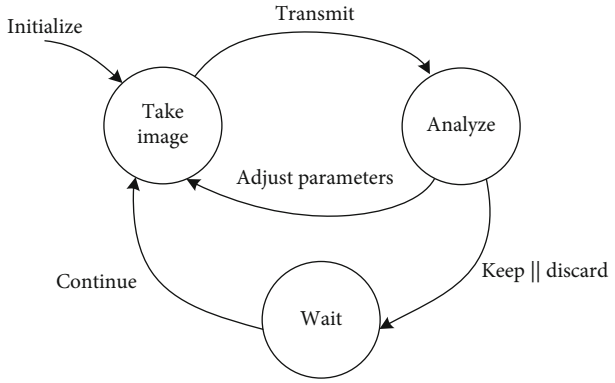


Figure 2: The finite state automaton of OTCS.

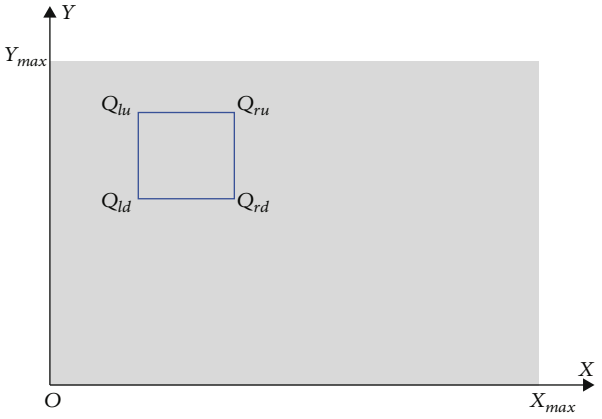


Figure 3: The pixel coordinate of the image.

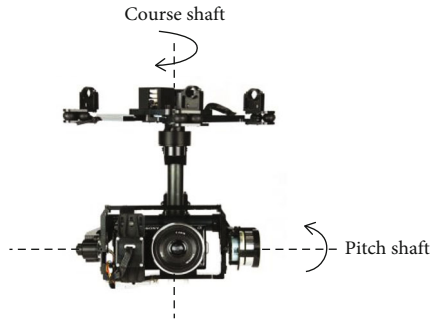


Figure 4: Pan-and-tilt posture.

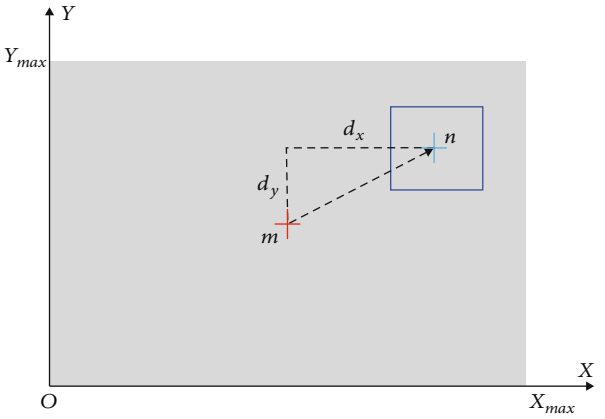


Figure 5: The location of the target object in the image.

two components, i.e., d_x and d_y, which correspond to the deflection of the course and pitch angle, respectively.

$$d_x = x_{ld} + \frac{x_{ru} - x_{ld}}{2} - \frac{X_{\max}}{2}, \tag{4}$$

$$d_y = y_{ld} + \frac{y_{ru} - y_{ld}}{2} - \frac{Y_{\max}}{2}. \tag{5}$$

Since d_{mn}, d_x, and d_y are pixel distances, and we need an actual distance between the object and the camera. It is necessary to reduce the computation error. We place an object with a known height of A vertically at the distance L in front of the camera; the center of the object is on the vertical line between the optical center and the object plane. We take an image of the object, and the pixel height of the object in the image is a. As shown in Figure 6, the focal length f of the camera is known; then, D_L can be computed according to the camera model:

$$D_L = \frac{a \times f}{A}. \tag{6}$$

As shown in Figure 6(b), D_L is the distance between the center of object O_s. Therefore, the actual distance D_{mn}, D_X, and D_Y, corresponding to d_{mn}, d_x, and d_y, are expressed as follows:

$$\begin{cases} D_{mn} = \dfrac{d_{mn} \times D_L}{f}, \\ D_L = \sqrt{D^2 - D_{mn}^2}, \end{cases}$$

$$D_L = \sqrt{\frac{D^2}{1 + \left(d_{mn}^2 / f^2\right)}}, \tag{7}$$

$$\begin{cases} D_X = \dfrac{d_x \times D_L}{f}, \\ D_Y = \dfrac{d_y \times D_L}{f}. \end{cases}$$

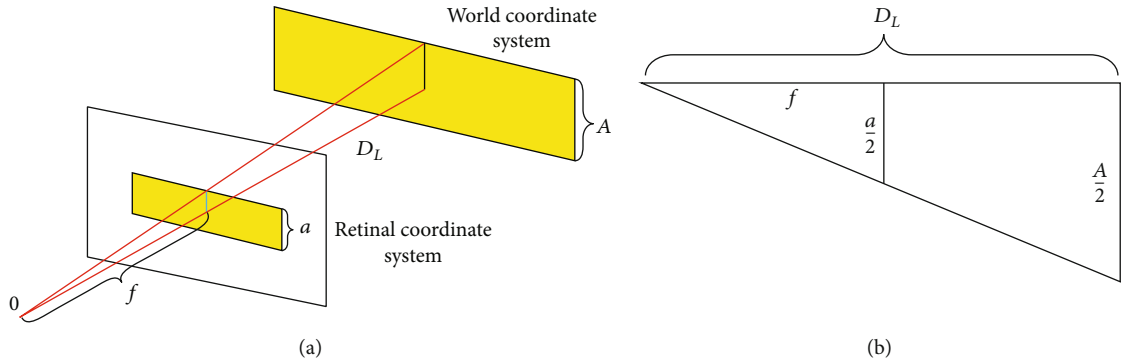


FIGURE 6: Image principle of the object.

4.2. Static Object Close-Up. In order to obtain HD images of the target object, we first analyze the scenario of static object close-up while the UAV is hovering at a certain height in the air. After the analyzer recognizes the target, controller controls the pan-and-tilt deflection according to its output information so that the camera can aim at O_s. Controller calls PTC and CC algorithm to control gimbaled camera according to the computed parameters.

According to the relative position of the object, adjusting the pan-and-tilt and camera focal length is needed to take HD images. As shown in Figure 7, D_L is perpendicular to D_X, and the angle α is the deflection angle.

$$\alpha = \arctan \frac{D_X}{D_L}. \tag{8}$$

Similarly, pitch angle deflection can be computed:

$$\beta = \arctan \frac{D_Y}{D_L}. \tag{9}$$

Then, the object is in the center of the screen after pan-and-tilt deflection. In order to obtain images with larger object and improve definition, the object in the image needs to be enlarged to the appropriate size via zooming. If the object, i.e., O_s, meets the Equation (5), the largest image can be obtained. The principle of camera imaging is shown in Figure 8.

f and D are the initial focal length and the distance between the camera and the object, respectively. $Y'/2$ is half of the Y-axis direction of the recognition frame, corresponding to half the height of the object, i.e., $h/2$. Considering the error of deflection, the minimum length of the detection box in Y-axis direction is

$$Y' - 2\mu - \mu. \tag{10}$$

The desired focal length can be expressed as follows:

$$f' = \frac{\left(\left(Y' - 3\mu\right)/2\right) \times D}{h/2}. \tag{11}$$

During the process of static object close-up, the relative position between the gimbaled camera and the object is fixed. After detecting the target object, controller calls the PTC algorithm and CC algorithm based on the parameters computed based on Equations (6)–(11). Algorithm 1 depicts this process.

4.3. Dynamic Object Close-Up. In order to improve the efficiency of object close-up, it is required that the system complete the close-up while the object is moving, not after the object has stopped. Since the target object can be detected by the analyzer when it appears on the camera screen, we use Kalman filtering to predict the motion state of the target and control gimbaled camera for a dynamic close-up. The Kalman filter is an algorithm that estimates the state of a system from measured data, which can optimally predict the state of the target by using a minimum-variance estimator [25, 26]. The Kalman filtering algorithm includes statement equation and measurement equation:

$$\begin{cases} x(k+1) = \Phi(k+1) \bullet x(k) + \Gamma(k+1) \bullet w(k), \\ y(k+1) = H(k+1) \bullet x(k+1) + v(k+1), \end{cases} \tag{12}$$

where $x(k)$ is the state of the system at time k, y is measurement vector, and w and v are state and measurement noise. Φ, Γ, and H are defined as the state transition matrix, noise transition matrix, and measurement matrix, respectively.

After obtaining the value of measurements, considering the estimation error, we have the Kalman filter equation in iterative form:

$$\begin{cases} \hat{x}(k|k) = \Phi(k) \bullet x(k-1|k-1) + K(k) \bullet [y(k) - H(k) \bullet \Phi(k) \bullet x(k-1|k-1)], \\ K(k) = P(k, k-1) \bullet H^T(k) \bullet \left[H(k) \bullet P(k, k-1) \bullet H^T(k) + R(k)\right]^{-1}, \\ P(k|k-1) = \Phi(k) \bullet P(k-1|k-1) \bullet \Phi^T(k) + \Gamma(k, k-1) \bullet Q(k-1) \bullet \Gamma^T(k, k-1), \\ P(k|k) = [I - K(k) \bullet H(k)] \bullet P(k|k-1)\,; k = 0, 1, \cdots \end{cases} \tag{13}$$

Equation (13) fall into two groups: time update equations and measurement update equations. The time update

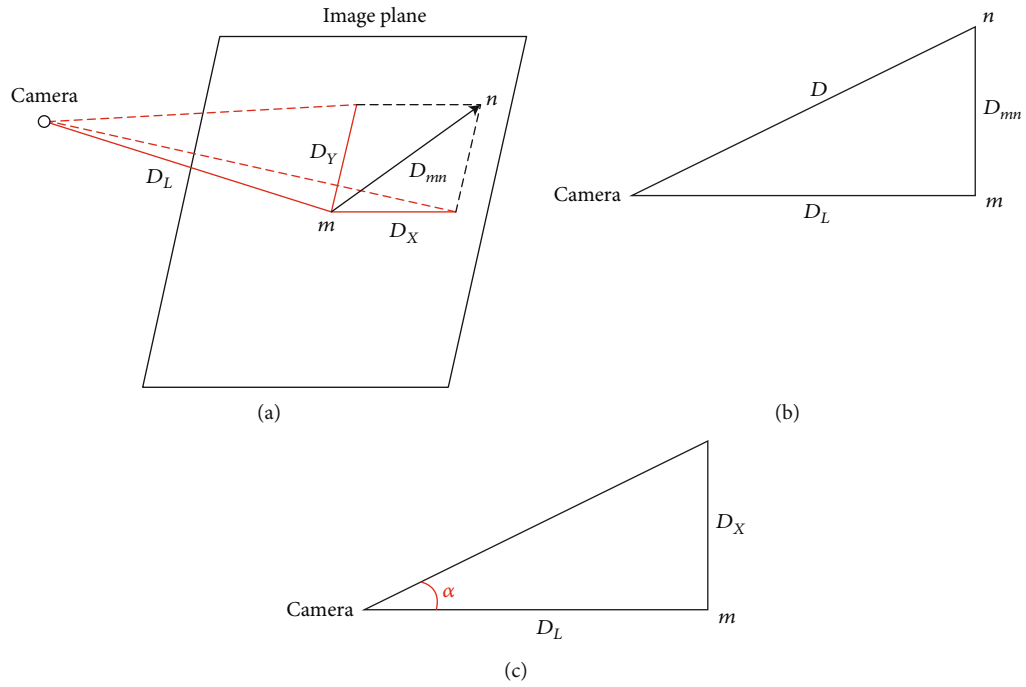


Figure 7: The computation of angles.

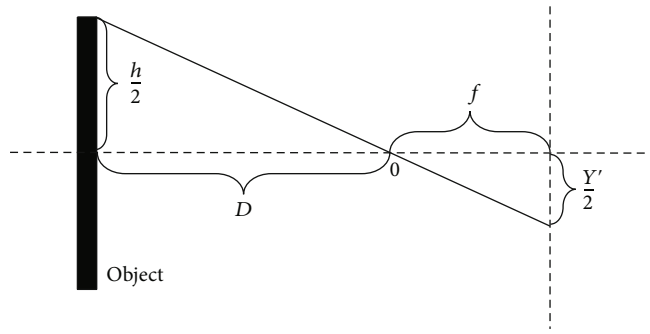


Figure 8: Camera imaging.

Input: Original image, Camera parameters
Output: Close-up image of the object
1: Run YOLOv3
2: **while** *IsExist* = *true* **do:**
3: Compute relative position parameter D_X, D_Y, D_L according to (9)
4: Compute α, β according to (10), (11)
5: PTC α, β
6: Compute f' according to (12)
7: CC zoom f' and focus
8: Take image P_{sm}
9: **if** $P_{sm}\, confidence > \varepsilon_2$ **do:**
10: Output P_{sm}
11: **end if**
12: **end while**

Algorithm 1: Static object close-up algorithm.

equations are responsible for projecting forward the current state and error covariance estimates to obtain the a priori estimates for the next time step. The Kalman filter computes a Kalman gain for each new measurement that determines how much the input measurement will influence the system state estimate. In other words, when a noisy measurement comes in to update the system state, the Kalman gain will trust its current state estimate more than this new inaccurate information, where $K(k)$ is the Kalman gain matrix and $P(k|k-1)$ is the prediction error covariance matrix. $\hat{x}(k|k)$ denotes the estimator based on given measurements; thus, we can predict the location of the target at time $k+1$:

$$\hat{x}(k+1|k) = \Phi(k+1) \bullet \hat{x}(k|k). \tag{14}$$

The state of the target is set as $x = (Q_{lu}, Q_{rd}, v_p)$, where Q_{lu} and Q_{rd} are the upper left and lower right vertices of the bounding box, respectively. $v_p = (v_x, v_y)$ is the pixel speed of the target in the screen. Then, we can use the Kalman filter to predict the location $\hat{x}$ and compute the deflection of the pan-and-tilt, i.e., $\widehat{\alpha}$ and $\widehat{\beta}$ based on (8)–(9). In order to keep the target tracking, we set angular velocity to control the pan-and-tilt. Let the interval of taking images be Δt, the optimal deflection velocity is

$$\begin{cases} \omega_\alpha = \dfrac{\Delta\alpha}{\Delta t} = \dfrac{\widehat{\alpha} - \alpha_0}{\Delta t}, \\ \omega_\beta = \dfrac{\Delta\beta}{\Delta t} = \dfrac{\widehat{\beta} - \beta_0}{\Delta t}, \end{cases} \tag{15}$$

where α_0 and β_0 are current deflection angle. The process of camera zooming may lead to the loss of the target. Therefore, we present a detection-tracking-shooting policy, which is to control the camera to take close-up image when the speed of the target is lower than 2 meters/second. The speed $\hat{v}$ will be computed through the relative position.

The dynamic close-up control algorithm is illustrated in Algorithm 2. Steps 1-2 get the measurements value through YOLOv3 algorithm. Steps 2-5 predict and track the target through the Kalman filter. Steps 6-11 compute control parameters and control the gimbaled camera to take close-up images. Finally, the image will be analyzed whether it can meet the requirement.

5. Experiments and Analysis

5.1. Experimental Prototype. In order to verify the feasibility of the OTCS and the performance of the gimbaled camera control algorithm presented in this paper, the prototype is built based on DJI Matrice 100 with an assembled communication module and gimbaled camera with network interfaces, and the algorithms are implemented on the onboard computer, i.e., NVIDIA Jetson Xavier NX. Moreover, the real-time images are transmitted to the ground station via the 4G network. The on-board computer controls the gimbaled camera via sending data messages. The camera supports 3.5x optical zoom and 5x digital zoom. A training dataset for object detection needs to be established for the efficiency of YOLOv3. To detect the object as accurately as possible from the image, we take a total of 2000 images from diverse angles and lightning conditions label them. Then, the labeled data set is divided into training set, validation set, and test set in a 6 : 2 : 2 scale.

In the test, the controller sends instructions to the camera through the HTTP protocol and saves the captured images on the web page. Then, controller extracts the feedback results from the website to obtain the captured pictures. The lens of the camera is first aligned at any position, and the object is placed to ensure that it can be photographed. The distance between the target object and the gimbaled camera is measured to simulate the GPS ranging function of the UAV. The test starts after placement. We use laptops and hosts equipped with high-performance graphics cards as edge nodes and ground stations, respectively.

5.2. Result Analysis. In the experiments, we first analyze the performance of three computing modes, in which data is processed in the UAV, edge node, and ground station. Since the delay of gimbaled camera control is decided by the mechanical structure, we only test the system delay. The default resolution of the images taken by the camera unit of this system is 4096×2160. Considering the needs of various real-time applications, images with 5 different resolutions, i.e., 640×480, 1280×960, 1600×1200, 2560×1920, and 4096×2160, are tested. Figure 9 shows images' processing delay of three different computing modes. As the definition of the photo increases, the system delay increases. Due to the increase of definition, YOLOv3 needs more time to detect. And it can be seen that the image resolution has a great effect on the processing time in the on-board computing. In edge computing, the resolution has less effect because ground station can provide more computing power. And it is almost unaffected due to the powerful computing power of the ground station in cloud computing.

Secondly, images need to be transmitted to edge node and ground station for processing in edge computing and cloud computing modes. We test the transmission delay of this process, which is shown in Figure 10. 4G communication is adopted in cloud computing, and we assume UAV moves within 4G coverage. Therefore, the transmission efficiency would not be affected by the distance between UAV and ground station. On the other hand, we adopt Wi-Fi in edge computing, and the delays at distances of 5 meters and 20 meters are given. It can be found that distance has a significant impact on Wi-Fi. When the distance reaches 20 meters, the delay is close to 4G, which weakens the advantages of edge computing over cloud computing.

Figure 11 shows the total computing delay of three computing modes, and the distance of Wi-Fi is 5 meters. We can see that the total delay of the on-board computing and edge computing using Wi-Fi transmission does not change much with the increase of the image resolution. However, the total computing delay of cloud computing using 4G increases rapidly with the increase of image resolution because of limited network bandwidth. Therefore, OTCS using on-

Input: Original image, Camera parameters
Output: Close-up image
1: Run YOLOv3 Algorithm // Get the measurements value
2: **while** *IsExist* = *true* **do:**
3: Compute optimal estimator according to (13)
4: Predict the state $\hat{x}$ according to (14)
5: Update Kalman filter
6: Compute $\omega_\alpha, \omega_\beta$ according to (15)
7: PTC $\omega_\alpha, \omega_\beta$
8: **if** $\hat{v} < 2m/s$ **do:**
9: CC zoom f' and focus
10: Take image P_{sm}
11: **end if**
12: **if** $P_{sm}\, confidence > \varepsilon_2$ **do:**
13: Output P_{sm}
14: **end if**
15: **end while**

Algorithm 2: Dynamic object close-up control.

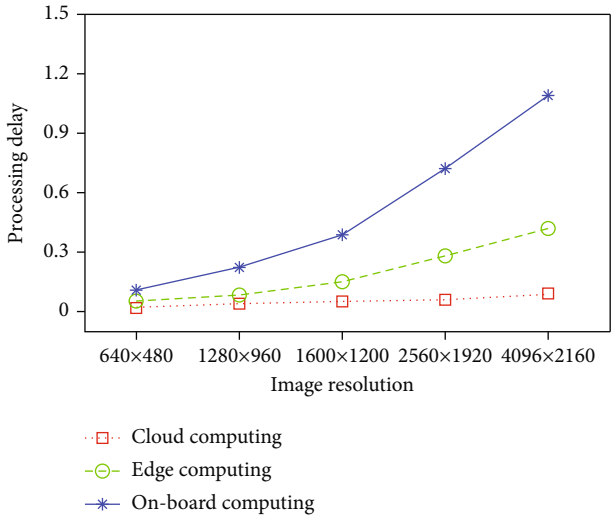


Figure 9: Processing delay.

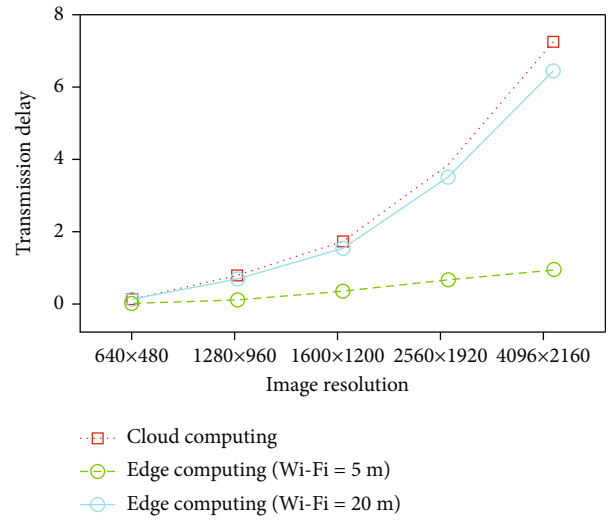


Figure 10: Transmission delay.

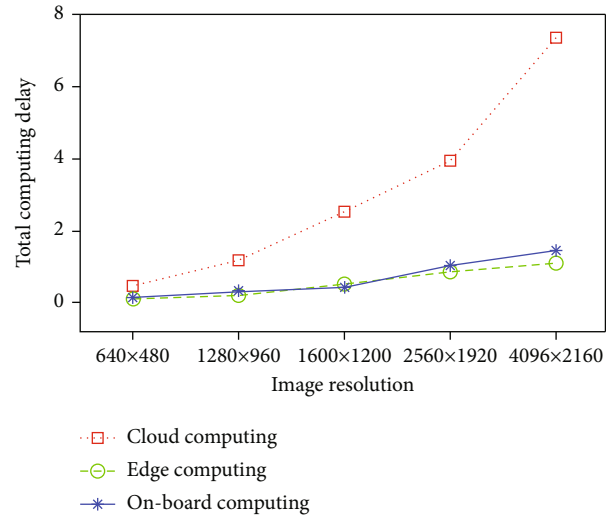
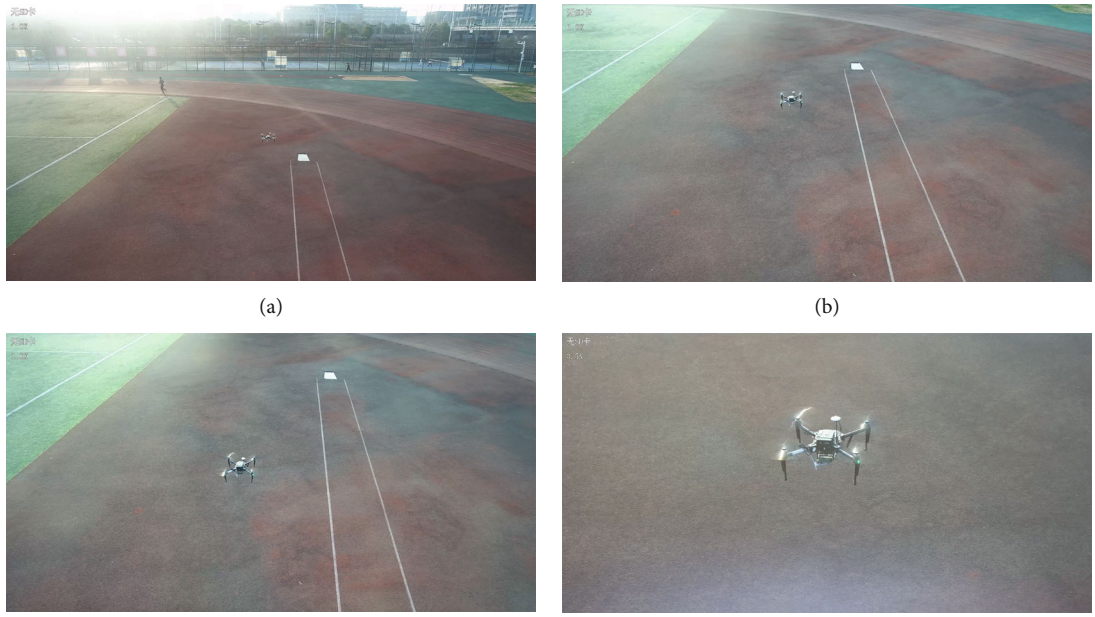


Figure 11: Total computing delay.

board computing can provide reliable computing power and is not affected by transmission distance. In contrast, the performance of edge computing and cloud computing depends more on UAV's communication capabilities.

We further test the dynamic control algorithm presented in this paper. Figure 12 shows the four snapshots during this process. The optical zoom of the camera is 3.5 times, and the image resolution is 4096 × 2160. The target object is a UAV, which moves at a nonuniform speed relatively. As shown in Figure 12(a), analyzer detects the object and computes the relative position of the object. Then, controller calls PTC algorithm so that pan-and-tilt starts deflecting. We can see that the gimbaled camera keeps track of the target object during its movement and takes a close-up image in Figures 12(b) and 12(c). Note that the screen is not zoomed in at first that is to avoid losing the target caused by its moving. In Figure 12(d), the object has been maximized in the image. Due to computing errors and the instability of the UAV flight process, the object does not appear in the center of the image.

We also compared our proposed prediction-based detection-tracking-shooting control method and traditional detection-shooting method, which take images after detection directly [11]. The UAV equipped with gambled camera is hovering at a height of 3 meters. The target object is initially placed where the camera can detect and then let it move. Tables 1 and 2 show the comparison results with different speeds. It can be seen that the success rate of the detection-shooting control is lower when the target moves fast. This is because camera zooming would cause the detection range to shrink, resulting in the loss of the target object. In contrast, our proposed algorithm could predict the target's position and take images when it moves slower, which could avoid object loss and increase the close-up rate.



(a) (b) (c) (d)

Figure 12: Image stream of dynamic close-up.

Table 1: Success rate of detection-tracking-shooting method.

Moving speed (m/s)	Repeating times	Close-up rate (%)	Object loss rate (%)
<1	20	100.0	0.0
1 ~ 3	20	85.0	5.0
>3	20	60.0	20.0

Table 2: Success rate of detection-shooting method.

Moving speed (m/s)	Repeating times	Close-up rate (%)	Object loss rate (%)
<1	20	80.0	20.0
1 ~ 3	20	15.0	85.0
>3	20	0.0	100.0

6. Conclusions

UAV-based object detection is a typical UAV image application. Obtaining additional image information of the target object after detection is of great practical importance. To achieve this goal, a UAV-carried gimbaled camera control model is presented in this paper. In the control model for the image systems, we analyze two scenarios, i.e., static and dynamic close-ups. Then, two control algorithms are put forward. The former could take static close-up images, while the latter takes close-up images in the latter scenario by estimating the relative position. A series of simulation experiments with diverse parameter settings is conducted based on the constructed prototype. Experiment results have shown that our gimbaled camera control algorithms could effectively obtain close-up images in diverse scenarios. In the future, we will study the application scenario of UAV-carried gimbaled camera to detect multiple objects and obtain close-up images.

Data Availability

All data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest

We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 61772271.

References

[1] B. Liu, Q. Zhu, and H. Zhu, "Trajectory optimization and resource allocation for UAV-assisted relaying communications," *Wireless Networks*, vol. 26, no. 1, pp. 739–749, 2020.

[2] T. Liu, M. Cui, G. Zhang, Q. Wu, X. Chu, and J. Zhang, "3D trajectory and transmit power optimization for UAV-enabled multi-link relaying systems," *IEEE Transactions on Green Communication and Networking.*, vol. 5, no. 1, pp. 392–405, 2021.

[3] Y. Xu, G. Yu, X. Wu, Y. Wang, and Y. Ma, "An enhanced Viola-Jones vehicle detection method from unmanned aerial vehicles imagery," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 7, pp. 1845–1856, 2017.

[4] H. Lim, S. N. Sinha, M. F. Cohen, and M. Uyttendaele, "Real-time image-based 6-DOF localization in large-scale

environments," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2012, pp. 1043–1050, 2012.

[5] Y. Tang, Y. Hu, J. Cui et al., "Vision-aided multi-UAV autonomous flocking in GPS-denied environment," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 1, pp. 616–626, 2019.

[6] J. Biswas and M. Veloso, "Depth camera based indoor mobile robot localization and navigation," in *2012 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1697–1702, Saint Paul, MN, USA, 2012.

[7] X. Xiang, M. Zhai, N. Lv, and A. el Saddik, "Vehicle counting based on vehicle detection and tracking from aerial videos," *Sensors*, vol. 18, no. 8, pp. 2560–2576, 2018.

[8] T. Tang, Z. Deng, S. Zhou, L. Lei, and H. Zou, "Fast vehicle detection in UAV images," *International Workshop on Remote Sensing with Intelligent Processing (RSIP)*, vol. 2017, pp. 1–5, 2017.

[9] A. Rozantsev, V. Lepetit, and P. Fua, "Detecting flying objects using a single moving camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 5, pp. 879–892, 2017.

[10] X. Zhao, F. Pu, H. Wang, H. Chen, and Z. Xu, "Detection, tracking, and geolocation of moving vehicle from UAV using monocular camera," *IEEE Access*, vol. 7, pp. 101160–101170, 2019.

[11] X. Dou, M. Chen, B. Chen, and Y. Xu, "Research on computing models of systems for realtime image applications based on UAV," *Journal of Chinese Computer Systems*, vol. 30, 2020.

[12] C. Martínez, I. F. Mondragón, M. Olivares-Méndez, and P. Campoy, "On-board and ground visual pose estimation techniques for UAV control," *Journal of Intelligent and Robotic Systems*, vol. 61, no. 1–4, pp. 301–320, 2011.

[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, Las Vegas, 2016.

[14] L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, "PIXHAWK: a system for autonomous flight using onboard computer vision," in *Proc. IEEE Int. Conf. Robot. Autom*, pp. 2992–2997, Shanghai, 2011.

[15] W. Diao, X. Sun, X. Zheng, F. Dou, H. Wang, and K. Fu, "Efficient saliency-based object detection in remote sensing images using deep belief networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 2, pp. 137–141, 2016.

[16] P. Dasgupta, "A multiagent swarming system for distributed automatic target recognition using unmanned aerial vehicles," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 38, no. 3, pp. 549–563, 2008.

[17] H. Lu, Y. Li, T. Uemura, H. Kim, and S. Serikawa, "Low illumination underwater light field images reconstruction using deep convolutional neural networks," *Future Generation Computer Systems*, vol. 82, pp. 142–148, 2018.

[18] Y. Zhao, T. Rui, Y. Li, and X. Zuo, "A UAV patrol system using panoramic stitching and object detection," *Computers and Electrical Engineering*, vol. 80, pp. 106473–106481, 2019.

[19] P. Zhang, Y. Zhong, and X. Li, "SlimYOLOv3: narrower, faster and better for real-time UAV applications," in *IEEE International Conference on Computer Vision*, Seoul, 2019.

[20] W. Wang, Y. Peng, G. Cao, X. Guo, and N. Kwok, "Low-illumination image enhancement for night-time UAV pedestrian detection," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5208–5217, 2021.

[21] D. Liang, S. Gao, H. Sun, and N. Liu, "UAV detection in motion cameras combining kernelized correlation filters and deep learning," *Acta Aeronautica et Astronautica Sinica*, vol. 42, no. 9, p. 323733, 2020.

[22] S. Chen, S. Guo, and Y. Li, "Real-time tracking a ground moving target in complex indoor and outdoor environments with UAV," in *2016 IEEE International Conference on Information and Automation (ICIA)*, pp. 362–367, Ningbo, China, 2016.

[23] H. Zou, Z. Gong, S. Xie, and W. Ding, "A pan-tilt camera control system of UAV visual tracking based on biomimetic eye," in *2006 IEEE International Conference on Robotics and Biomimetics*, pp. 1477–1482, Kunming, China, 2006.

[24] S. Chu, F. Zhang, N. Ji, Z. Jin, and R. Pan, "Pan-and-tilt self-portrait system using gesture interface," in *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*, pp. 599–605, 2017.

[25] W. Yuan, T. Hong, and M. Kadoch, "Improved Kalman filter variants for UAV tracking with radar motion models," *Electronics*, vol. 9, no. 5, p. 768, 2020.

[26] T. Basar, "A New Approach to Linear Filtering and Prediction Problems," in *Control Theory: Twenty-Five Seminal Papers (1932-1981)*, pp. 167–179, IEEE, 2001.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 2511147, 9 pages
https://doi.org/10.1155/2022/2511147

Research Article

BERT_LF: A Similar Case Retrieval Method Based on Legal Facts

Weifeng Hu,[1] **Siwen Zhao**,[1] **Qiang Zhao**,[1] **Hao Sun**,[1] **Xifeng Hu**,[1] **Rundong Guo**,[1] **Yujun Li**,[1] **Yan Cui**,[2] **and Long Ma**[3]

[1]*School of Information Science and Engineering, Shandong University, Qingdao 266200, China*
[2]*Institute of Sociology, Chinese Academy of Social Sciences, Beijing 100732, China*
[3]*Troy University 321D McCall Hall (MSCX), Troy AL 36082, USA*

Correspondence should be addressed to Yujun Li; liyujun@sdu.edu.cn and Yan Cui; cuiyanshky@sina.com

Received 23 February 2022; Accepted 30 March 2022; Published 30 April 2022

Academic Editor: Yingjie Wang

With the development of smart justice in China, the Supreme People's Court implements the system of compulsory retrieval for similar cases continuously and comprehensively, more and more judicial big data has been continuously disclosed, and the application of case retrieval is more extensive, and then, the accuracy of similar case search results needs to be urgently improved. Legal case retrieval is a special search task; for a given query case, it refers to the searching for similar cases. Different from traditional text search, legal case retrieval has different characteristics and greater challenges, for its query case is longer and more complex than common keyword queries and short article queries. In addition, the definition of dependencies between query cases and candidate cases differs from general dependencies based on text or topic. In order to solve these problems, we propose a method for similarity case retrieval based on the legal facts, and our model combine the topic distribution and legal entity facts to make the document representation vector more suitable for legal scenarios. At the same time, the method of paragraph aggregation based on BERT is used to encode context semantic information and solve the problem of long text. The experimental results show that our method is superior to the existing method.

1. Introduction

In many legal systems, similar case retrieval is of great significance to ensure legal fairness. With the development of smart justice in China and the increasing number of digitized legal documents, automatic retrieval of legal cases has attracted more and more attention in the research field of information retrieval (IR) [1–3]. In recent years, researchers have made many typical contributions to the retrieval of legal information [4–8].

The purpose of legal case retrieval is to identify cases that are similar to the given case. China has provided a guiding case series that can be referred to for the trial of similar cases. Guiding cases are composed of titles, keywords, key points of adjudication, relevant laws and regulations, basic case facts, adjudication results, adjudication reasons, and notes including the names of the effective adjudication and adjudicators. The problem of studying similar cases is essentially the study of text similarity. However, the legal case retrieval task differs greatly from traditional text retrieval in terms of the length of the case text, the definition of relevance, and the accessibility of the legal dataset. Based on the research of Shao et al. [9], there are several challenges in solving the problem under the existing text similarity method:

Challenge 1. Legal cases are often long texts, which causes models to fail to handle all useful information when establishing vector representations of text. At present, the memory ability of the most commonly used neural network models in the text field, such as LSTM, is not strong, and their application effect in long text is not good, which also leads to the poor effect of the general text similarity model in the legal field. Xiao et al. [10] proposed a model that combines local sliding window attention and global task-driven full attention, called Lawformer, for processing long texts

Challenge 2. The similarity of legal cases differs from the generic textual similarities and, to some extent, also goes beyond the general definition of subject matter relevance

[2]. It needs to explore the similarity of the legal facts contained in the legal case text. Traditional text similarity method can indeed learn the semantics similarity, but the model does not understand the knowledge of the legal field, so it may not be able to learn the deeper legal-related logical relationship under the surface semantics, which leads to the failed of finding highly similar legal cases using text similarity methods alone. Therefore, it is crucial to identify the similarities of cases in terms of legal issues and legal processes, which requires a full understanding of the legal case text

Challenge 3. Collecting large amounts of legal case data, as well as similar case datasets, is a challenge. On the one hand, in many legal systems, the download of large legal documents is restricted. On the other hand, the cost of obtaining accurate correlation judgments is higher due to the need for expertise in the legal field. The lack of data hampers the training process for deep neural models

What is more, the text structure of legal judgment documents is different from that of an ordinary text. The generic text similarity model mainly considers the structural characteristics of the text, such as syntactic structure, but although the legal judgment instrument is an unstructured text, it often has specific format requirements, so the general text similarity method cannot accurately represent the legal text; if the structural characteristics of the judgment document can be combined with the calculation of the similarity for the general text, it may produce better results.

To solve the problems above, we proposed a BERT-LF model; for challenge 1, literature [9] and literature [11–14] explored the long text problem applied by BERT, respectively, and their work included sentence-level fraction aggregation, paragraph-level fraction aggregation, and paragraph-level representation aggregation, so that the problem has been roughly solved, which inspired us to infer the similarity of the entire legal case by aggregating paragraph-level semantic interactions. For challenge 2, we proposed a legal case representation method based on legal facts, combining with the topic distribution. The deep combination of legal facts, document topic, and semantic information makes the document representation vector more suitable for legal scenarios. Further, we used an attention mechanism to distinguish the importance of legal information between paragraphs. For challenge 3, we crawled the judgment document data from "China Judgements Online" to train the topic model to adapt it to the legal scene. Our experiments were conducted on the legal case retrieval dataset [15], and the results proved the effectiveness of the proposed method.

2. Related Work

In the past researches, a large number of text retrieval models have been proposed, especially for specific texts. Literature [16–18] solved the problems of complex feature dimension and difficult retrieval of text data. Common approaches for early semantic representation include vector spatial models, topic models, and their variants such as the classic LDA [19]. But research in literature [20] showed that the similarity under the same topic still needed to be improved. With the advent of word embedding, information retrieval has now shifted to neural information retrieval. Researchers began using dense vector representations of words and documents based on deep learning models [21–24] as input of machine learning algorithms. Traditional bag-of-words IR models include BM25 [25], TF-IDF [26], and LMIR [27]. Mandal et al. [28] compared the effects of four unsupervised text vector generation models, TF-IDF, word2vec, LDA, and doc2vec, when calculating legal text similarity, and tested it on an Indian dataset containing 47 case pairs; the result showed that doc2vec worked best. Vo et al. [29] also indicated that text semantic representations based on word embedding are helpful in the field of legal text retrieval. Meanwhile, researches in literature [30, 31] showed the effectiveness of neural embedding of texts in legal information retrieval.

In view of the text similarity problem for Chinese, some scholars made a series of improvements to the classical similarity method. Li et al. [32] proposed a text similarity calculation algorithm based on improved VSM, which took into account the influence of the same feature words between similar texts. Huang et al. [33] proposed a supervised-WMD algorithm, which added new document features and movement costs to WMD and solved the problem that the WMD cannot take useful classification information into account.

Further, in the field of Chinese legal case retrieval, Lv and Hou [34] improved topic distribution model and designed a legal case recommendation algorithm. They argued that the words generated by the topic distribution model have different representations of legal texts. Thus, they reduced the probability distribution of words which appeared frequently but carried little weight with the legal text; what is more, they improved the probability distribution of words that did not appear frequently but were helpful for the representation of the legal text. In the similarity module of Xiang [35], the keywords were extracted by natural semantics and TF-IDF, the keywords of the judgment document were formed by semantic and frequency, and then, the judgment document was converted to vectors by the keyword table. Obviously, if you just consider whether the keywords are the same, you will ignore the contextual information. Wang et al. [36] compared the effect of the TF-IDF model, the LDA model, and the improved LLDA model on the task of case similarity and found that the TF-IDF had the worst effect and the LLDA model had the best effect. And they also point out that if you want to get a good effect, the parameters of the topic model need manual intervention. These similarity calculation methods based on word perspectives ignore the meaning of word order and context. Some subsequent studies have used the vector representation of word2vec to calculate case similarity. Deng [37] fused the word2vec, doc2vec, and TF-IDF algorithms and used them in the calculation of case similarity. Li [38] designed a method for calculating the similarity of documents that combined bipartite diagrams and syntactic information. The compressed document content was used to calculate the text similarity, and good results were obtained. Liu et al. [39] proposed a similar case recommendation model based on neural networks, which first used legal facts to

guide the generation of text representation vectors for each case, and then used the generated vectors to calculate the similarity scores of any pair of cases, and the set of cases with the highest similarity was used as the recommended similar cases. Although the above researchers have achieved certain results, most of the models are not designed for long legal documents.

Since BERT [40] has made significant improvements in various NLP tasks and achieved state-of-the-art performance in 11 missions, pretrained language models have attracted a great deal of attention in the field of information retrieval. Recently, several studies have elucidated the application of BERT in legal case retrieval, such as literature [9] and literature [11–14].

3. Materials and Methods

3.1. Task Description. Legal case retrieval task refers to finding cases similar to a given query case in the candidate cases set [8]. Formally, given a query case q, and a set of candidate cases $D = \{d_1, d_2, \cdots, d_n\}$, the task of legal case retrieval is to determine the supporting case $D* = \{d_i^* | d_i^* \in D \wedge \text{noticed}(d_i^*, q)a\}$, where noticed (d_i^*, q) indicates that d_i^* is legally similar to the query case q. Both the queries and candidates are long texts containing descriptions of legal facts.

3.2. Architecture Overview. As shown in Figure 1, in general, the entire framework of our model contains three modules, the first part is the legal feature encoding module, which contains the semantic encoding part based on BERT, the topical encoding module based on LDA model, and the encoding part based on legal entities; the second part is the aggregation of encoding; before entering the third part, the second part is responsible for encoding and aggregating the output of the first part; and the third part is the relational computation based on the attention mechanism.

3.3. Legal Feature Encoding. When judging whether two cases are similar, we are actually considering whether the legal facts and the logic of the events contained in the two cases are similar. Therefore, legal fact information is extracted through three coding modules; we capture the semantic context information of the case through the BERT-based module, cluster the topic information by the topical encoding module, and strengthen the role of legal facts information more accurately through the legal entity encoding module. And then, we aggregate all the above encodings to represent the paragraph-level information.

3.4. Semantic Encoding. Drawing on the analysis of the literature [9, 11–14], in the part of semantic encoding, we use paragraph aggregation architecture based on BERT. Firstly, we divide the long text into paragraphs that BERT can handle and then get semantic encoding of the query and candidate paragraphs based on a pretrained BERT model. On the one hand, it can take advantage of BERT's strong semantic learning ability, and on the other hand, it can solve the problem of long text encoding for legal cases.

Formally described, for a query document q and candidate document d_k which can be represented as $q = (p_{q1}, p_{q2}, \cdots, p_{qN})$ and $d_k = (p_{k1}, p_{k2}, \cdots, p_{kM})$ where N and M denote the total number of paragraphs for q and d_k, respectively. For each paragraph in q and d_k, we construct a paragraph pair (p_{qi}, p_{kj}), where $1 \le i \le N$ and $1 \le j \le M$, along with the reserved tags (i.e., [CLS] and [SEP]), serve as the input of BERT. We use the pretrained Chinese BERT model which called BERT-Base-Chinese (https://github.com/google-research/bert/blob/master/multilingual.md) to obtain a representation for each passage. Positional embeddings are added to capture word order, and these embeddings are fed into the transformer layers, where each layer of transformers generates a new upper and lower culture embedding representation by calculating the weighted sum of the token embeddings. The weight value is calculated by multihead attention mechanism. Words with a large attention weight are considered more relevant to the target word. Different attention matrices capture different types of word relationships, such as exact matching or synonym relationships. Finally, the final hidden layer vector output of the first token [CLS] is represented as the semantic aggregate of query-candidate paragraphs. As shown in Figure 1, we use the output embedding of the first token as the representation for the entire query-passage pair:

$$C_{ij} = \text{BERT}\left(p_{qi}, p_{kj}\right). \tag{1}$$

By this way, we can get an interaction matrix of all query-candidate paragraphs C, where the semantic representation of each paragraph for p_{qi} and p_{kj} is C_{ij}, $C_{ij} \in R^{HB}$. Next, interaction matrix C is further encoded with GRU model. Then, we get a sequence of hidden states generated by the forward GRU $h_{qk} = [h_{qk1}, h_{qk2}, \cdots, h_{qkN}]$, $h_{qki} \in R^{HR}$.

3.5. Topical Encoding. In this section, we obtain the topic probability interaction matrix of the query paragraph and the candidate paragraph pairs according to the inverse process of generating documents in LDA model.

As we all know, the process of generating documents in LDA is document generation, topic generation, and word generation, which are divided into the following five steps:

(1) Select a document m based on the prior probability

(2) Sample the topic distribution θ of the generated document from the Dirichlet distribution

(3) Sample the topic of the j-th word of the document from the polynomial distribution θ of the topic

(4) Generate a word distribution corresponding to the topic from the Dirichlet distribution based on prior knowledge φ

(5) Sample from the polynomial distribution of words φ to produce the final word ω

However, in the process of inferring the distribution of topics in a document, only the word w in document d is observable, the subject z is hidden, and the posterior

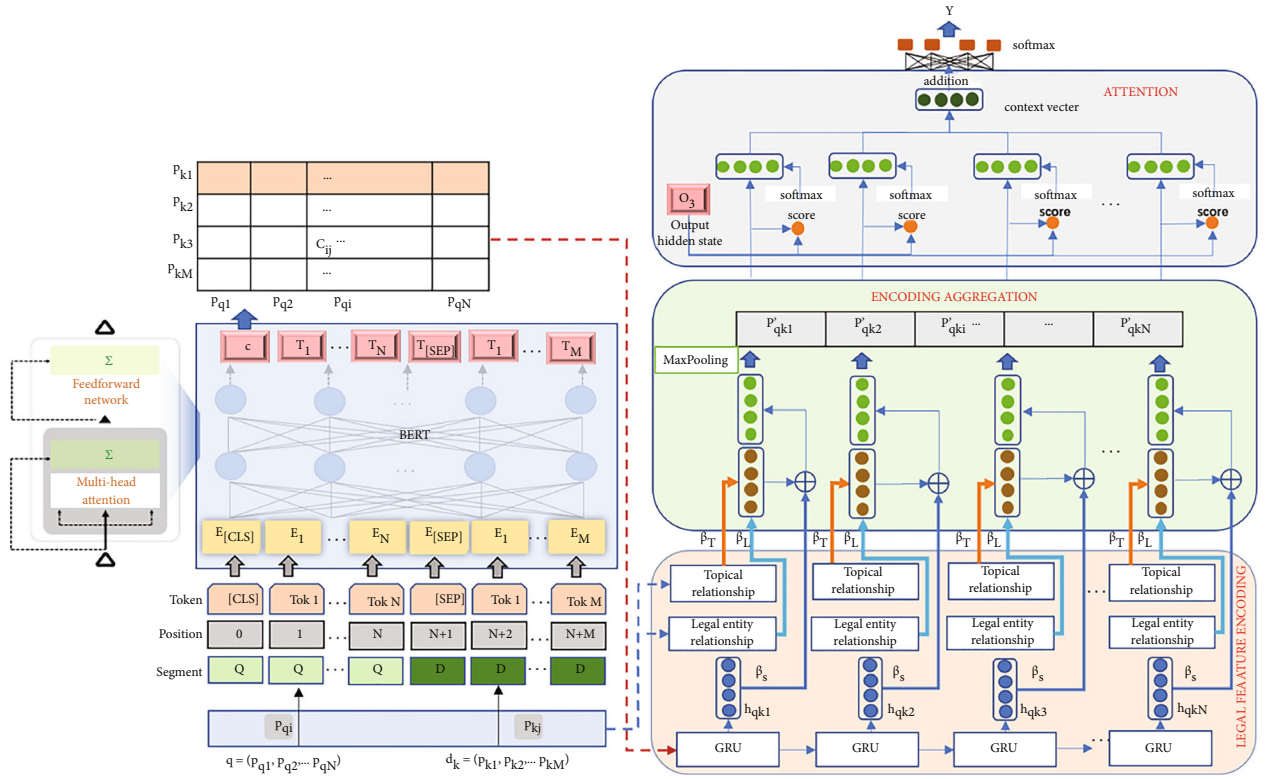


Figure 1: An illustration of BERT-LF.

distribution formula of the topic on each document is calculated as follows:

$$p\left(\overrightarrow{\theta_m}\middle|\overrightarrow{z_m},\overrightarrow{\alpha}\right)=Dir\left(\overrightarrow{\theta}_m\middle|\overrightarrow{n}_m+\overrightarrow{\alpha}\right),\tag{2}$$

where $\overrightarrow{\theta_m}$ is the topic distribution, $Dir(\overrightarrow{\theta}_m|\overrightarrow{n}_m+\overrightarrow{\alpha})$ represents the Dirichlet distribution of $\overrightarrow{\theta_m}$, $\overrightarrow{\alpha}$ is a priori distribution parameter of topic distribution (Dirichlet distribution), $\overrightarrow{n}_m$ is the subject number vector of document m, and $\overrightarrow{z_m}$ is the topic determined by the topic distribution. And then, the posterior distribution of topics is solved by Gibbs sampling method.

According to formula (2) the topic distribution of paragraphs P_{qi} and P_{kj} is $ZP_{qi}=[ZP_{qi-1},ZP_{qi-2},\cdots,ZP_{qi-v}]$ and $ZP_{kj}=[ZP_{kj-1},Z_{kj-2},\cdots,ZP_{kj-v}]$, respectively. Then, we use a similarity formula (3) to get the similarity interaction matrix between the query paragraphs and the candidate paragraphs about topic probability distribution $T_{qk}=[T_{q1k},T_{q2k},\cdots,T_{qNk}]$, where each element is represented by T_{qik}.

$$T_{qik}=\frac{ZP_{qi-1}*ZP_{kj-1}+ZP_{qi-2}*ZP_{kj-2}+\cdots+ZP_{qi-v}*ZP_{kj-v}}{\sqrt{\left(ZP_{qi-1}\right)^2+\left(ZP_{qi-2}\right)^2+\cdots+\left(ZP_{qi-v}\right)^2}*\sqrt{\left(ZP_{kj-1}\right)^2+\left(ZP_{kj-2}\right)^2+\cdots+\left(ZP_{kj-v}\right)^2}},\tag{3}$$

where v is a hyperparameter, representing the number of topics.

3.6. Legal Entity Encoding. We mainly focus on criminal cases in China. Referring to the previous research [39, 41, 42], this study mainly focuses on several parts of legal case including criminal offence, criminal entity type and compensation behavior, criminal consequences, reconciliation, and criminal charge. These legal entities contain the legal facts that have a decisive influence on the judgment.

Although legal judgment documents are unstructured text, the composition and writing order of judgment documents often depend on certain writing norms. In this research, we use regular expressions to extract legal facts and synonymously expand the legal facts contained in each case. Different types of cases have different legal facts. Taking the crime of intentional injury as an example, the legal entities are as follows:

Criminal entity type: government officials, minors, mentally ill, first offender, previous convictions, recidivists, etc.

Criminal offence: wound, hurt, injury, mutual beatings, intentional injury due to trivial disputes, etc.

Criminal consequences: economic loss, death, minor injury, serious injury, minor injury of the first degree, etc.

Compensation behavior: repentance, meritorious service, voluntary admission of guilt, voluntary surrender, etc.

Reconciliation: reach a settlement agreement, obtain the victim's forgiveness, etc.

Criminal charge: intentional injury crime

For the given query-candidate paragraph pair, the resulting entity sets are criminal entity type: E_{ct-qi}, E_{ct-kj}, criminal offence: E_{co-qi}, E_{co-kj}, criminal consequences: E_{cc-qi}, E_{cc-kj}, compensation behavior: E_{cb-qi}, E_{cb-kj}, reconciliation: E_{r-qi}, E_{r-kj}, and criminal charge: E_{ccg-qi}, E_{ccg-kj}. Then, we calculate the similarity of query-candidate paragraph pairs in terms of legal entities.

Firstly, we splice all entities of query paragraphs and candidate paragraphs into two short texts, T_{qi} and T_{kj}, in the order of criminal entity type, criminal offence, criminal consequences, compensation behavior, reconciliation, and criminal charge.

And then, we use the pretrained Chinese BERT BERT-Base-Chinese to obtain a representation for T_{qi} and T_{kj} separately and obtain tensor TS_{qi} and tensor TS_{kj}.

$$\begin{aligned} TS_{qi} &= \mathrm{BERT}\left(T_{qi}\right), \\ TS_{kj} &= \mathrm{BERT}\left(T_{kj}\right). \end{aligned} \tag{4}$$

Finally, we calculate the cosine similarity of tensor TS_{qi} and tensor TS_{kj}, as the entity similarity value of query-candidate paragraph pair.

$$\mathrm{sim\_}e\left(E_{qi}, E_{kj}\right) = \cos\left(TS_{qi}, TS_{kj}\right). \tag{5}$$

Then, we can obtain an interaction matrix between the query paragraphs and the candidate paragraphs about legal entity related $LE_{qk} = [LE_{q1k}, LE_{q2k}, \cdots, LE_{qNk}]$, where each element $LE_{qik} = [sim\_e(E_{qi}, E_{k1}), sim\_e(E_{qi}, E_{k2}), \cdots sim\_e(E_{qi}, E_{kM})]$.

3.7. Encoding Aggregation and Similarity Calculation. In this section, semantic encoding, topic distribution encoding, and legal entity encoding are aggregated, and the similarity of query-candidate pairs is calculated as follows:

$$E_{\mathrm{qki}} = \beta_s h_{\mathrm{qki}} + \beta_T T_{\mathrm{qik}} + \beta_L LE_{\mathrm{qik}}, \tag{6}$$

where β_s, β_T, and β_L are weight parameters. Then, for each paragraph of the query document, we use max pooling operation to get the strongest matching paragraph in candidate documents, resulting in a sequence vector expressed as $E_{qk} = [E'_{qk1}, E'_{qk2}, \cdots, E'_{qkN}]$, where E'_{qki} is obtained by the following aggregation operation:

$$E'_{qki} = \mathrm{MaxPool}(E_{i1}, E_{i2}, \cdots, E_{iM}), E'_{qki} \in R^{H_B}. \tag{7}$$

For the output of aggregated encoding, we add an attention mechanism to further encode the location information. The attention weight is calculation as follows:

$$\alpha_{qki} = \frac{\exp\left(E'_{qki} \bullet u_{qk}\right)}{\sum_{i'} \exp\left(E'_{qki'} \bullet u_{qk}\right)}, \tag{8}$$

where u_{qk} is calculated by

$$u_{qk} = W_u \bullet \mathrm{MaxPool}\left(E_{\mathrm{qk}}\right) + b_u, \tag{9}$$

where $W_u \in R^{HR \times HR}$, and $b_u \in R^{HR}$. Then, we use the following attentive aggregation operation to get the document-level representation:

$$d_{qk} = \sum_i \alpha_{qki} E'_{qki}. \tag{10}$$

Finally, we use a softmax function on d_{qk} to predict the relationship between two legal documents.

4. Experiments

4.1. Datasets and Evaluation Metrics. In this study, we use two legal text datasets, one is legal judgment document crawled from "China Judgements Online", and our topical encoding module is trained on this dataset. The other one is the LeCaRD open-source dataset provided by Tsinghua University, and our BERT-LF model is experimented on the dataset of LeCaRD.

The crawled legal judgment documents contain about 3.6 million legal judgment documents, covering more than 100 kinds of charges, of which the charge distribution with more than 3500 is shown in Table 1.

LeCaRD is a legal case retrieval dataset in China's legal system. It consists of 107 query cases and 10700 candidate cases, which are selected from more than 43000 criminal judgment corpora in China. The dataset is based on a series of key factors combined with subjective and objective evaluation as the correlation judgment standard. In order to ensure the diversity of cases, the dataset adopts sampling strategy, containing common query cases and controversial query cases.

4.2. Baseline Methods and Experimental Settings. We compared our model with the following baseline models:

4.2.1. Traditional Bag-of-Words Retrieval Models. We chose the traditional bag-of-words retrieval models including BM25, TF-IDF, and LMIR, following the previous work [15]. And all parameters of these three models are set to default values in an existing package [43].

4.2.2. Neural Network Model. We compared our model with BERT-PLI [9] since it is BERT-based model and solved the problem of long text of legal cases; most importantly, our model is an improvement based on BERT-PLI. For the baseline module of BERT-PLI, we set N=2 and M=8, HB = 768.

Table 1: Distribution of partial charges for legal judgment document data.

Charge name	Number	Charge name	Number
Larceny	798577	Intentionally destroying possessions	24963
Dangerous driving crime	625856	Bribery	21368
Intentional injury crime	394277	Illegal business crime	21047
Traffic accident crime	292823	Contract fraud crime	20101
Fraud	121544	Extortion crime	19560
Providing venues for drug users	110178	Affray crime	18565
Defiance and affray crime	106057	Corruption crime	18199
Robbery	59913	Official embezzlement crime	16772
Casino crime	59039	Negligence causing death crime	11193
Disrupting public service crime	44303	Rape crime	10447
Credit card fraud crime	38564	Offering bribes crime	8841
Illegal detention crime	35574	Crime of refusing to execute judgments or orders	5588
Illegally holding drugs crime	33392	Negligently causing serious accident crime	5530
Deforestation crime	34297	Crime of unlawful intrusion into residence	3660
Gambling crime	25304	Nongovernmental staff bribery crime	3581

As for RNN, HR is set as 256 and only one hidden layer is used. During the training process, we use the Adam optimizer and set the start learning rate as 2e-5.

In our BERT-LF model, the parameter settings are as follows: for legal feature encoding module, we set the total number of paragraphs for query document $N = 2$ and the total number of paragraphs for candidate document $M = 8$, HB = 768, which is determined by the size of the BERT hidden vector. As for GRU, HR is set as 256 and only one hidden layer is used. In training set, 10% queries from the training set and all of their candidates are treated as the validation set. We train the model on the training data left for 40 epochs and select the best model in the training process according to the precision measure on the validation set. During the training process, we use the Adam optimizer and set the start learning rate as 2e-5. For LDA model in topical encoding module, we set the quantity of topic $v = 7$.

5. Results and Discussion

For evaluation, we use two metrics including precision and ranking following literature [15]. Precision metrics include P@5, P@10, and mean average precision (MAP), and ranking metrics include NDCG@10, NDCG@20, and NDCG@30.

5.1. Overall Results. Comparison results between models are shown in Table 2. The comparison among the traditional three bag-of-words retrieval models show that LMIR performs best among the precision metrics, including P@5, P@10, and MAP, which is consistent with the conclusion of literature [15]. Under the same experimental conditions of this study, LMIR performed best in the three bag-of-words models. However, traditional retrieval models are difficult to handle long text retrieval, and the input length of these models is limited and cannot represent documents well. BERT-PLI outperforms traditional bag-of-words retrieval models in all ranking metrics, and it is structurally able to consider the entire case document and has better semantic understanding than traditional models. And BERT-LF is the best in all six indicators, and it not only considers the completed case documents and improves the semantic understanding ability but also adds the topic model and entity model to logically judge and analyze the legal elements between paragraphs.

5.2. Model Ablation. In order to further analyze the effects of each module, we conducted ablation experiments, removing the gain embedding and gain mask from BERT-LF, both or one at a time, and observe the impact on the performance compared to the full model. The experimental results are shown in Table 3. Only the topical encoding module is represented by SEM-TP. Only the legal entity encoding module is represented by SEM-EE. Only the semantic encoding module is represented by SEM-EC, the topical encoding added by the legal entity encoding module is represented by SEM-TE, the topical encoding added by the semantic encoding module is represented by SEM-T, and the legal entity encoding added by the semantic encoding module is represented by SEM-E. First, the ablations of the main components result in performance declining, verifying the effectiveness of these components for BERT-LF. SEM-EE and SEM-TE achieve a very large drop, indicating that using only legal element entities or paragraph topics of paragraphs cannot represent the entire text, and the semantic module is very important in the function of text encoding in this experiment. In addition, the performance of SEM-EC (LSTM) and SEM-EC (GRU) also drops significantly, which proves that the addition of topic model and legal element entity model improves the accuracy of paragraph logic judgment. Second, SEM-T (LSTM) and SEM-T (GRU) only use semantic encoding and topic encoding, with reduced accuracy. Third, the performance of SEM-E (LSTM) and SEM-E (GRU) drops slightly, which indicates that the encoding

Table 2: Comparison results between BERT-LF and baseline models.

Model	P@5	P@10	MAP	NDCG@10	NDCG@20	NDCG@30
BM25	0.39	0.39	0.475	0.725	0.785	0.872
TF-IDF	0.339	0.355	0.456	0.668	0.721	0.796
LMIR	0.45	0.394	0.539	0.731	0.791	0.88
BERT-PLI	0.32	0.355	0.436	0.743	0.807	0.891
BERT-LF	0.49	0.445	0.592	0.816	0.864	0.919

Table 3: The result of ablation study.

Model	P@5	P@10	MAP	NDCG@10	NDCG@20	NDCG@30
SEM-TP	0.38	0.365	0.444	0.717	0.779	0.884
SEM-EE	0.24	0.279	0.338	0.549	0.638	0.804
SEM-EC (LSTM)	0.36	0.335	0.415	0.711	0.78	0.879
SEM-EC (GRU)	0.32	0.355	0.436	0.743	0.807	0.891
SEM-TE	0.28	0.32	0.364	0.608	0.684	0.822
SEM-T (LSTM)	0.38	0.385	0.474	0.745	0.796	0.889
SEM-T (GRU)	0.456	0.412	0.546	0.788	0.816	0.905
SEM-E (LSTM)	0.46	0.41	0.546	0.799	0.818	0.904
SEM-E (GRU)	0.478	0.41	0.56	0.802	0.822	0.908
BERT-LF (LSTM)	0.51	0.43	0.58	0.803	0.832	0.909
BERT-LF (GRU)	0.49	0.445	0.592	0.816	0.846	0.919

of text topics cannot be ignored either. In the sequence encoding part of semantic module, LSTM and GRU are verified, respectively; the result showed that GRU has better effect, which is consistent with the conclusion of literature [9].

6. Conclusions

In this study, we proposed a model BERT-LF for similarity case retrieval based on the legal facts; our model combined the topic distribution and legal entity facts to make the document representation vector more suitable for legal scenarios. The study adopts the architecture of cutting and aggregation on paragraph, divides the long legal text into short paragraphs according to the logical order of the case, and then represents the query-candidate paragraph pairs through the BERT-based text encoding method. On the one hand, we can use the powerful semantic encoding ability of BERT; on the other hand, we can solve the problem of long text coding of legal cases. In order to accurately excavate the legal elements in legal cases, this study excavates several legal entities that have a decisive impact on the case judgment, including charges, crime, types of criminal entity, criminal consequences, compensation behavior, and reconciliation. Through convolution neural network and attention mechanism, it not only encodes the position information of legal semantics in paragraphs but also logically judges and strengthens the legal elements between paragraphs. The experimental results demonstrate that our approach is effective in legal case retrieval and the combination with topic distribution and legal entity facts can further improve models for this task.

Data Availability

The dataset used to support the topic encoding module of this study is available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Authors' Contributions

Weifeng Hu and Siwen Zhao contributed equally to this work.

Acknowledgments

This work was supported by the National Key R&D Program of China under Grant 2020YFC0833303 and the Major Project of Independent Innovation in Qingdao 21-1-2-18-xx.

References

[1] T. Bench-Capon, M. Araszkiewicz, K. Ashley et al., "A history of AI and law in 50 papers: 25 years of the international conference on AI and law," *Artificial Intelligence and Law*, vol. 20, no. 3, pp. 215–319, 2012.

[2] M. V. Opijnen and C. Santos, "On the concept of relevance in legal information retrieval," *Artificial Intelligence and Law*, vol. 25, no. 1, pp. 65–87, 2017.

[3] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social

networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[4] Z. P. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[5] Z. P. Cai and Z. B. He, "Trading private range counting over big IoT data," in *The 39th IEEE International Conference on Distributed Computing Systems*, pp. 144–153, Dallas, TX, USA, 2019.

[6] D. W. Oard and W. Webber, "Information retrieval for e-discovery," *Foundations and Trends in Information Retrieval*, vol. 7, no. 2-3, pp. 99–237, 2013.

[7] P. Bhattacharya, K. Ghosh, S. Ghosh, and A. Pal, *Overview of the FIRE 2019 AILA Track: Artificial Intelligence for Legal Assistance*, FIRE (Working Notes), 2019.

[8] J. Rabelo, M.-Y. Kim, R. Goebel, M. Yoshioka, Y. Kano, and K. Satoh, "A Summary of the COLIEE 2019 Competition," in *JSAI International Symposium on Artificial Intelligence*, pp. 34–49, Springer, Cham, 2019.

[9] Y. Shao, J. Mao, Y. Liu et al., "BERT-PLI: modeling paragraph-level interactions for legal case retrieval," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence Main track*, pp. 3501–3507, Yokohama, 2020.

[10] C. Xiao, X. Hu, Z. Liu, C. Tu, and M. Sun, "Lawformer: a pre-trained language model for Chinese legal long documents," *AI Open*, vol. 2, pp. 79–84, 2021.

[11] Z. A. Yilmaz, W. Yang, H. Zhang, and J. Lin, "Cross-Domain Modeling of Sentence-Level Evidence for Document Retrieval," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 3490–3496, Hong Kong, China, 2019.

[12] Z. Dai and J. Callan, "Deeper text understanding for ir with contextual neural language modeling," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 985–988, Paris, France, 2019.

[13] Z. Wu, J. Mao, Y. Liu et al., "Leveraging passage-level cumulative gain for document ranking," in *Proceedings of The Web Conference 2020*, pp. 2421–2431, Taipei, Taiwan, 2020.

[14] C. Li, A. Yates, S. MacAvaney, B. He, and Y. Sun, "PARADE: passage representation aggregation for document reranking," 2020, https://arxiv.org/abs/2008.09093.

[15] Y. Ma, Y. Shao, Y. Wu et al., "LeCaRD: a legal case retrieval dataset for Chinese law system," in *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021.

[16] X. Ben, Y. Ren, J. Zhang et al., "Video-based facial micro-expression analysis: a survey of datasets, features and algorithms," *Institute of Electrical and Electronics Engineers transactions on pattern analysis and machine intelligence*, vol. PP, p. 1, 2021.

[17] X. Ben, P. Zhang, Z. Lai, R. Yan, X. Zhai, and W. Meng, "A general tensor representation framework for cross-view gait recognition," *Pattern Recognition*, vol. 90, pp. 87–98, 2019.

[18] X. Ben, C. Gong, P. Zhang, X. Jia, Q. Wu, and W. Meng, "Coupled patch alignment for matching cross-view gaits," *IEEE Transactions on Image Processing*, vol. 28, no. 6, pp. 3142–3157, 2019.

[19] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[20] X. Ben, C. Gong, P. Zhang, R. Yan, Q. Wu, and W. Meng, "Coupled bilinear discriminant projection for cross-view gait recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 3, pp. 734–747, 2020.

[21] X. Yang, F. Feng, W. Ji, M. Wang, and T. S. Chua, "Deconfounded video moment retrieval with causal intervention," in *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021.

[22] X. Yang, S. Wang, J. Dong, J. Dong, M. Wang, and T. S. Chua, "Video moment retrieval with cross-modal neural architecture search," *IEEE Transactions on Image Processing*, vol. 31, pp. 1204–1216, 2022.

[23] X. Yang, J. Dong, Y. Cao, X. Wang, M. Wang, and T. S. Chua, "Tree-augmented cross-modal encoding for complex-query video retrieval," in *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pp. 1339–1348, Xi'an, China, 2020.

[24] J. Dong, X. Li, C. Xu et al., "Dual encoding for video retrieval by text," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, 2021.

[25] S. E. Robertson and S. Walker, "Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval," in *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 232–241, London, 1994.

[26] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, 1988.

[27] J. M. Ponte and W. B. Croft, "A language modeling approach to information retrieval," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 275–281, Melbourne, Australia, 1998.

[28] A. Mandal, R. Chaki, S. Saha, K. Ghosh, A. Pal, and S. Ghosh, "Measuring similarity among legal court case documents," in *Proceedings of the 10th annual ACM India compute conference*, pp. 1–9, Bhopal India, 2017.

[29] N. P. A. Vo, C. Privault, and F. Guillot, "Experimenting word embeddings in assisting legal review," in *International Conference on Artificial Intelligence and Law*, pp. 189–198, London, United Kingdom, 2017.

[30] A. Mandal, K. Ghosh, A. Bhattacharya, A. Pal, and S. Ghosh, *Overview of the FIRE 2017 IRLeD Track: Information Retrieval from Legal Documents.*, FIRE (Working Notes), 2017.

[31] X. Zheng and Z. P. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 38, no. 5, pp. 968–979, 2020.

[32] L. Li, A. Zhu, and T. Su, "Research and implementation of an imporved VSM-based text similarity algorithm," *Computer Applications and Software*, vol. 29, no. 2, pp. 282–284, 2012.

[33] G. Huang, C. Guo, M. J. Kusner, Y. Sun, F. Sha, and K. Q. Weinberger, "Supervised word mover's distance," *Advances in Neural Information Processing Systems*, pp. 4862–4870, 2016, https://proceedings.neurips.cc/paper/2016/hash/10c66082c124f8afe3df4886f5e516e0-Abstract.html.

[34] B. Lv and W. L. Hou, "Typical case recommendation of court texts based on topic model," *Microelectronics & Computer*, vol. 35, no. 2, pp. 128–132, 2018.

[35] L. X. Xiang, *Design and Implementation of Referee Document Recommendation System Based on Natural Semantic Processing*, Nanjing University, China, 2015.

[36] Y. Wang, J. Ge, Y. Zhou et al., "Topic model based text similarity measure for Chinese judgment document," in *International Conference of Pioneering Computer Scientists, Engineers and Educators*, pp. 42–54, Singapore, 2017.

[37] W. C. Deng, *Research on Judicial Intelligence Based on Deep Learing*, School of Computer Science and Technology, Harbin Institute of Technology, 2017.

[38] L. J. Li, *Computing Document Similarity for the Legal Case Retrieval*, School of Computer Science and Technology, Nanjing Normal University, 2018.

[39] B. Y. Liu, S. Li, L. Ye, and H. Zhang, "Similar case recommendation algorithm based on legal elements," *Intelligent Computer and Applications*, vol. 11, no. 6, pp. 1–4, 2021.

[40] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," 2018, https://arxiv.org/abs/1810.04805.

[41] Z. P. Cai, Z. B. Xiong, H. H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–38, 2021.

[42] Y. Lyu, Z. Wang, Z. Ren et al., "Improving legal judgment prediction through reinforced criminal element extraction," *Information Processing & Management*, vol. 59, no. 1, p. 102780, 2022.

[43] R. Rehurek and P. Sojka, *Gensim-statistical semantics in python*, Retrieved from genism. org, 2011.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 5390771, 11 pages
https://doi.org/10.1155/2022/5390771

Research Article

A Secure Downlink Transmission Scheme for a UAV-Assisted Edge Network

Xinmei Gao,[1,2] Yan Huo,[1] Qinghe Gao,[1] Hongjun Zhao,[3] and Long Ma[4]

[1]*School of Electronics and Information Engineering, Beijing Jiaotong University, Beijing 100044, China*
[2]*Hebei Key Laboratory of Power Internet of Things Technology, North China Electric Power University, Baoding, Hebei 071003, China*
[3]*Beijing Research Institute of Automation for Machinery Industry LTD., Beijing 100120, China*
[4]*Computer Science Department, Troy University, Troy, AL 36082, USA*

Correspondence should be addressed to Yan Huo; yhuo@bjtu.edu.cn

Received 26 January 2022; Revised 9 March 2022; Accepted 28 March 2022; Published 21 April 2022

Academic Editor: Yingjie Wang

As an extension of centralized cloud services in a space-air-ground integrated network, an unmanned aerial vehicle (UAV-)-enabled edge network brings computation as close ground terminals of need as possible. In this scenario, the UAV is considered a mobile base station (MBS) to achieve data forwarding and processing from cloud servers to terminals. Note that downlink signals from the MBS to ground terminals are vulnerable to passive wiretapping by a malicious node. We formulate an average secrecy rate maximization problem to tackle the wiretapping issue by simultaneously transmitting confidential information and artificial noise (AN). We decompose the problem into two subproblems, including the UAV transmit power allocation subproblem and the UAV trajectory subproblem. Then, we adopt the successive convex approximation scheme and the alternating optimization method to develop our iterative algorithm to achieve secure transmission. Simulation results demonstrate that the proposed scheme is significantly better than other benchmarks in UAV-enabled downlink communication secrecy performance.

1. Introduction

Data-intensive applications such as video streaming are experiencing an unprecedented surge as digital savvy generation rises. In conjunction with the rapid development of mobile computing and the Internet of Things (IoT), large volumes of data have been moving within edge networking [1]. As an extension of backbone networks, edge computing enables terminals with limited resources to upload computing-intensive tasks to edge servers for execution by deploying servers near edge users. It can greatly reduce computational load and communication costs of terminals and cloud servers [2]. However, complex environments may introduce high infrastructure deployment costs, especially for remote areas and emergency relief applications.

Unmanned aerial vehicles (UAVs-)-enabled wireless network is considered a promising paradigm to provide beyond line-of-sight (LOS) signal transmission. Due to high maneuverability, flexible deployment, low cost, and strong hover ability, it is expected to be one of the critical steps in a space-air-ground integrated network (SAGIN). Many works on UAVs-enabled communications include air-to-ground (A2G) channel modeling, optimal throughput-based deployment, and multi-UAV cooperative communication [3]. These technologies may be conveniently introduced into most scenarios and applications, e.g., surveillance missions, weather monitoring, emergency search, and recognition missions [4].

With the gradual maturity of UAV technology, UAVs with flexible operation, good scalability, and adaptability to

various harsh environments bring more possibilities for network construction, route inspection, and other scenarios. Combining UAVs with a mobile edge computing (MEC) network and assuming UAVs as MEC servers, we can realize rapid deployment and flexible unloading of computing tasks. At the same time, UAVs limited by endurance time and detection range need to plan resources and paths during application.

Note that data in UAV-enabled edge networks contain sensitive and private information [5]. The security of data transmission in the open wireless environment is of critical importance for the wide deployment and acceptance of edge services in the future [6, 7]. Traditional methods to address wireless communication security are based on cryptography. It may cause high key management costs and computational complexity. The idea of physical layer security is to use channel information to enhance transmission security, which is an effective supplement to the upper layer security. Some scholars devoted to UAV-aided secure cooperation research [8, 9]. They considered a UAV a mobile relay or a friendly jammer to achieve secure transmission. This scenario is similar to a relay network with cooperative jamming. Essentially, the more general model in UAV-enabled secure communication includes a UAV aerial base station (ABS) and numerous mobile users [10]. A UAV can actively send signals instead of just forwarding information. In this case, an enabled mobile base station leads to changeable channel states, which may complex cooperative jamming design. Thus, it is a crux to design a feasible cooperative jamming scheme for secure communication in a mobile UAV-ABS scenario.

For secure communication of UAV-ABS scenarios, several techniques have been introduced to achieve positive secrecy rates or low secrecy outage probability [11]. In [12], the authors intended to jointly optimize the UAV's trajectory, transmit power, and power allocation of jamming signals to achieve the maximum average secrecy rate (ASR) for a UAV-enabled communication system. For a multi-UAV scenario, the authors in [13] used the orthogonal frequency division multiple access (OFDMA) technology to assign idle UAVs to send artificial noise (AN) to achieve secure transmission.

Motivated by the existing works, we discuss secure downlink transmission for a multiantenna UAV-ABS in this paper. In addition, inspired by [14], our system model adopts a hybrid probability channel, including a LOS link and a non-line-of-sight (NLOS) link. In this scenario, we intend to find the ASR. The main contributions of the paper are summarized as follows.

(i) Considering a UAV-ABS edge network with the random subcarrier selection-orthogonal frequency division multiplexing-direction modulation (RSCS-OFDM-DM) technology, we propose an ASR maximization problem to jointly optimize a feasible three-dimensional (3D) trajectory and power allocation

(ii) We exploit the successive convex approximation (SCA) scheme and the alternating optimization (AO) method to design our iterative algorithm to solve the initial nonconvex problem

(iii) We provide numerous simulation results, including the optimal flight trajectory, ASR, and average transmit power to evaluate our secure transmission method

The rest of the paper is organized as follows. The related work is presented in the next section. Next, we provide the system model and problem formulation and following propose an efficient iteration algorithm to solve our average secrecy rate maximization problem. Finally, we discuss and evaluate the performance of our method and conclude the article in the last two sections.

2. Related Work

2.1. Secure Transmit for a UAV Network. Due to the flexibility of UAV-enabled networking, many scholars have studied secure data transmission methods as shown in Table 1. In [15], Wu et al. studied energy-saving secure communication for a downlink A2G link. They discussed the impact of a jitter UAV on secure performance and formulated a power minimization and allocation problem with security constraints. Then, they further used secrecy coverage probability (SCP) and ergodic secrecy capacity (ESC) to investigate how to exploit the UAV jitter feature to enhance secrecy in [16]. Considering various service requirements, the authors in [17] proposed an achievable minimum secrecy rate maximization problem and designed a joint optimization method based on user scheduling, power allocation, and trajectory. In addition, [18] considered UAVs networks with downlink millimeter-wave direction modulation simultaneous wireless information and power transfer under nonorthogonal multiple access (NOMA) and orthogonal multiple access schemes. The authors derived secure outage probability (SOP) and effective secure throughput (EST) to measure security and reliability. In [19], Shengnan et al. proposed a secure transmission scheme of UAVs relay-assisted cognitive radio network (CRN), which optimized UAV relay's flight trajectory and transmit power to maximize secrecy rate. In [20], the authors considered a scenario of GPS interference and Eve covert operation. They proposed a robust joint optimization problem of UAV jamming power and trajectory to maximize the average security rate without completed information of UAV receiver and Eve.

However, the above work does not discuss the misuse of friendly jamming signals due to variable flight altitudes and unreasonable jammer configurations. In [21, 22], the authors introduced a RSCS-OFDM-DM technology to efficiently utilize multiple antennas to reduce unnecessary resource deployment. The use of the technology can reduce the impact on system secrecy performance when the eavesdropper and the legitimate user are in the same beam direction and reduce the complexity of the receiver through fast Fourier transformation.

2.2. Optimization in MEC Networks. There are rich literatures on MEC resource management that aims at optimizing

Table 1: Secure transmission for a UAV-enabled network.

References	Metrics	Contributions
[15]	Secrecy rate	Design a joint optimization to transmit confidential signals and artificial noise to achieve secrecy requirements.
[16]	SCP and ESC	Propose a UAV-jitter-based method to enhanced secrecy performance for an A2G wiretapping scenario.
[17]	Secrecy rate	Use a UAV-ABS with the NOMA technology to develop a secure multiuser data transmit scheme.
[18]	SOP and EST	Design a direction modulation (DM) scheme for a mmWave UAV network with the simultaneous wireless information and power transfer (SWIPT) technical.
[19]	Secrecy rate	Propose a secure data transmission scheme for a UAV relay-assisted CRN to improve spectrum utilization and communication secrecy rate.
[20]	ASR	Use a block coordinate descent method to jointly optimize jamming power and trajectory to maximize ASR.

system latency [23, 24], energy consumption [25, 26], and overall cost of system latency and/or energy consumption [27, 28]. In [23], Ren et al. investigated the joint communication and computation resource allocation problem under the cooperation of cloud computing and edge computing to minimize system delay of all mobile devices. Park et al. in [24] proposed a Cloud-Ran architecture for cloud computing and local edge node collaborative computing. They intend to minimize end-to-end delay by jointly optimizing computing and communication resources. Zhang et al. [25] studied the problem of task unloading and resource allocation in dense network Cloud-Ran architecture to optimize network energy efficiency. Then, Zhou et al. designed a double deep Q network (DDQN-)-based method to joint optimize computation offloading and resource allocation in a dynamic multiuser MEC system in [26]. Their objective is to minimize the energy consumption of the entire MEC system by considering the delay constraint and the uncertain resource requirements of heterogeneous computation tasks. In [27], we studied the joint task unloading and resource allocation of MEC in NOMA-based HetNets. And Dai et al. designed an optimized computing offload and resource allocation strategy using DRL based on 5G beyond terminal edge cloud coordination network to minimize system energy consumption in [28].

Considering the flexible deployment of UAVs, some works introduced UAVs into MEC networks and discussed optimization issues for a UAV-enabled MEC network. In [29], the authors proposed a secure communication scheme for a dual UAV-MEC system. They optimized the resource and trajectories of UAV servers to maximize secure computing power. In [30], the authors studied a secure transmission problem for dual UAV-assisted MEC systems. One UAV is called to help ground terminals (GTs) calculate unloading tasks, and the other UAV acts as a jammer to suppress malicious eavesdroppers. By jointly optimizing the communication resources, computing resources, and UAV's trajectories, they discussed the maximization problem of the minimum secure capacity in time division multiple access and NOMA scenarios. In addition, [31] proposed an innovative UAV-MEC system that involved an interaction between IoT devices, UAVs, and edge clouds. UAVs and edge clouds in the system cooperate in providing MEC services for a group of IoT devices. By jointly optimizing UAV location, communication, computing resource allocation, and task segmentation decision, the weighted sum of service delay of IoT devices and UAV energy consumption is minimized.

Note that these studies assumed wired or dedicated wireless links with sufficient bandwidth among edge nodes deployed in a fixed paradigm. Yet, the assumption is not suitable for data secure transmit in the existing MEC, especially in massive edge users or sparse distribution of network facilities scenarios [32]. Accordingly, we intend to study secure data transmission during task offloading in a UAV-assisted edge network.

3. System Model

Considering flexibility and low-cost features of UAVs, they are very suitable as edge servers to provide signal coverage and information forward for edge terminals. In an UAV-ABS edge network, a UAV can be regarded as an edge server, and ground terminals are edge nodes, as shown in Figure 1. When communicating with a UAV, the edge node is vulnerable to wiretapping by an eavesdropper. This may cause private information disclosure and reduce the secure level of an edge network. In order to ensure secure data transmission, we intend to design a novel scheme to solve the security issue in this scenario.

3.1. A UAV-ABS Network Model. According to Figure 1, a typical UAV-ABS network model includes a UAV source with a N_T-element linear antenna array, a legitimate ground terminal (GT) with a single antenna, and a ground eavesdropper (Eve) with a single antenna. We assume that positions of GTs and the eavesdropper are known and static, which are defined as $\mathbf{U} = (x_U, y_U, 0)$ and $\mathbf{E} = (x_E, y_E, 0)$, respectively. To simplify the optimization problem, the UAV flight duration from the initial position to the final position is T time, and it can discretize into N time slots, where $T = N\delta_t$ and δ_t is the fixed length of a transmission time slot. Assuming δ_t is small enough, the flight in each time slot can be regarded as a uniform motion. Therefore, in time slot $n(n \triangleq 1, 2, \cdots, N)$, the coordinates of a UAV can be denoted as $\mathbf{L}[n] \triangleq (x[n], y[n], h[n])$. The UAV can exploit

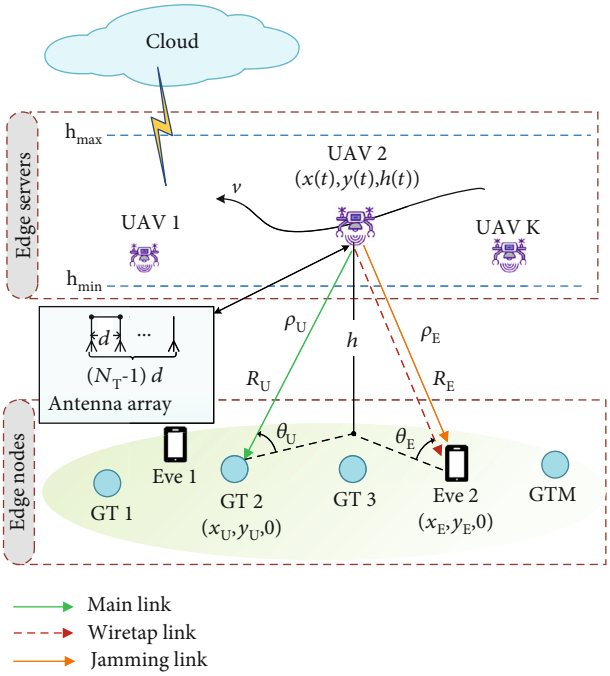


FIGURE 1: A downlink UAV-ABS model.

its multiple antennas to broadcast confidential signals and sends AN directly to the eavesdropper via the RSCS-OFDM-DM technology [21].

In the system, the UAV uses its linear antenna array to send OFDM symbols to the target GTs by randomly selecting multiple subcarriers from the subcarrier set. Supposing that there are N_s orthogonal subcarriers [21], the set is defined as follows:

$$S_s = \{f_m | f_m = f_c + m\Delta f\}, (m = 0,\cdots,N_s - 1)\}, \quad (1)$$

where m is the number of subcarriers and Δf is the bandwidth of a subchannel. In this paper, we assume that the total bandwidth of subcarriers is much less than the center carrier frequency, i.e., $N_s\Delta f \ll f_c$. The signals transmitted by the kth $(k = 1,\cdots,N_T)$ antenna at time slot n can be defined as follows:

$$S_k(n) = \sqrt{\alpha_1[n]P_s}xe^{j\phi_k} + \sqrt{\alpha_2[n]P_s}w_k, \quad (2)$$

where x is confidential signals with $\mathbb{E}\{x^*x\} = 1$, ϕ_k is the initial vector of the kth antenna, and w_k is the artificial noise. The transmit power at time slot n is P_s, and $\alpha_1[n]$ and $\alpha_2[n]$ are the distribution ratio of transmit power to confidential signals and AN, respectively.

The above signals with AN are emitted to the open wireless channel. We assume that the wireless channel experiences a small-scale Rayleigh fading and a large-scale path loss. This channel consists of a LOS link and a NLOS one, where $P_L + P_N = 1$. The probability of the LOS link is related to the elevation angle θ and that of time slot n is defined as follows:

$$P_L(\theta[n]) = \frac{1}{1 + a\exp\left[-b((180^\circ/\pi)\theta[n] - a)\right]}, \quad (3)$$

where a and b are two constants, depending only on the wireless environment, which are provided in work [33]. Similarly, the probability of the NLOS link is denoted as follows:

$$P_N(\theta[n]) = 1 - P_L(\theta[n]). \quad (4)$$

By combining these two links, the expected channel power gain can be computed as follows:

$$|\rho(\theta[n])|^2 = \frac{\beta_0\eta_L P_L(\theta[n])}{R[n]^{\alpha_L}} + \frac{\beta_0\eta_N P_N(\theta[n])}{R[n]^{\alpha_N}}, \quad (5)$$

where $\beta_0 \triangleq 20\log_{10}(C/4\pi d_0 f_c)$ is the path loss at a reference distance $d_0 = 1$ meter. $R[n]$ is the distance of the first antenna of the UAV and the target receiver in time slot n. α_L and α_N denote the pass loss exponents for the LOS and NLOS links. η_L and η_N are the excessive path loss coefficients for the LOS and NLOS links.

Considering $R[n]$, we further define the distance between the kth antenna of the UAV, and the receiver is $R_k[n] = R[n] - (k-1)d\cos(\theta[n])$, where $d = C/2f_c$ is the distance between the antennas. Thus, the reference phase of $R[n]$ is $\varphi_0(R[n]) = 2\pi f_c(R[n]/C)$, and the phase shifting of the kth antenna is computed as follows:

$$\psi_k(\theta[n], R_k[n]) = \frac{2\pi(f_c + k_n\Delta f)R_k[n]}{C} - \varphi_0(R[n]), \quad (6)$$

where k_n is the serial number of the randomly selected subcarrier, corresponding to m.

In Figure 1, we assume that the elevation angle of the legitimate GT and the distance between the first reference antenna of the UAV and the GT are $(\theta_U[n], R_U[n])$. Similarly, these parameters of eavesdropper are $(\theta_E[n], R_E[n])$. For the GT, the received signals synthesized by all array antennas can be expressed as follows:

$$\begin{aligned} y(\theta_U[n], R_U[n]) &= \rho(\theta_U[n])\sqrt{\alpha_1[n]P_s}x \\ &\quad + \rho(\theta_U[n])\sqrt{\alpha_2[n]P_s}\mathbf{h}^H\mathbf{w} + \sum_{k=1}^{N_T} n_0, \end{aligned} \quad (7)$$

where $\mathbf{h}$ is a channel vector from UAV to the GT; i.e.,

$$\mathbf{h} = \frac{1}{\sqrt{N_T}}\left[e^{j\psi_1(\theta_U[n],R_U[n])},\cdots,e^{j\psi_{N_T}(\theta_U[n],R_U[n])}\right]^T. \quad (8)$$

And $n_0 \sim \mathcal{CN}(0,\sigma^2)$ is the additive white Gaussian noise. $\mathbf{w}$ is the artificial noise that is designed in the zero space of $\mathbf{h}$, i.e., $\mathbf{w} = (\mathbf{I}_{N_T} - \mathbf{h}\mathbf{h}^H)\mathbf{z}$.Here, $\mathbf{z}$ is a vector

1: Input
$\mathbf{P}^{(0)}$: the initial power allocation coefficients.
$\mathbf{Q}^{(0)}$: the initial UAV 3D trajectory.
ξ_U^0: a slack variable.
2: **Set:** iteration index $i = 1$, the maximum iteration step $L = 20$, and the threshold $\omega = 10^{-3}$
3: **repeat**
4: Compute the optimal P^i by solving (20) with Q^{i-1}.
5: Compute the optimal Q^i by solving (27) with P^i
6: Obtain the current optimal objective function R^i
7: Update iteration index $i = i + 1$
8: **until**$(R - R_{\text{old}})/R_{\text{old}} \le \omega$ or $i > L$
9: **Output** the objective value $R_{\text{old}} = R^i$

Algorithm 1: The proposed hybrid iteration algorithm.

Table 2: Simulation parameters.

Parameters	Values
Number of antennas, N_T	4
Carrier frequency, f_c	3 GHz [21]
Bandwidth of subchannel, Δf	50 kHz
Channel gain, β_0	−42 dB
Duration of each time slot, δ_t	0.5 s [29]
UAV maximum velocity, $v_{\max}$	20.6 m/s [17, 35]
UAV minimum flight altitude, $h_{\min}$	100 m
UAV maximum flight altitude, $h_{\max}$	200 m
UAV maximum average transmit power, P_s	30 dBM [29]
Noise power, σ^2	−110 dBm [29]
Channel environment coefficients, a, b	20, 0.2 [33]
LOS link excess path loss coefficient, η_{L}	−2.14 dB [15]
NLOS link excess path loss coefficient, η_{N}	−3.14 dB [15]
LOS link path loss exponent, α_{L}	2
NLOS link path loss exponent, α_{N}	3

composed of N_T independent and identically distributed (i.i.d.) circular symmetric complex Gaussian random variables with zero mean and unit variance. It satisfies the distribution of $\mathbf{z} \sim \mathcal{CN}(0, \mathbf{I}_{N_T})$. Thus, $\mathbf{h}^H\mathbf{w} = 0$ holds. The received signals of the GT can be rewritten as follows:

$$y(\theta_U[n], R_U[n]) = \rho(\theta_U[n])\sqrt{\alpha_1[n]P_s}x + \sum_{k=1}^{N_T} n_0. \tag{9}$$

Similarly, the received signals of the eavesdropper is defined as follows:

$$\begin{aligned} y(\theta_E[n], R_E[n]) &= \rho(\theta_E[n])\sqrt{\alpha_1[n]P_s}x \\ &\quad + \rho(\theta_E[n])\sqrt{\alpha_2[n]P_s}\boldsymbol{h_E}^H\boldsymbol{w} + \sum_{k=1}^{N_T} n_0, \end{aligned} \tag{10}$$

where $\mathbf{h_E}$ is a channel steering vector from the UAV to the eavesdropper. Then, we can derive the signal to interference plus noise ratio of the GT and eavesdropper as follows:

$$\gamma_U[n] = \frac{|\rho(\theta_U[n])|^2\alpha_1[n]P_s}{N_T\sigma^2}, \tag{11}$$

$$\gamma_E[n] = \frac{|\rho(\theta_E[n])|^2\alpha_1[n]P_s}{|\rho(\theta_E[n])|^2\alpha_2[n]P_s\|\mathbf{h_E}^H\mathbf{w}\|^2 + N_T\sigma^2}. \tag{12}$$

3.2. UAV Models. In the system, we consider the mobile UAV can fly in 3D space. And the distance of the flight in the nth time slot, $\mathbf{D}[n]$, satisfies the following constraints:

$$\|\mathbf{D}[n]\| = \|\mathbf{L}[n] - \mathbf{L}[n-1]\| \le v_{\max}\delta_t, n = 1, \cdots, N, \tag{13}$$

where $v_{\max}$ is the maximum flying speed of a UAV and $\|\cdot\|$ denotes the Euclidean norm of a vector. In addition, to avoid collisions with buildings and ensure effective communication links during flight, the flying height of a UAV should meet a height constraint, i.e.,

$$h_{\min} \le h[n] \le h_{\max}, n = 1, \cdots, N, \tag{14}$$

where $h_{\min}$ and $h_{\max}$ are the lowest and the highest height of a UAV.

For the mobile UAV, we introduce an AN projection matrix to the null space of a channel steering vector of confidential signals. In this case, signals received by the eavesdropper will have a phase offset. Thus, the eavesdropping correctness will be decreased. During the flight, if the total power of a UAV is P_{tot}, the transmit power at time slot n is computed as follows:

$$P_s \triangleq P_{\text{tot}}/N. \tag{15}$$

And the power allocation of confidential signals and AN is as follows:

$$\alpha_1[n] + \alpha_2[n] = 1, \alpha_1[n], \alpha_2[n] \ge 0, \forall n \in N. \tag{16}$$

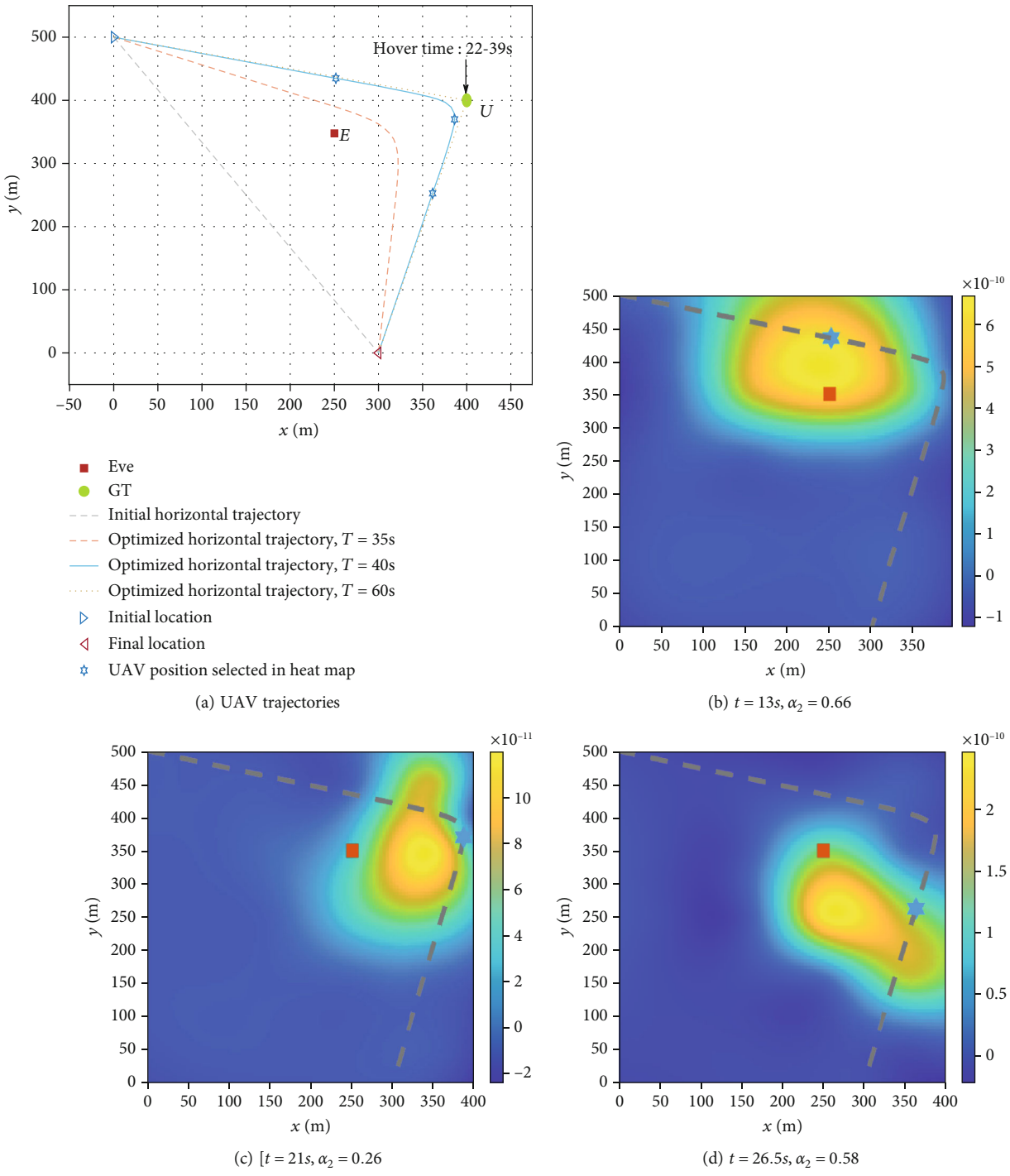


Figure 2: Horizontal trajectories analysis and jamming temperatures at different positions with $T = 40\,s$.

In the next section, we expect to find the optimal power distribution ratio by changing transmit power.

3.3. Problem Formulation. In order to realize secure communication of a UAV-ABS network, we exploit the RSCS-OFDM-DM technology to maximize the average secrecy rate by jointly optimizing UAV trajectory and the UAV power allocation ratio. The maximization problem is as follows:

$$\begin{aligned} \max_{\mathbf{Q},\mathbf{P}} \quad & SR \\ \text{s.t.} \quad & (13), (14), \text{and } (16), \end{aligned} \tag{17}$$

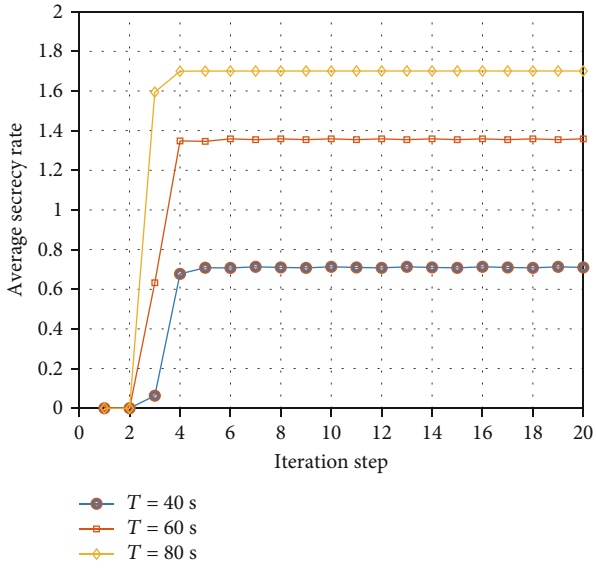


Figure 3: Convergence analysis.

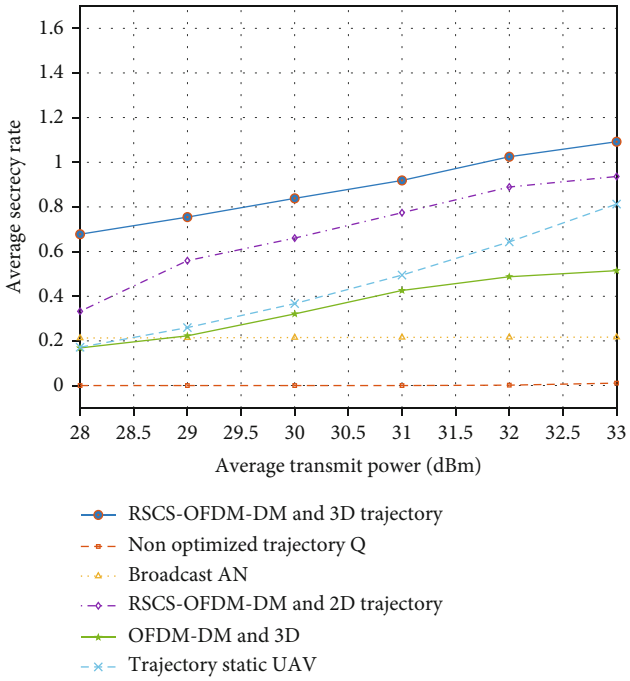


Figure 4: ASR *vs* transmit power.

where $\mathbf{Q} \triangleq \{\mathbf{L}[n] \in \mathbb{R}^{3\times 1} | \forall n\}$ represents the position set of a UAV, $\mathbf{P} \triangleq \{\alpha_i[n] \in \mathbb{R} | i \in \{1, 2\}, \forall n\}$ is the set of power allocation coefficients, and the objective function is defined as follows:

$$SR = \frac{1}{N}\sum_{n=1}^{N}\left[\log_2(1+\gamma_U[n]) - \log_2(1+\gamma_E[n])\right]^+, \tag{18}$$

where $[a]^+ = \max\{a, 0\}$; when the secrecy rates are favorable, UAVs and legitimate ground nodes can communicate normally.

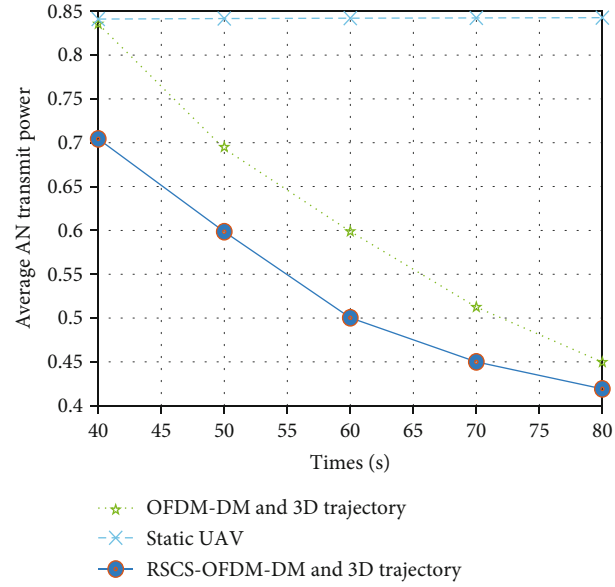


Figure 5: Average AN power.

4. Maximal ASR Scheme Design

Solving problem (17) directly is challenging due to its nonconvexity. We use AO and SCA methods to find the approximate optimal solution of (17). In particular, we study the optimization problem from two perspectives, i.e., UAV transmit power allocation and UAV 3D trajectory design during each flight time slot.

4.1. Power Allocation Subproblem. In this subproblem, we assume that the UAV source 3D trajectory is predefined. We introduce two slack variables, τ and $\mathbf{u}$, into (17). Then, the UAV transmit power allocation subproblem is formulated as follows:

$$\max_{\boldsymbol{P},\tau,\boldsymbol{u}} \tau \tag{19a}$$

$$\text{s.t.} \frac{1}{N}\sum_{n=1}^{N}\{\log_2(1+A_1[n]\alpha_1[n]) - \mu[n]\} \geq \tau, \tag{19b}$$

$$\log_2\left(1+\frac{A_2[n]\alpha_1[n]}{A_3[n](1-\alpha_1[n])+1}\right) \leq \mu[n], \tag{19c}$$

$$\text{and } (16),$$

where $\mathbf{u} \triangleq \{\mu[n] \in \mathbb{R} | n \in N\}$ is the upper boundary of eavesdropper's transmit rate, $A_1[n] = |\rho(\theta_U[n])|^2 P_s/N_T\sigma^2$, $A_2[n] = |\rho(\theta_E[n])|^2 P_s/N_T\sigma^2$, and $A_3[n] = |\rho(\theta_E[n])|^2 P_s \|\boldsymbol{h_E}^H \boldsymbol{w}\|^2/N_T\sigma^2$. Because the UAV trajectory is predefined, $A_1[n]$, $A_2[n]$, and $A_3[n]$ can be regarded as constants. Yet, the constraint (19c) is still nonconvex. Then, we further transform it based on the first-order Taylor approximation method; i.e., $g(x) = g(x^*) + g'(x^*)(x - x^*)$, where x^* is the result of

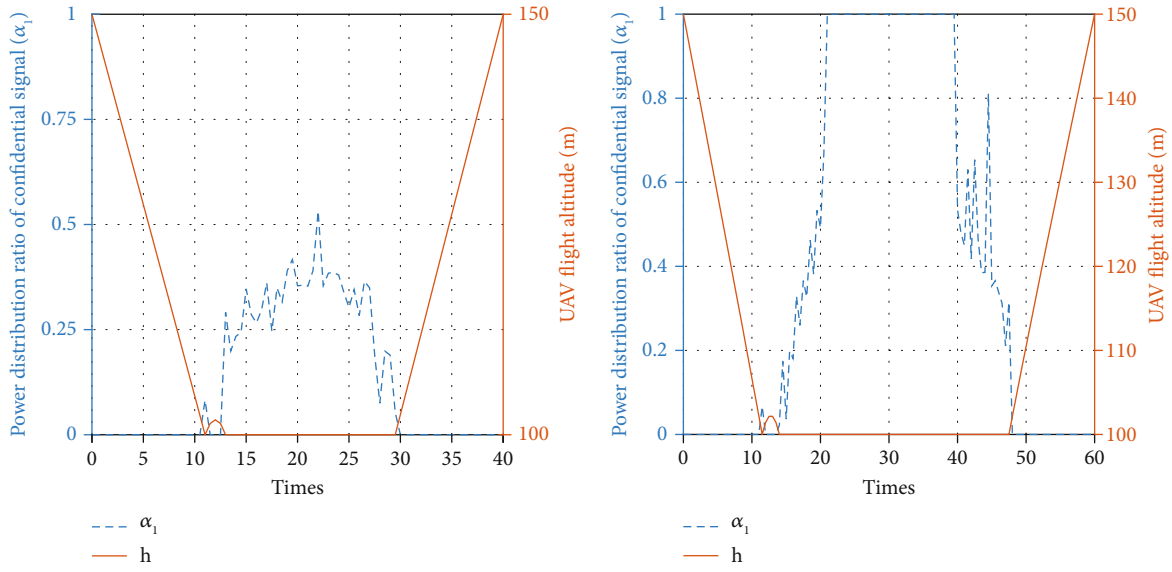


FIGURE 6: The altitude and power allocation coefficient.

the previous iteration x. Then, (19c) can be rewritten as follows:

$$\begin{aligned} \max_{\mathbf{P},\tau,\mathbf{u}} \quad & \tau \\ g(\alpha_1[n]) \le \mu[n], \quad & \\ \text{s.t.} \quad & (16) \text{ and } (19b). \end{aligned} \tag{20}$$

We can exploit CVX toolbox to solve the subproblem since it satisfies the convex optimization requirements.

4.2. UAV 3D Trajectory Subproblem. When the power allocation coefficients are known, we still introduce two slack variables, $\zeta_{\mathbf{U}}$ and $\zeta_{\mathbf{E}}$, to simplify the original problem to the UAV 3D trajectory subproblem as follows:

$$\max_{\mathbf{Q},\zeta_{\mathbf{U}},\zeta_{\mathbf{E}}} \frac{1}{N}\sum_{n=1}^{N} SR[n], \tag{21a}$$

$$\text{s.t.} \xi_U[n] \ge \|\boldsymbol{L}[n] - \boldsymbol{U}\|^2, \tag{21b}$$

$$\xi_E[n] \le \left\|\boldsymbol{L}[n] - \boldsymbol{E}^2\right\|, \tag{21c}$$

$$\xi_U[n] \ge h_{\min}^2, \tag{21d}$$

where $SR[n] = \log_2(1 + D_1[n]|\rho(\theta_U[n])|^2) - \log_2(1 + (D_1[n]|\rho(\theta_E[n])|^2/D_2[n]|\rho(\theta_E[n])|^2 + 1))$, $\zeta_{\mathbf{U}} \triangleq \{\xi_U[n]|\forall n\}$, $\zeta_{\mathbf{E}} \triangleq \{\xi_E[n]|\forall n\}$, $D_1[n] = \alpha_1[n]P_s/N_T\sigma^2$, and $D_2[n] = \alpha_2[n]P_s\|\boldsymbol{h_E}^H\boldsymbol{w}\|^2/N_T\sigma^2$. Because the flight time slot of the UAV is tiny, we assume that the elevation change before and after trajectory iteration is small. The channel power gains of the legitimate GT and the eavesdropper can be rewritten as follows:

$$|\rho_U[n]|^2 = \frac{\eta_{\mathrm{L}}P_{\mathrm{L}}(\theta_U^*[n])\beta_0}{\xi_U[n]^{\alpha_{\mathrm{L}}/2}} + \frac{\eta_{\mathrm{N}}P_{\mathrm{N}}(\theta_U^*[n])\beta_0}{\xi_U[n]^{\alpha_{\mathrm{N}}/2}}, \tag{22}$$

$$|\rho_E[n]|^2 = \frac{\eta_{\mathrm{L}}P_{\mathrm{L}}(\theta_E^*[n])\beta_0}{\xi_E[n]^{\alpha_{\mathrm{L}}/2}} + \frac{\eta_{\mathrm{N}}P_{\mathrm{N}}(\theta_E^*[n])\beta_0}{\xi_E[n]^{\alpha_{\mathrm{N}}/2}}. \tag{23}$$

Then, the objective function can be rewritten as follows:

$$\widetilde{SR}[n] = \log_2\left(1 + D_1[n]|\rho_U[n]|^2\right) - \log_2\left(1 + \frac{D_1[n]|\rho_E[n]|^2}{D_2[n]|\rho_E[n]|^2 + 1}\right). \tag{24}$$

Note that the constraint (21c) is still a nonconvex constraint on $\mathbf{L}[n]$, which causes the subproblem to be nonconvex. We use the SCA method to transform this nonconvex subproblem into a convex one. We assume that the coordinate of the previous iteration is $\mathbf{L}^*[n]$. Since the first-order Taylor expansion of a convex function at one point is its lower bound, the constraints (21c) and $\widetilde{SR}[n]$ can be approximates as follows:

$$\xi_E[n] \le 2(\mathbf{L}^*[n] - \mathbf{E})^T(\mathbf{L}[n] - \mathbf{L}^*[n]) + \|\mathbf{L}^*[n] - \mathbf{E}\|^2, \tag{25}$$

$$\widetilde{SR}[n] \ge \widehat{SR}^*[n] = g(\xi_U[n]) - \log_2\left(1 + \frac{D_1[n]|\rho_E[n]|^2}{D_2[n]|\rho_E[n]|^2 + 1}\right), \tag{26}$$

where $g(\xi_U[n])$ is also the first-order Taylor expansion of the first term in (24). Accordingly, the subproblem (21a) is approximately equivalent to the following convex problem:

$$\max_{\mathbf{Q},\zeta_U,\zeta_E} \quad \frac{1}{N}\sum_{n=1}^{N}\widehat{SR}^*[n] \tag{27}$$
$$\text{s.t.} \quad (13), (14), (21b), (21d), \text{and } (25).$$

4.3. Hybrid Iteration Algorithm. We design a hybrid iteration algorithm to solve the problem (17). In the ith iteration, we obtain the optimal transmit allocation ratio $\mathbf{P}^i$ with a given UAV 3D trajectory $\mathbf{Q}^{i-1}$ by solving subproblem (20). Next, the optimal UAV 3D trajectory is $\mathbf{Q}^i$ with a given transmit allocation ratio of $\mathbf{P}^i$ by solving the subproblem (27). The details of our algorithm are summarized in **Algorithm**1.

In each iteration of Algorithm 1, two convex subproblems (20) and (27) are solved by SCA algorithm. We define that the number of UAVs is K, the number of GTs is M, and the number of time slots is N. The number of optimization variables in (20) is only related to K and N. If the number of iterations is assumed to be L_1, the computational complexity of (20) can be calculated as $\mathcal{O}(L_1K^3N^3)$ [34]. Similarly, the number of optimization variables in (27) is related to K, M, and N. And the computational complexity of (27) is $\mathcal{O}(L_2K^3M^3N^3)$ when the number of iterations is L_2. Therefore, the computational complexity of **Algorithm1** is $\mathcal{O}(L(L_1K^3N^3 + L_2K^3M^3N^3))$, where L is the iteration number of Algorithm 1.

5. Simulation and Discussion

In this section, we evaluate the performance of our proposed scheme through numerical simulation. Unless otherwise specified, simulation parameters are shown in Table 2.

In Figure 2, we first discuss the optimal trajectory of the UAV and the corresponding interference temperature. According to Algorithm 1, we find the optimal horizontal trajectory of the UAV under flight time $T = 35$ s, 40 s, and 60 s, as shown in Figure 2(a) It is found that as flight time increased, the UAV preferred to stay closer to the GT and away from the eavesdropper to increase the ASR. In particular, the UAV will hover over the legitimate GT as long as possible to increase the ASR when $T = 60$ s. Figures 2(b)–2(d) demonstrate that the ground eavesdropper is subject to intense interference temperature at three different positions of UAV's flight trajectory. In Figure 2(c), $\alpha_2 = 0.26$, the transmitted AN is relatively small, so the order of magnitude of the interference temperature is smaller than Figures 2(b) and 2(d). Compared with Figure 2(d), the eavesdropper in Figures 2(c) and 2(d) is located at the edge of the interference temperature mass. The interference temperature is constrained by the AN power and the distance between the UAV and the ground eavesdropper. In addition, we find that the interference temperature radiates to the eavesdropper with the UAV as the center in Figure 2(b). Yet, the center of the interference temperature in Figures 2(c) and 2(d) Figure appears between the UAV and the eavesdropper's position with a slight offset. The reason is that AN signals sent by the UAV is affected by GT's positions.

In Figure 3, we discuss the convergence of the proposed algorithm at flight time $T = 40$ s, 60 s, and 80 s. The ASR of the first two iterations is almost zero because the trajectory optimization of the UAV at the beginning of the iteration is similar to the initial trajectory. Then, the ASR increases sharply as the increasing number of iterations and gradually converges to a fixed value after four or five iterations. It shows that the proposed algorithm can effectively converge to the optimal solution. Also, the longer the flight time, the higher ASR after convergence. The reason is that the hovering time of the UAV becomes longer as the flight time increases.

In order to verify the effectiveness of our iterative algorithm, we next compare the ASR performance of different schemes in Figure 4. In the figure, we know that the ASR performances of either two-dimensional (2D) or 3D trajectory optimization are greater than that of static UAV and nonoptimized trajectory schemes. This explains the impact of trajectory optimization on the system secrecy performance. The reason is that the UAV trajectory design can help to obtain a better channel between a UAV and a GT. In addition, the RSCS-OFDM-DM technology has a better jamming effect on an eavesdropper via comparing with OFDM-DM and broadcast AN schemes. Overall, the proposed algorithm can jointly optimize the 3D flight trajectory and transmit signal power to improve physical layer security of the UAV-enabled network. Then, we compare the average transmit AN power of different schemes in Figure 5. It shows that the scheme using both random subcarrier selection and mobile UAV has higher energy efficiency. In the case of the same total transmit power, our scheme can send the smallest average transmit AN power to achieve the same secrecy performance.

Figure 6 provides the impact of the UAV's flight altitude h on power distribution ratio of confidential signals when the flight durition is $T = 40$ s and $T = 60$ s, respectively. It can be seen that h is inversely proportional to α_1. When the UAV is closer to the eavesdropper and farther from the GT, $\alpha_1 = 0$. This means that all transmit power is used to emit artificial noise. As the UAV approaches the GT and the eavesdropper, α_1 increases, while the UAV's altitude decreases. It can improve the quality of the channel between the UAV and the legitimate ground node to ensure secure communication. For the $T = 60$ s case, the UAV hovers over the GT during the period of 22 s to 39 s. In this period, the quality of the legitimate channel is the best. All power is used to transmit confidential signals.

6. Conclusion

In this paper, we study how to improve security of data transmission for a UAV-enabled edge network by jointly optimizing the 3D flight trajectory and power allocation design. First, we formulate an optimization problem by

establishing a UAV communication system model based on the RSCS-OFDM-DM technology. Next, we divide the optimization problem into two subproblems for discussion, i.e., UAV transmit power allocation and UAV 3D trajectory design, and then present a hybrid iterative algorithm to find the optimal solution of the optimization problem. Finally, we compare the secrecy performance of the proposed scheme with other five schemes. Also, we verify that the mobile UAV and random subcarrier selection can improve the secrecy energy efficiency.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 2019JBZ001, in part by the Beijing Natural Science Foundation, under Grant 4202054, in part by the National Natural Science Foundation of China, under Grant 61871023, in part by the Hangzhou Innovation Institute, Beihang University, under Grant 2020-Y5-A-022, and in part by the S&T Program of Hebei, under Grant SZX2020034.

References

[1] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 144–153, Dallas, TX, USA, 2019.

[2] S. Han, X. Xiaodong, S. Fang et al., "Energy efficient secure computation offloading in NOMA-based mMTC networks for IoT," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5674–5690, 2019.

[3] X. Sun, D. W. K. Ng, Z. Ding, X. Yanqing, and Z. Zhong, "Physical layer security in UAV systems: challenges and opportunities," *IEEE Wireless Communications*, vol. 26, no. 5, pp. 40–47, 2019.

[4] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[5] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[6] Y. Huo, Y. Tian, L. Ma, X. Cheng, and T. Jing, "Jamming strategies for physical layer security," *IEEE Wireless Communications*, vol. 25, no. 1, pp. 148–153, 2018.

[7] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[8] W. Wang, J. Tang, N. Zhao et al., "Joint precoding optimization for secure SWIPT in UAV-aided NOMA networks," *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 5028–5040, 2020.

[9] Y. Huo, Y. Tian, H. Chunqiang, Q. Gao, and T. Jing, "Jamming strategies for physical layer security," *Wireless Communications and Mobile Computing*, vol. 2017, Article ID 1832051, 11 pages, 2017.

[10] S. Enayati, H. Saeedi, H. Pishro-Nik, and H. Yanikomeroglu, "Moving aerial base station networks: a stochastic geometry analysis and design perspective," *IEEE Transactions on Wireless Communications*, vol. 18, no. 6, pp. 2977–2988, 2019.

[11] Z. Cai, Z. Xiong, X. Honghui, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks: a survey toward private and secure applications," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–38, 2021.

[12] M. T. Mamaghani and Y. Hong, "Joint trajectory and power allocation design for secure artificial noise aided UAV communications," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 3, pp. 2850–2855, 2021.

[13] R. Li, Z. Wei, L. Yang, D. W. K. Ng, J. Yuan, and J. An, "Resource allocation for secure multi-UAV communication systems with multi-eavesdropper," *IEEE Transactions on Communications*, vol. 68, no. 7, pp. 4490–4506, 2020.

[14] J. Lyu and H.-M. Wang, "Secure UAV random networks with minimum safety distance," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 3, pp. 2856–2861, 2021.

[15] W. Huici, Y. Wen, J. Zhang, Z. Wei, N. Zhang, and X. Tao, "Energy-efficient and secure air-to-ground communication with jittering UAV," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 3954–3967, 2020.

[16] W. Huici, H. Li, Z. Wei, N. Zhang, and X. Tao, "Secrecy performance analysis of air-to-ground communication with UAV jitter and multiple random walking eavesdroppers," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 1, pp. 572–584, 2021.

[17] H.-M. Wang and X. Zhang, "UAV secure downlink NOMA transmissions: a secure users oriented perspective," *IEEE Transactions on Communications*, vol. 68, no. 9, pp. 5732–5746, 2020.

[18] X. Sun, W. Yang, and Y. Cai, "Secure communication in NOMA-assisted millimeter-wave SWIPT UAV networks," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 1884–1897, 2020.

[19] C. Shengnan, J. Xiangdong, G. Yixuan, and Z. Yuhua, "Physical layer security communication of cognitive UAV mobile relay network," in *2021 7th International Symposium on Mechatronics and Industrial Informatics (ISMII)*, pp. 267–271, Zhuhai, China, 2021.

[20] Y. Roh, S. Jung, and J. Kang, "Cooperative UAV jammer for enhancing physical layer security: robust design for jamming power and trajectory," in *MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)*, pp. 464–469, Norfolk, VA, USA, 2019.

[21] T. Shen, S. Zhang, R. Chen et al., "Two practical random-subcarrier-selection methods for secure precise wireless transmissions," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 9018–9028, 2019.

[22] F. Shu, W. Xiaomin, H. Jinsong, J. Li, R. Chen, and J. Wang, "Secure and precise wireless transmission for random-subcarrier-selection-based directional modulation transmit antenna array," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 4, pp. 890–904, 2018.

[23] J. Ren and Y. Guanding, "Collaborative cloud and edge computing for latency minimization," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 5031–5044, 2019.

[24] S.-H. Park, S. Jeong, J. Na, O. Simeone, and S. Shamai, "Collaborative cloud and edge mobile computing in C-RAN systems with minimal end-to-end latency," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 7, pp. 259–274, 2021.

[25] Q. Zhang, L. Gui, F. Hou, J. Chen, S. Zhu, and F. Tian, "Dynamic task offloading and resource allocation for mobile-edge computing in dense cloud RAN," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3282–3299, 2020.

[26] H. Zhou, K. Jiang, X. Liu, X. Li, and V. C. M. Leung, "Deep reinforcement learning for energy-efficient computation offloading in mobile-edge computing," *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 1517–1530, 2022.

[27] X. Chen, G. Zheng, and X. Zhao, "Energy-minimization task offloading and resource allocation for mobile edge computing in NOMA heterogeneous networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 16001–16016, 2020.

[28] Y. Dai, K. Zhang, S. Maharjan, and Y. Zhang, "Edge intelligence for energy-efficient computation offloading and resource allocation in 5G beyond," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 12175–12186, 2020.

[29] L. Weidang, Y. Ding, S. H. Yuan Gao, W. Yuan, N. Zhao, and Y. Gong, "Resource and trajectory optimization for secure communications in dual unmanned aerial vehicle mobile edge computing systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 4, pp. 2704–2713, 2022.

[30] X. Yu, T. Zhang, D. Yang, Y. Liu, and M. Tao, "Joint resource and trajectory optimization for security in UAV-assisted MEC systems," *IEEE Transactions on Communications*, vol. 69, no. 1, pp. 573–588, 2021.

[31] Y. Zhe, Y. Gong, S. Gong, and Y. Guo, "Joint task offloading and resource allocation in UAV-enabled mobile edge computing," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3147–3159, 2020.

[32] T. Salam, W. U. Rehman, and X. Tao, "Data aggregation in massive machine type communication: challenges and solutions," *IEEE Access*, vol. 7, pp. 41921–41946, 2019.

[33] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, 2014.

[34] C. Zhan and Y. Zeng, "Aerial–ground cost tradeoff for multi-UAV-enabled data collection in wireless sensor networks," *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1937–1950, 2020.

[35] DJI, "DJI inspire 2 technical parameters," https://www.dji.com/inspire-2.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4044783, 10 pages
https://doi.org/10.1155/2022/4044783

Research Article

Pilot Allocation and Data Power Optimization Based on Access Point Selection in Cell-Free Massive MIMO

Zhiwen Duan[1] **and Feng Zhao**[2]

[1]*The Key Laboratory of Cognitive Radio and Information Processing, Guilin University of Electronic Technology, Guilin 541004, China*
[2]*The Guangxi Colleges and Universities Key Laboratory of Complex System Optimization and Big Data Processing, Yulin Normal University, Yulin 537000, China*

Correspondence should be addressed to Feng Zhao; zhaofeng@guet.edu.cn

Received 8 February 2022; Accepted 18 March 2022; Published 7 April 2022

Academic Editor: Yingjie Wang

Focusing on the pilot contamination problem of cell-free massive multi-input multi-output (MIMO), a pilot allocation algorithm based on access point (AP) selection is proposed. To improve the system performance further, data power optimization is carried out. First, the AP that serves each user is selected through the user-centered idea and the large-scale fading matrix. Use the same service AP numbers between users to measure the pilot contamination intensity and then assign orthogonal pilots to strong potential pilot contamination users. A spectrum efficiency (SE) scheme to maximize the minimum user is proposed for the fairness problem in data power optimization. It is proved that the objective function of data power optimization accords with linear programming, and a dichotomy is given to solve the problem. Simulation results show that the pilot allocation algorithm based on AP selection can significantly improve the total SE of the system and reduce the computational complexity of the system. At the same time, the data power optimization algorithm improves minimum SE for users in the system.

1. Introduction

With the progress and development of the times, wireless communication has facilitated people's lives, and the amount of wireless data has also increased significantly. For example, wireless sensors collect a large amount of data in the physical world; new intelligent cyberphysical systems (CPS) collect data in different dimensions: the widespread use of the Internet of Things in smart cities and industrial 4.0 [1–6]. In the future, a large amount of wireless data will put forward broader connection requirements for 6G and even higher versions of wireless communication technology. It also put forward higher performance requirements. To meet the requirements of high energy efficiency, high SE, and ubiquitous network connection requirements for the next-generation wireless communication, especially the communication demand of high-density user (UE) scenario, some scholars put forward the concept of cell-free massive multi-input multi-output (CF massive MIMO), and this technology has become one of the critical technologies of 6G [7]. Through the distributed deployment of APs, CF massive MIMO reduces the distance between UE and AP, providing consistent and good service for UE. At the same time, because there is no boundary of cellular network, it avoids the problems that UEs have to switch cell services frequently, and the boundary service quality is poor. It also has the characteristics of channel hardening, strong macrodiversity, and the ability to resist multiuser interference [8–10]. However, when AP acquires channel state information (CSI) through the pilot information sent by UEs, due to many UEs and the limitation of the coherence time, the pilot will be multiplexed among multiple UEs, which leads to the problem of pilot contamination. Pilot contamination is the obstacle and bottleneck to improve the performance of CF massive MIMO [11].

To solve the pilot contamination problem of CF massive MIMO and improve SE, researchers mainly design pilot allocation and data power optimization schemes. References

[7, 12, 13] use prior information, such as user location and large-scale fading matrix, to design pilot allocation methods, while references [14–16] design a power optimization scheme from the aspects of reducing power consumption and improving SE. A greedy pilot allocation algorithm is proposed in reference [7]. It can improve the SE of the worst users. However, the initial random pilot allocation ignores the potential pilot contamination between users, which may not substantially improve the system's performance in subsequent iterations. Reference [12] proposes a pilot allocation algorithm based on tabu search. An iterative algorithm is constructed to avoid optimal local results by defining the domain, searching the objective function, and introducing a taboo list. Finally, it is proved that the pilot allocation method outperforms the random and greedy pilot allocation methods. Reference [13] proposed a pilot allocation scheme based on the Hungarian algorithm. First, several users are selected by the size of the large-scale fading matrix, the number of users chosen is equal to the number of orthogonal pilots, and then these pilots are assigned to users. Then, an optimization problem is constructed, and finally, the Hungarian algorithm is used to solve the problem. In reference [14], an optimization problem of minimizing the total transmission power under the condition of satisfying the user's quality of service is proposed. It is proved that the problem is a linear programming problem, and the optimal global solution can be found in polynomial time. Reference [15] proposed a fractional power control method, which determines the power control coefficient by calculating the ratio of the large-scale fading value of a single user to the sum of the large-scale fading values of all users. This method can effectively suppress the interference between users. In the case that both the user and AP have multiple antennas, the reference [16] adopts the power distribution strategy of sum-rate maximization and minimum rate maximization on the uplink and downlinks, respectively. By solving the power optimization problem on the uplink and downlinks, the total data rate of the uplink and the fairness of the user performance of the downlink are improved, respectively. From the reference [14–19], it can be seen that data power optimization can improve the system's performance, such as improving the fairness among users and improving the system's overall performance. For the pilot allocation problem in reference [7, 12, 13], it is assumed that all AP services to every user will increase the computational complexity of the system. It is necessary to eliminate pilot contamination in CF massive MIMO and consider the practicability of dense users. To solve this problem, with the idea of user-centered, the literature [17, 18] can effectively reduce the computational complexity of the system by selecting part of AP to serve users. Still, it does not consider the pilot contamination problem among users after choosing AP. Therefore, In the CF massive MIMO, it is necessary to perform AP selection, allocate the pilot, and optimize the data power simultaneously.

In this paper, we study the advantages and existing challenges of the CF massive MIMO uplink system and propose a pilot allocation algorithm based on AP selection to suppress the influence of pilot contamination and improve the system's SE. Lastly, optimizing data power improves the minimum SE of users in the system. First of all, by analyzing the large-scale fading matrix between users and AP, the AP serving user is selected based on the large-scale fading matrix and user-centered idea. The user AP service matrix is constructed. Then, based on the AP service matrix, the number of the same service AP between the user is obtained, that is, the AP coincidence degree. Users of the same AP service are divided into the same group for pilot allocation. However, groups with more users will reuse more pilots. Therefore, groups with more users are preferentially selected for pilot allocation so as to use as many mutually orthogonal pilots as possible to reduce interference. The AP coincidence degree measures the potential pilot contamination intensity between UEs for the same group of UEs. The UEs with high pilot contamination intensity prioritize assigning orthogonal pilots. For the problem of fairness, a data power optimization scheme to max-min user's SE is proposed. The data power optimization scheme is proved to belong to the linear programming problem. And the dichotomy is used to solve the problem. Simulation results show that the pilot allocation and data power optimization method based on AP selection can effectively suppress the pilot contamination problem and improve the system's total SE. It also can reduce the computational complexity of the system and improve the fairness between users.

2. System Model

This paper studies the uplink system of CF massive MIMO based on time division duplexing (TDD) mode and the system model (see Figure 1) [19].

This system is mainly composed of M APs, K UEs, and several CPUs; each UE and AP are assumed to be a single antenna. AP is randomly assigned to the covered area, assuming that the UE is at low speed or static state. All APs are connected to the CPU through the backhaul link and perform signal processing in the CPU. In the case of partial AP serving users, the CPU uses the selection information of AP to select the signals collected by partial AP for user channel estimation and signal detection.

Suppose that the total length of each coherent block is $T = \tau_p + \tau_u + \tau_d$, where τ_p is the pilot length, τ_u is the length of the uplink data transmission, and τ_d is the length of data transmission in the downlink. The channel matrix between the k-th user and the m-th AP is expressed by $g_{m,k}$. It is assumed that the channel is a correlated Rayleigh fading channel, then $g_{m,k}$ is the large-scale fading matrix and $g_{m,k} \sim N_{\mathbb{C}}(0, R_{m,k})$.

In the system setting, it is assumed that AP and UE do not have a priori CSI at the beginning of the coherent interval; so, channel estimation is needed in each coherent interval. Therefore, the communication in the uplink includes two stages: uplink pilot transmission and uplink data transmission. In the pilot transmission stage, each UE is assigned a pilot. The received pilot information is used for channel estimation. The estimated channel is used to detect the received data, thereby calculating the SE of each UE.

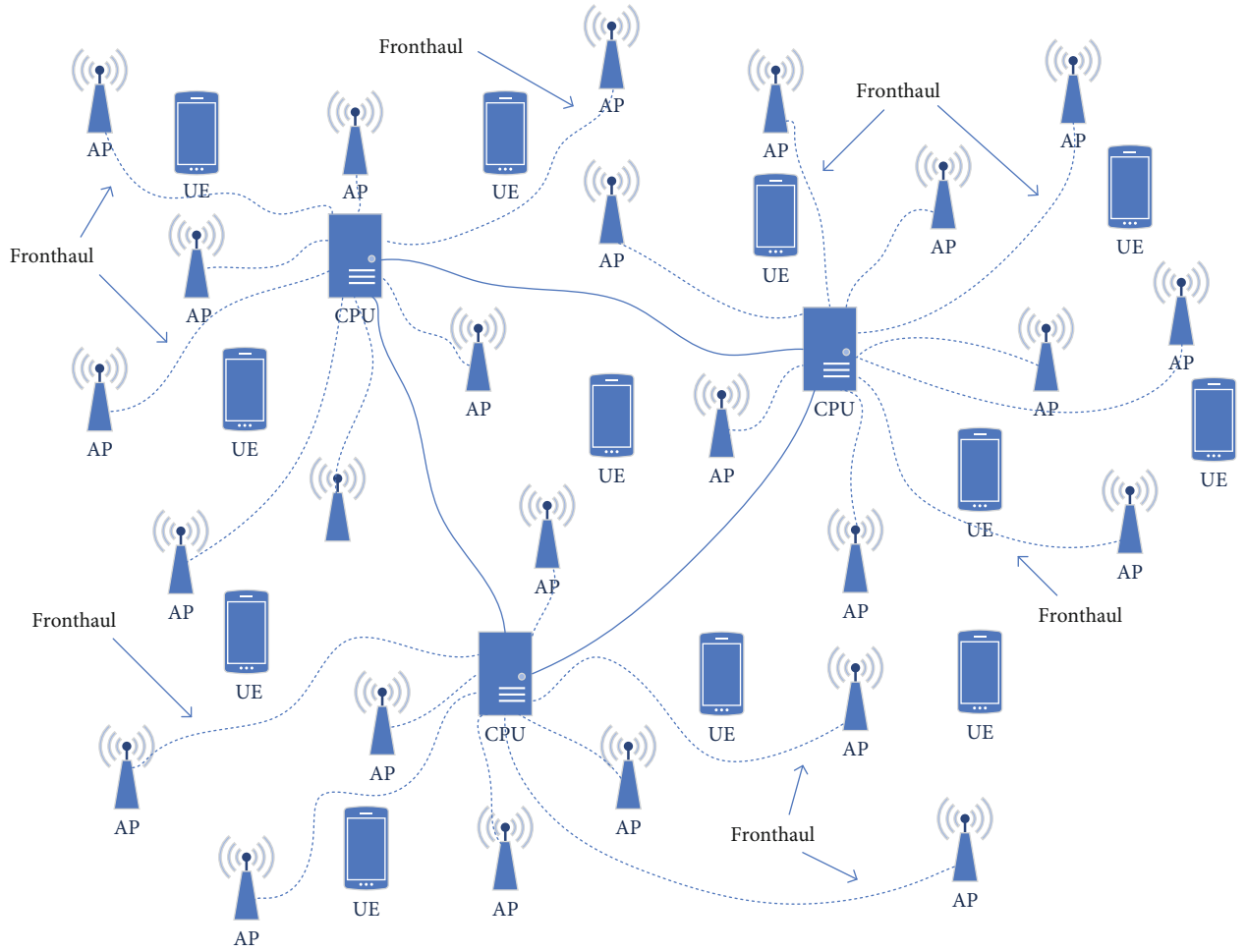


Figure 1: CF massive MIMO network system.

2.1. Uplink Pilot Transmission and Uplink Data Transmission. The set of $\varphi^u = \{\varphi_1, \varphi_2 \cdots \varphi_{\tau_p}\}$ represents all orthogonal pilots, and $\|\varphi_t\|^2 = \tau_p$. In the uplink pilot transmission phase, the pilot signal received by the m-th AP can be expressed as $y_{p,m}$:

$$y_{p,m} = \sum_{j=1}^{K} \sqrt{p_j^p} g_{m,j} \varphi_{t_j}^{\mathrm{T}} + \mathbf{N}_m, \tag{1}$$

where p_j^p is the pilot transmission power of user j, and $\mathbf{N}_m$ represents the noise received by the m-th AP. Using minimum mean squared error (MMSE) to estimate the channel, then the estimated channel between the k-th user and the m -th AP $\widehat{g}_{m,k}$ [20] is

$$\widehat{g}_{m,k} = \sqrt{p_k^p \tau_p} \mathbf{R}_{m,k} \Psi_{t_k,m}^{-1} y_{p,m\, t_k}, \tag{2}$$

where

$$\Psi_{t_k,m} = \sum_{j \in B_t} \tau_p p_j^p \mathbf{R}_{m,j} + \sigma^2 \mathbf{I}, \tag{3}$$

where B_t represents the set of users who use pilot t in the pilot transmission phase. From Equation (3), it can be seen that there will be pilot contamination among users who reuse the same pilot.

In the uplink data transmission phase, similar to the pilot transmission phase, then the data signal received by the m-th AP can be expressed as $y_{u,m}$:

$$y_{u,m} = \sum_{j=1}^{K} \sqrt{p_j^u} g_{m,j} x_j + \mathrm{N}_m, \tag{4}$$

where x_j is the data signal sent by user j, and p_j^u is the data transmission power of user j. Before the AP selection is made, it is necessary to combine all the signals received by the AP to detect the data. Channel estimation, detection matrix, and data detection all adopt MMSE. The achievable SE is [19, 20]

$$\mathrm{SE}_k^{(ul,1)} = \frac{\tau_p}{T} \mathbb{E}\left\{\log_2\left(1 + \mathrm{SINR}_k^{(ul,1)}\right)\right\}, \tag{5}$$

where

$$\mathrm{SINR}_k^{(ul,1)} = p_k^u \widehat{g}_k^{\mathrm{H}} \left(\sum_{j=1, j\neq k}^{K} p_j^u \widehat{g}_j \widehat{g}_j^{\mathrm{H}} + \sum_{j=1}^{K} p_j^u \mathbf{C}_j + \sigma^2 \mathbf{I}_M \right)^{-1} \widehat{g}_k. \tag{6}$$

3. Pilot Assignment

3.1. AP Selection. For users, the AP located near the user contributes the maximum SE. In contrast, the AP, which is far from the UE, has less gain in macrodiversity. The potential pilot contamination among users who assign the same pilot is mainly related to its large-scale fading matrix. Therefore, AP is selected with the help of a user-centered idea and each user's large-scale fading matrix [21, 22]. To sum up, the AP selection formula for the k-th user is

$$\sum_{m=k^{(1)}}^{k^{(Q)}} \frac{\bar{R}_{m,k}}{\sum_{m'=1}^{M} R_{m',k}} \geq \beta\%, \tag{7}$$

where $\{k^{(1)}, k^{(2)} \cdots, k^{(Q)}\}$ represents the Q APs selected by user k $(Q \leq K)$, the descending set of the large-scale fading matrix between the UE and the AP is $\{\bar{R}_{k^{(1)},k}, \bar{R}_{k^{(2)},k}, \cdots, \bar{R}_{k^{(Q)},k}\}$, and β represents a set constant. Assuming that the set of Q APs selected by the k-th user is represented by A_k, the corresponding AP service matrix $D_{k,m}$ is defined as

$$D_{k,m} = \begin{cases} 1 & \text{if m} \in A_k, \\ 0 & \text{if m} \notin A_k. \end{cases} \tag{8}$$

The position where the service matrix $D_{k,m} = 1$ represents that the m-th AP serves the k-th UE. Through the service matrix, the AP coincidence matrix $B \in \mathbb{C}^{K\times K}$ between two users and the matrix $d \in \mathbb{C}^{1\times M}$ of the number of AP service users can be obtained, which is defined as follows:

AP coincidence matrix is as follows:

$$B_{k,k'} = \sum_{m=1}^{M} b_{k,k'}^m, \tag{9}$$

where

$$b_{k,k'}^m = \begin{cases} 1 & \text{if } D_{k,m} = D_{k',m} = 1 \\ 0 & \text{else} \end{cases}. \tag{10}$$

The number of users served by the m-th AP can be expressed as

$$d(m) = \sum_{k=1}^{K} D_{k,m}, \tag{11}$$

which is an example of the user's AP selection (see Figure 2). The AP of several service users is selected according to the user's large-scale fading matrix. As mentioned earlier, the CPU selects signals received by a part of AP according to the AP selection information for channel estimation and subsequent signal detection. If there is the same AP between users, the channel estimation error will be generated when the same pilot is assigned to the two users. As shown the UE1, UE2, and UE3 in Figure 2, the service matrix $D_{1,1} = 1, D_{1,2} = 1, D_{1,3} = 1, D_{1,4} = 1, D_{2,3} = 1, D_{2,4} = 1, D_{2,5} = 1, D_{3,6} = 1, D_{3,7} = 1, D_{3,8} = 1$, UE1, and UE2 share AP3 and AP4, for coincidence matrix$B_{1,2} = 2$, but there is no case of sharing the same AP between UE2 and UE3, for coincidence matrix $B_{2,3} = 0$. Therefore, the scheme of assigning the same pilot between UE1 and UE2 will cause more pilot contamination and degrade the system's performance than that of UE2 and UE3. The pilot allocation algorithm in this paper is also based on this idea, and the detailed pilot allocation algorithm is described later.

To sum up, the service matrix is a manifestation of the user's choice of service AP. In contrast, the size of the coincidence matrix represents the number of coincident AP between two users and the potential pilot contamination intensity between two users and then designs the pilot allocation algorithm through this value. In the case of users at low speed or even static, the service matrix and coincidence matrix can be considered to be constant for a period of time; so, with the help of AP selection theory, the detection matrix uses part of MMSE (P-MMSE). For the SINR in Equation (6), there are [19]

$$\mathrm{SINR}_k^{(\mathrm{ul},2)} = p_k^u \widehat{g}_k^{\mathrm{H}} \mathbf{D}_k \left(\sum_{j\in O_k} p_j^u \mathbf{D}_k \widehat{g}_j \widehat{g}_j^{\mathrm{H}} \mathbf{D}_k + \mathbf{Z}_k' \right)^{\dagger} \mathbf{D}_k \widehat{g}_k, \tag{12}$$

where

$$\mathbf{Z}_k' = \mathbf{D}_k \left(\sum_{j\in O_k} p_j^u \mathbf{C}_j + \sigma^2 \mathbf{I}_M \right) \mathbf{D}_k, \tag{13}$$

where O_k represents users who have part of the same AP service as user k. The proof of Equation (12) can refer to the related content in reference [23].

3.2. Pilot Allocation. Inspired by the user-centered idea and the scalable problem of CF massive MIMO [19, 22], a pilot allocation scheme based on AP selection is proposed in this paper. To solve the problem of a large amount of calculation and pilot contamination in the case of total AP service for every user, the AP that provides services for each user is selected by the user's large-scale fading matrix. The degree of AP coincidence between users is proposed to quantify the possible pilot contamination intensity when users after selecting AP. For the pilot allocation of users, the AP that serves the most users is first selected, and the pilot allocation of users is carried out under the AP. For users with high repetition, orthogonal pilots are assigned first. To reduce channel estimation error, users under the same AP assign orthogonal pilots as much as possible [24]. Then, select other users under the AP to assign pilots until all users are

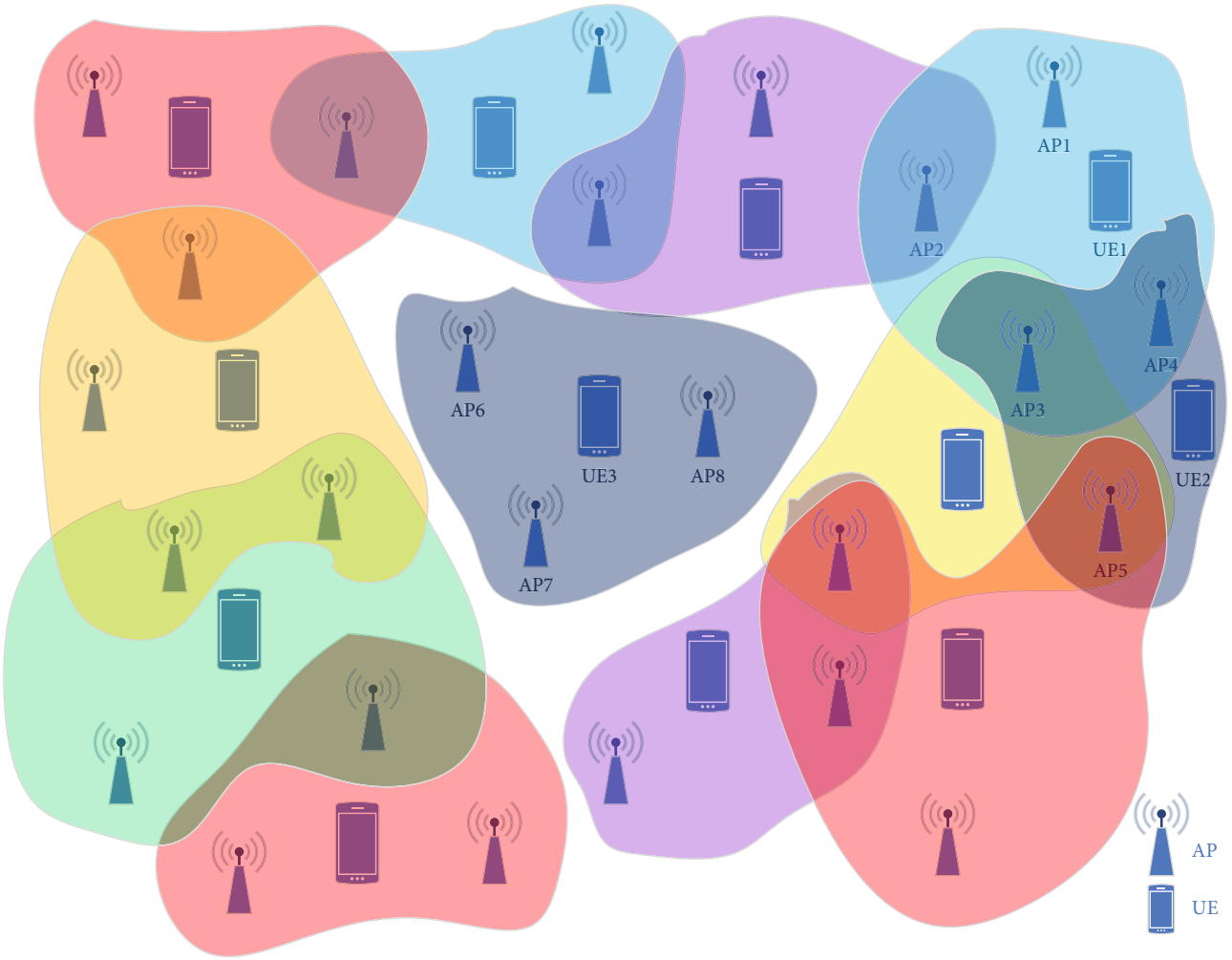


Figure 2: User selects AP sample.

assigned pilots. If all orthogonal pilots are assigned, other users will reuse the same pilots. The detailed steps of the proposed scheme are described in Algorithm 1 and described in the following five parts.

3.2.1. Initialization of Pilot Allocation. The length of the pilot is defined as τ_p, and $\varphi^u = \{\varphi_1, \varphi_2 \cdots \varphi_{\tau_p}\}$ is the set of orthogonal pilots. U is defined as the set of unassigned users, the set of AP service users is defined as U_1, and the set of unassigned users under this AP is $U_2 = U \cap U_1$.

3.2.2. Confirmation of AP Priority. For users under the same AP service, the subsequent channel estimation and signal detection will lead to greater errors and degrade the system performance if the same pilot is assigned. Therefore, when assigning pilots, the users of an AP service are taken as a group to assign pilots. For the AP to serve more users, to reduce pollution, the users under the AP are given priority to assign orthogonal pilots. Hence, the order of the selected AP is according to the number of AP service users. Select the AP through the matrix d in the Equation (11). If the AP in the selection is m_1, then

$$m_1 = \arg\max(d). \tag{14}$$

3.2.3. Confirmation of User Priority. For users' choice, the potential pilot contamination intensity is defined according to the AP coincidence degree between users. For users with a high coincidence degree, their potential pilot contamination is strong; so, orthogonal pilots are given priority. Through the coincident matrix B, select the user with a high coincidence degree of AP and mark it as u, and then

$$u = \arg\max_{o \in U_2} \left(\sum_{k=1}^{K} B_o \right). \tag{15}$$

3.2.4. Pilot Selection. The cumulative pilot contamination between the user and the user who has been assigned pilot is calculated by Equation (3), to obtain the pilot with minimum cumulative pilot contamination, which is recorded as

Input: K, M, D, R, τ_p, F, d
Output: P
initialization: $m = 1$, pilot allocation matrix P and initial user set U.
while $m \le M$ and $U \neq 0$ do
  select the AP m_1 by Equation (14), then set $d(m_1) = -1$ and get a set of S users U_1.
  if $m = 1$ **then**
    $U_2 = \{u_1, u_2, \cdots, u_s\}$ by sorting U_1 in descending order by Equation (15), and $U = U \setminus U_2$.
    if $S \le \tau_p$ **then**
      assign orthogonal pilots to the user set U_2 in turn according to the pilot set $\varphi_1^u = \{\varphi_1, \varphi_2 \cdots \varphi_{\tau_p}\}$.
      add 1 for reuse times of $\varphi_2^u = \{\varphi_1, \varphi_2 \cdots \varphi_S\}$.
    else
      first assign all orthogonal pilots, and unassigned user $U_3 = \{u_{\tau_p+1}, \cdots, u_s\}$.
      while $U_3 \neq 0$ **do**
        select multiplexed pilots φ_i through Equation (16), user u_i in U_3, then assign.
        add 1 for reuse times of φ_i, and $U_3 = U_3 \setminus u_i$.
      end while
    end if
  else
    unassigned users $U_2 = U_1 \cap U$ (assuming I users), and $U = U \setminus U_2$.
    if $U_2 \neq 0$ **then**
      $U_3 = \{u_1, u_2, \cdots, u_I\}$ in descending order by Equation (15) for U_2.
      while $U_3 \neq 0$ **do**
        select multiplexed pilots φ_i through Equation (16), user u_i in U_3 and the number of pilot multiplexing is less than F.
        add 1 for reuse times of φ_i, and $U_3 = U_3 \setminus u_i$.
      end while
    end if
  end if
  $m = m + 1$.
end while

Algorithm 1: Pilot assignment based on AP selection.

φ, and then

$$\varphi = \arg \min_{t_k} tr\left(\Psi_{t_k,m}\right). \tag{16}$$

3.2.5. Other. To avoid the extreme situation that the multiplexing times of the same pilot is too much and the multiplexing times of other pilots are too few, which leads to more severe pilot contamination, the maximum multiplexing times of each pilot is set as F. When the number of users in an AP is less than or equal to the pilot length, all orthogonal pilots can be assigned to minimize pilot contamination among the same group of users.

4. Data Power Optimization

For users, the smaller data transmission power will affect the communication quality of users, and the larger data transmission power will cause power waste and increase the interference between users. Therefore, data power optimization can effectively reduce power waste and interference while ensuring a certain communication quality. Based on pilot allocation, the optimization of data power can further improve the performance of the system. Here, the max-min fairness problem to increase user fairness is proposed, which is described as follows:

$$\begin{aligned} &\max_{p_k^u} \min_{k=1,2,\cdots K} \mathrm{SINR}_k, \\ &\text{subject to } p_k^u \le p_{\max}^u, \forall k \in [1, 2, \cdots K], \\ &p_k^u \ge 0, \forall k \in [1, 2, \cdots K], \end{aligned} \tag{17}$$

where $p_{\max}^u$ is the maximum power value of the transmitted data, and the equivalent form of the formula (17) is

$$\begin{aligned} &\max_{p_k^u, \kappa} \kappa \\ &\text{subject to } \mathrm{SINR}_k \ge \kappa, \forall k \in [1, 2, \cdots K], \\ &p_k^u \le p_{\max}^u, \forall k \in [1, 2, \cdots K], \\ &p_k^u \ge 0, \forall k \in [1, 2, \cdots K]. \end{aligned} \tag{18}$$

Regarding κ as a variable, we can see that the optimization problem of (18) is a linear programming problem [25], which can be solved by CVX [26]. Here, the dichotomy is used to solve the convex optimization problem, and the algorithm for solving the optimization problem (18) is shown in Algorithm 2:

Input: $\kappa_{\min}$, $\kappa_{\max}$, ξ.
Output: p_k^u.
initialize: $\kappa_{\min}$, $\kappa_{\max}$, and threshold ξ.
while $\|\kappa_{\max} - \kappa_{\min}\| > \xi$ do.
  let $\kappa = (\kappa_{\min} + \kappa_{\max})/2$, and solve the optimization problem (18) with CVX.
  if problem solved **then**
  $\kappa_{\min} = \kappa$, and $\kappa_{\max}$unchanged.
  else
    $\kappa_{\min}$unchanged, and $\kappa_{\max} = \kappa$.
  end if
end while.

Algorithm 2: Data power optimization algorithm.

5. Simulation Result

In this section, Monte Carlo is used to simulate the above algorithm in MATLAB, and the performance of pilot allocation algorithm and data power optimization algorithm based on AP selection is obtained.

5.1. Parameter Setting. Here, we consider a scenario with M APs and K UEs, where both AP and UEs are single antennas. UE and AP are randomly distributed in the area of $1 \times 1\text{km}^2$. In order to avoid the boundary effect, the encircling method is used to deal with it [7]. When the system runs in TDD mode, the maximum transmission power of $p_{\max} =$ 100mW for data and pilot and the pilot length is $\tau_p = 10$; the coherence time is $T = 200$. The detailed parameters are as follows (see Table 1).

6. Results and Discussion

According to the above parameters, MATLAB is used to simulate the proposed pilot allocation method. The proposed pilot allocation method is compared with the greedy allocation method in reference [7], the pilot allocation method based on dissimilarity clustering (DCPA) in reference [27], the random pilot allocation method, and the ideal state without pilot contamination. In the figure, the proposed method is represented by proposed, greedy represents the greedy method, DCPA represents the dissimilarity cluster based pilot assignment method, and random represents the random pilot allocation. NoPC represents the one without pilot contamination.

The relationship between the sum SE and the cumulative distribution function (CDF) of the above methods in CF massive MIMO systems is compared (see Figure 3). P-MMSE is used for data detection, and scalable represents a method based on AP selection. As shown in Figure 3, the proposed method is better than the previously mentioned methods in terms of total SE. For the greedy pilot allocation method, because the potential contamination between users is ignored in the initial allocation, and then the greedy method is used to improve the performance, even if the number of iterations is increased, the subsequent pilot allocation may fall into a loop, resulting in no great improvement in performance, but the computational complexity of this method is relatively simple. When initially assigning pilots, the method proposed in this paper considers the potential pilot contamination between users based on the coincidence degree of AP, then by assigning orthogonal pilots to users with high coincidence degree, the pilot contamination problem between users is effectively suppressed, and the performance of the system is improved. At the same time, the performance of the random pilot allocation scheme is the worst. Still, in the case of no pilot contamination, the obtained channel is the actual channel in the uplink channel estimation and detection, which has no interference between users; its performance is also the best. Still, this method is ideal and cannot be realized in practice.

The impact of the number of AP on system performance (see Figure 4): as can be seen from Figure 4, with the increase in the number of AP, the system's performance continues to improve. To increase the number of AP, the degree of channel hardening between users and AP continues to improve. Meanwhile, the channel interference between users continues to reduce; so, the performance continues to improve. As mentioned in reference [28], the number of antennas in an area has a certain influence on the hardening ratio of the channel. With the increase of the number of antennas, the hardening ratio of the channel increases, the interference between channels decreases, and the system's performance is improved. However, as can be seen from Figure 4, when the number of AP is small, the performance is significantly enhanced by increasing the number of AP. Later, with the increase in the number of APs, performance improvement tends to be smooth, and the amount of data received and processed continues to increase; so, the trade-off between the impact of AP number on performance and complexity is also worth studying. At the same time, what is considered here is the case that the AP is a single antenna. In future research, each AP has multiple antennas in B5G and even 6G, which is also a trend and a place worth studying [29].

The relationship between AP selection ratio and average SE (see Figure 5): as the AP selection ratio increases, the system performance continues to increase. However, the more APs select, the greater the number of calculations. At the same time, the pilot contamination between users is also growing. It can be seen that the performance gap between the curve of NoPC and the curve of the other two methods is getting larger and larger. For the case of the full selection

Table 1: Simulation parameters.

Simulation parameters	Symbol	Numerical value
AP number	M	400
Number of users	K	40
Pilot length	τ_p	10
Maximum pilot multiplexing times	F	6
Transmission power	$p_{\max}$	100 mW
Coherent time	T	200
Noise figure	δ	7 dB
Path loss	ϕ	3.76
Shadow fading	σ	10 dB

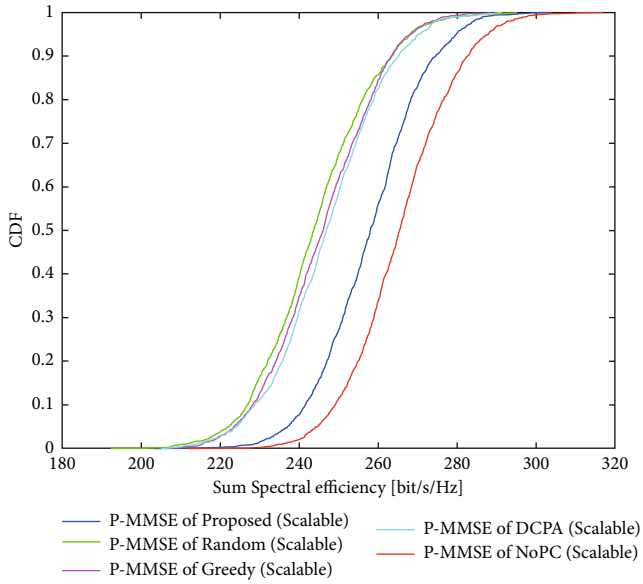


Figure 3: Sum spectral efficiency and CDF.

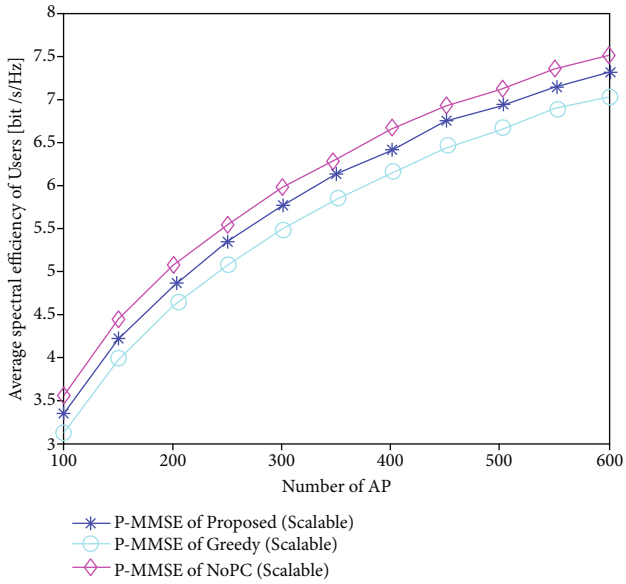


Figure 4: Average spectral efficiency and AP number.

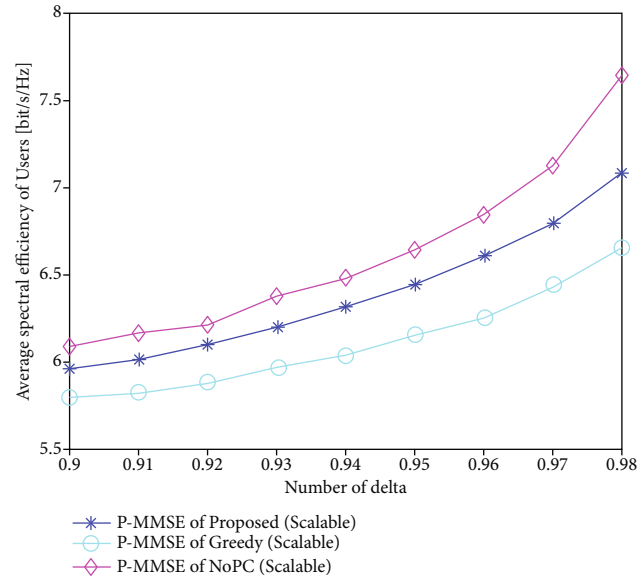


Figure 5: Average spectral efficiency and AP selection ratio.

of AP, the load of the backhaul link from AP to CPU increases, especially in the places where the number of users is large or dense, and the selection based on AP can effectively reduce the backhaul link's load. At the same time, the method proposed in this paper can also effectively suppress pilot contamination, which is a suitable compromise method.

The relationship between AP selection ratio and average SE (see Figure 5): as the AP selection ratio increases, the system performance continues to increase. However, the more APs select, the greater the number of calculations. At the same time, the pilot contamination between users is also growing. It can be seen that the performance gap between the curve of NoPC and the curve of the other two methods is getting larger and larger. For the case of the full selection of AP, the computational complexity of the system is large, especially in places where the number of users is large or dense, and the selection based on AP can effectively reduce the computational complexity of the system. At the same time, the method proposed in this paper can also effectively suppress pilot contamination, which is a suitable compromise method.

The relationship between the user's average number of AP selected and the total number of AP is under different region sizes (see Figure 6). As shown in Figure 6, the average number of AP chosen by users in various areas is much less than the total number of AP. According to the previous results, it is only necessary to select a small amount of APs for each user, rather than all the AP to serve a particular user. It reduces the computational complexity of the system and makes the actual implementation of CF massive MIMO possible.

The CDF diagram of the minimum SE of the system with data power optimization is shown (see Figure 7). It can be seen that the minimum user SE with data power optimization is significantly better than the minimum user SE without data power optimization. It shows that data power

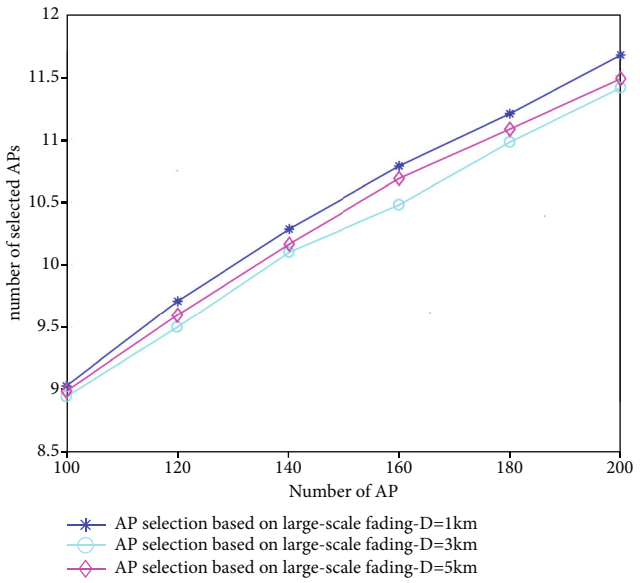


Figure 6: The average number of AP selected by the user and the total number of AP.

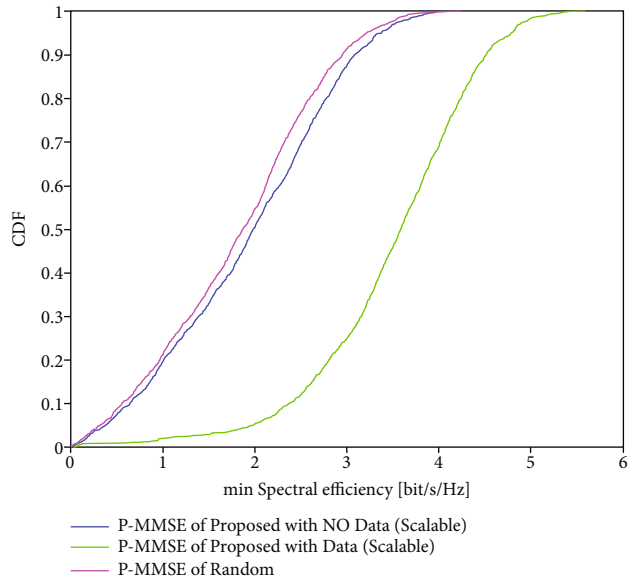


Figure 7: Minimum spectral efficiency and CDF.

optimization can improve system performance and fairness among users, mainly because users' interference intensity will increase when they are in full power. The interference to the users with poor performance will be more significant. Hence, the data power optimization fully considers this point, thus improving the minimum SE of users, while power optimization can also save part of the power consumption.

7. Conclusions

In this paper, aiming at the pilot contamination problem of CF massive MIMO uplink, a pilot allocation method based on AP selection is proposed. And the data power is optimized, which effectively reduces the computational complexity of the system also improves the fairness between users. Firstly, with the help of the user-centered idea, the AP service is selected for each user through the a priori large-scale fading matrix. The potential pilot contamination among users is quantified by the AP coincidence degree of each user. The users with large potential pilot contamination prioritize assigning orthogonal pilots then propose the data power optimization. Through the transformation of the formula, the optimization problem is turned into a convex optimization problem, and the dichotomy is used to solve it. Simulation results show that the proposed algorithm can effectively suppress pilot contamination while reducing the computational complexity of the system and improving the system's total SE and fairness between users. In future research, for the pilot allocation problem and data power optimization problem of CF massive MIMO multi-antennas, the issue of channel aging when users move at a certain speed and the actual hardware loss is worthy of in-depth study.

Data Availability

The simulation code data used to support the findings of this study have not been made available. Because the code supporting this paper is a laboratory project, the source program cannot be made public for the time being.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61871466 and in part by the Key Science and Technology Project of Guangxi under Grant AB19110044.

References

[1] S. Cheng, Z. Cai, J. Li, and H. Gao, "Extracting kernel dataset from big sensory data in wireless sensor networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 4, pp. 813–827, 2017.

[2] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[3] J. Q. Li, F. R. Yu, G. Deng, C. Luo, Z. Ming, and Q. Yan, "Industrial internet: a survey on the enabling technologies, applications, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1504–1526, 2017.

[4] B. Holfeld, D. Wieruch, T. Wirth et al., "Wireless communication for factory automation: an opportunity for LTE and 5G systems," *IEEE Communications Magazine*, vol. 54, no. 6, pp. 36–43, 2016.

[5] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *2019 IEEE 39th International Conference on*

Distributed Computing Systems (ICDCS), pp. 144–153, Dallas, TX, USA, 2019.

[6] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[7] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive MIMO versus small cells," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1834–1850, 2017.

[8] J. Zhang, S. Chen, Y. Lin, J. Zheng, B. Ai, and L. Hanzo, "Cell-free massive MIMO: a new next-generation paradigm," *IEEE Access*, vol. 7, pp. 99878–99888, 2019.

[9] M. Bashar, K. Cumanan, A. G. Burr, H. Q. Ngo, E. G. Larsson, and P. Xiao, "Energy efficiency of the cell-free Massive MIMO uplink with optimal uniform quantization," *IEEE Transactions on Green Communications and Networking*, vol. 3, no. 4, pp. 971–987, 2019.

[10] T. C. Mai, H. Q. Ngo, M. Egan, and T. Q. Duong, "Pilot power control for cell-free massive MIMO," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11264–11268, 2018.

[11] J. Koh, Y. Lim, C. Chae, and J. Kang, "On the feasibility of full-duplex large-scale MIMO cellular systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 9, pp. 6231–6250, 2018.

[12] H. Liu, J. Zhang, X. Zhang, A. Kurniawan, T. Juhana, and B. Ai, "Tabu-search-based pilot assignment for cell-free massive MIMO systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 2, pp. 2286–2290, 2020.

[13] S. Buzzi, C. D'Andrea, M. Fresia, Y. Zhang, and S. Feng, "Pilot assignment in cell-free massive MIMO based on the Hungarian algorithm," *IEEE Wireless Communications Letters*, vol. 10, no. 1, pp. 34–37, 2021.

[14] T. H. Nguyen, T. K. Nguyen, H. D. Han, and V. D. Nguyen, "Optimal power control and load balancing for uplink cell-free multi-user massive MIMO," *IEEE Access*, vol. 6, pp. 14462–14473, 2018.

[15] J. Zheng, J. Zhang, E. Björnson, and B. Ai, "Cell-free massive MIMO with channel aging and pilot contamination," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, pp. 1–6, Taipei, Taiwan, 2020.

[16] S. Buzzi, C. D'Andrea, A. Zappone, and C. D'Elia, "User-centric 5G cellular networks: resource allocation and comparison with the cell-free Massive MIMO approach," *IEEE Transactions on Wireless Communications*, vol. 19, no. 2, pp. 1250–1264, 2020.

[17] E. Björnson and L. Sanguinetti, "A new look at cell-free massive MIMO: making it practical with dynamic cooperation," in *2019 IEEE 30th annual international symposium on personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–6, Istanbul, Turkey, 2019.

[18] G. Femenias, N. Lassoued, and F. Riera-Palou, "Access point switch ON/OFF strategies for green cell-free mssive MIMO networking," *IEEE Access*, vol. 8, pp. 21788–21803, 2020.

[19] E. Björnson and L. Sanguinetti, "Scalable cell-free Massive MIMO systems," *IEEE Transactions on Communications*, vol. 68, no. 7, pp. 4247–4261, 2020.

[20] E. Björnson and L. Sanguinetti, "Making cell-free massive MIMO competitive with MMSE processing and centralized implementation," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 77–90, 2020.

[21] H. Liu, J. Zhang, S. Jin, and B. Ai, "Graph coloring based pilot assignment for cell-free massive MIMO systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 9180–9184, 2020.

[22] S. Buzzi and C. D'Andrea, "Cell-free massive MIMO: user-centric approach," *IEEE Wireless Communications Letters*, vol. 6, no. 6, pp. 706–709, 2017.

[23] B. Emil, H. Jakob, and S. Luca, "Massive MIMO networks: Spectral, energy, and hardware efficiency," *Foundations and Trends in Signal Processing*, vol. 11, no. 3-4, pp. 154–655, 2017.

[24] C. Pan, H. Mehrpouyan, Y. Liu, M. Elkashlan, and N. Arumugam, "Joint pilot allocation and robust transmission design for ultra-dense user-centric TDD C-RAN with imperfect CSI," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 2038–2053, 2018.

[25] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge university press, 2004.

[26] M. Grant and S. Boyd, *CVX: Matlab Software for Convex Programming, Version 2.1*, 2014, http://cvxr.com/cvx.

[27] G. Femenias and F. Riera-Palou, "Cell-free millimeter-wave massive MIMO systems with limited fronthaul capacity," *IEEE Access*, vol. 7, pp. 44596–44612, 2019.

[28] A. Á. Polegre, F. Riera-Palou, G. Femenias, and A. G. Armada, "Channel hardening in cell-free and user-centric massive MIMO networks with spatially correlated Ricean fading," *IEEE Access*, vol. 8, pp. 139827–139845, 2020.

[29] J. Zhang, E. Björnson, M. Matthaiou, D. W. K. Ng, H. Yang, and D. J. Love, "Prospective multiple antenna technologies for beyond 5G," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 8, pp. 1637–1660, 2020.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 2568503, 17 pages
https://doi.org/10.1155/2022/2568503

Research Article

Prediction-Based Resource Deployment and Task Scheduling in Edge-Cloud Collaborative Computing

Mingfeng Su,[1,2] **Guojun Wang**,[3] **and Kim-Kwang Raymond Choo**[4]

[1]*School of Computer Science and Engineering, Central South University, Changsha 410083, China*
[2]*School of Business Information Technology, Hunan Vocational College of Commerce, Changsha 410205, China*
[3]*School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou 510006, China*
[4]*Department of Information Systems and Cyber Security, University Texas San Antonio, San Antonio, TX 78249, USA*

Correspondence should be addressed to Guojun Wang; csgjwang@gzhu.edu.cn

Received 13 September 2021; Revised 3 November 2021; Accepted 18 March 2022; Published 4 April 2022

Academic Editor: Yingjie Wang

Edge computing is becoming increasingly commonplace, as consumer devices become more computationally capable and network connectivity improves (e.g., due to 5G). With the rapid development of edge computing and Internet of Things (IoT), the use of edge-cloud collaborative computing to provide service-oriented network application (i.e., task) in edge-cloud IoT has become an important research topic. In this paper, we present an edge-cloud collaborative computing framework and our resource deployment algorithm with task prediction (RDAP). Based on our paradigm, tasks in the cloud service center are predicted using the two-dimensional time series, and task classification aggregation and delay threshold determination are combined to optimize task resource deployment of edge servers. A task scheduling algorithm with Pareto improvement (TSAP) is also proposed. At the edge servers, the Pareto progressive comparison is conducted in two stages to obtain the tangent point or any intersection point of the two objective curves of user's quality of service and effect of system service to optimize task scheduling. The experimental results show that for varying user task scales and different Zipf distribution α parameters, combining RDAP and TSAP (RDAP-TSAP) can improve the average user task hit rate. In addition, the average task completion time of users, the overall system service effect, and the total task delay rate of RDAP-TSAP are better than TSAP and the benchmark algorithms for task scheduling.

1. Introduction

As Internet of Things (IoT) and other related consumer devices (e.g., Internet of Vehicles and home/medical IoT) become more interconnected and pervasive in our digitally aware environment, there is a need for data analytics to be performed closer to the data sources [1–3]. Doing so allows us to improve users' quality of service, minimize latency, achieve privacy (to some extent), etc. This partly motivates network computing modes such as fog computing [4], transparent computing [5], edge computing [6, 7], and mobile edge computing [8].

In this paper, we focus on edge-cloud collaborative computing in edge-cloud IoT, to leverage the advantages of both cloud and edge computing for a range of services, such as data transmission, resources distribution, and service-oriented network application (i.e., task) offloading. By improving data collaborative processing of both cloud and edge servers, we can potentially minimize data processing delays, improve system scalability, achieve improved system services, etc. In addition, data may have spatiotemporal characteristics such as seasons [9–11], where there is periodic and trend information (explicit and implicit) in the spatiotemporal dimension. In other words, data changes or trends can be effectively predicted [12–14]. To implement edge-cloud collaborative computing in the edge-cloud IoT, cloud service center generally requires massive computing resources to perform data/predictive analytics, which can subsequently be used to guide the deployment of resources required for task operation of edge layer and promote efficient use of resources [15, 16]. At the edge layer, the edge server can balance the needs of users and service providers

through task collaborative offloading, and optimize task scheduling with multiple objectives, so as to enhance user service experience and improve the overall performance.

In edge-cloud IoT, the optimization of service caching (i.e., task resources) and task offloading (i.e., task scheduling) need to solve two problems. (1) The task load of different edge servers changes dynamically over time, and it is necessary to formulate a task resource push strategy according to task changes. (2) Task scheduling should not only consider the needs of users and shorten the task completion time, but also consider the interests of service providers and reduce the overall energy consumption of the system. Therefore, in edge-cloud IoT, we consider edge-cloud collaborative computing and introduce task prediction to study resource deployment and task scheduling optimization. The main contributions are as follows:

(i) A resource deployment algorithm with task prediction (RDAP) is proposed. In the cloud service center, task predictive analytics draw upon both horizontal and vertical time dimensions, and the deployment of resources required for the task operation of the edge servers is optimized to improve the average task hit rate (ATHR)

(ii) A task scheduling algorithm with Pareto improvement (TSAP) is proposed. At the edge server, considering the benefits of users and service providers, Pareto improvement is made to the two objectives of user's quality of service (QoS) and effect of system service (ESS) to optimize task scheduling

(iii) An edge-cloud collaborative computing framework is proposed, and an experimental environment for edge-cloud collaborative computing is constructed. Considering the impact of different user scales and Zipf distribution α parameters, the task scheduling is evaluated and analyzed from the average task completion time (ATCT), overall system service effect (OSSE), and total task delay rate (TTDR).

The rest of this paper is organized as follows. In Section 2, we introduce the related work. We present our designed framework and model of edge-cloud collaborative computing in Section 3. Then, in Sections 4 and 5, we, respectively, describe the task prediction and the resource deployment and task scheduling. Our evaluation setup and findings are presented in Section 6. Finally, we conclude this paper in Section 7.

2. Related Work

Task sharing on demand is one common application in our digitalized society [17, 18], and one of the key challenges is how to improve the QoS by reducing the task completion time. For example, Mao et al. [19] used reinforcement learning and neural network to design scheduling algorithm according to specific workloads, to efficiently schedule data processing jobs, and minimize the average job completion time in distributed clusters. Jalaparti et al. [20] proposed a scheduling method combining data placement and computational optimization to reduce cross rack data transmission and decrease task completion time. Ren et al. [21] predicted the stragglers in the cluster center, by leveraging the copy mechanism. Their approach also considers data location, task execution time, and task interdependence to reduce task delay in centralized and decentralized scheduling. To achieve increased ESS, cloud computing servers could attempt to enhance resource utilization and reduce system costs [22–24]. For example, Andrew et al. [25] put forward a new cluster scheduler for public IaaS platforms. Their scheduler is designed to dynamically allocate virtual machine (VM) instances to improve resource utilization and reduce costs. Liu et al. [26] proposed a joint execution strategy based on an improved genetic algorithm to reduce the overall energy consumption of the system. Nishtala et al. [27] designed a scalable QoS aware task management method, which uses deep reinforcement learning to reduce contention of shared resources, and minimizes data center energy consumption while ensuring QoS. These studies centrally execute user tasks in the cloud computing center. When task requests increase, coupled with reasons such as long transmission distances and limited backhaul links, problems such as high delay and increased energy consumption are likely to occur. The traditional cloud computing model is facing new challenges.

To mitigate some of the limitations associated with cloud computing, there have been attempts to utilize edge computing to offload all or part of the computationally intensive tasks to the edge servers and extend the computing power to the edge layer. Such an approach can potentially reduce data processing delays [28]. For example, Rodrigues et al. [29] proposed a method to minimize service delay, by offloading tasks that users cannot run to cloudlet servers at the edge network. They also attempt to reduce processing and transmission delays of tasks through VM migration and transmission power control. Mao et al. [30] presented a Lyapunov optimization-based dynamic computation offloading algorithm, which focuses on minimizing execution delay and execution cost of task failure, maximizing battery capacity of mobile devices, and gradually optimizing computationally intensive workloads to improve QoS and user's quality of experience (QoE). Chen et al. [31] used a mixed-integer nonlinear programming method to optimize task dispatch and resource allocation, solve mobile edge computing ultra-dense network task offloading, and minimize delays under the premise of considering device battery life. He et al. [32] proposed an incentive mechanism for online auction, which offloads user tasks to neighboring mobile devices to meet low latency requirements. However, these studies mainly focus on user's service needs and reduce the task completion time to optimize task offloading. Task offloading lacks consideration of system energy consumption optimization for service providers.

To enhance both QoS and ESS, edge computing needs to think of task offloading, network load, resource allocation, and transmission delay [33–35]. Wang et al. [36] proposed a local optimization algorithm for the univariate search technology, which introduces dynamic voltage scaling

technology in computational offloading, and uses variable replacement technology to find the optimal solution to minimize mobile energy consumption and application execution delay. Dinh et al. [37] put forward a task offloading framework from mobile devices to multiple edge devices, considering both fixed and flexible mobile device CPU frequencies based on the semi-definite relaxation approximation method to enhance task execution delay and device energy consumption. Zhang et al. [38] raised an energy-aware computing offloading scheme, which optimizes communication resources and computing resources allocation under limited energy and delay conditions, and finds out the hybrid nonlinear integer optimal solution of computing offloading and resource allocation through an iterative search algorithm. Wang et al. [39] came up with an optimal resource allocation scheme that combines AP energy transmission consumption, CPU processing frequency, user offloading file size, and user time allocation. The scheme thinks of computing and wireless power transmission to minimize the total AP energy consumption under the constraint of individual computing delay. Ding et al. [40] proposed a decentralized offloading strategy in a mobile edge computing environment with limited user equipment resources. The task execution location, CPU frequency, and transmission power are optimized based on code partition offloading to minimize application execution time and energy consumption. The above research work mainly considers task offloading. It is assumed that the edge server already has the relevant service cache required to execute the task, and service cache is also called task resource. When tasks are offloaded to edge servers, only the computing resource constraints are considered, but task resource constraints are not considered. However, in practical applications, the storage resources of edge servers are limited, and it is difficult to cache all task resources required for task execution [41]. It is necessary to dynamically formulate task resource deployment strategy according to the actual situation, and jointly optimize task resource and task offloading.

Therefore, the cloud computing capability is extended to the edge servers in edge-cloud IoT, and the edge-cloud collaborative computing framework is proposed. The task prediction is introduced to study the resource deployment and task scheduling optimization in the edge-cloud collaborative computing environment, and improve both QoS and ESS for users and service providers.

3. Framework and Modeling of Edge-Cloud Collaborative Computing

In this section, we first design the edge-cloud collaborative computing framework, and then model the edge-cloud collaborative computing, including quantifying user's quality of service and effect of system service. The main symbols and descriptions of this paper are shown in Table 1.

3.1. Edge-Cloud Collaborative Computing Framework. The edge-cloud collaborative computing framework is divided into the cloud layer, edge layer, and sensor layer, which are interconnected through the Internet. As shown in Figure 1,

Table 1: Description of main symbols.

Symbols	Description
U	Set of user
J	Set of task
E	Set of edge server
c	Cloud service center
q_u^j	QoS's coefficient of user u task j
r_u^j	System service revenue coefficient of user u task j
$\omega_{u,e}^j$	Proportion of task performed on receiving edge server for task j
$\omega_{e,e}^j$	Proportion of task assigned by the receiving edge server to other edge servers for task j
$\omega_{e,c}^j$	Proportion of task assigned by the receiving edge server to cloud service center for task j
f_e^j	Binary variable, 1/0 indicates whether the edge server e can execute the task j or not
l_u^j	The size of the task j for the end user u
k_u^j	Binary variable, 1/0 indicates whether the user u has the request task j or not
D	Distance coefficient between nodes
$d(x, y)$	Distance degree between nodes x and y
μ	Task execution preference weight coefficient
β	Revenue index
δ	Computing resources
ε	Storage resources
Δ	Task type, binary variable, 1/0 refers to time-sensitive and time-insensitive tasks, respectively
τ	Task execution time equivalent
∂	Average server energy consumption coefficient
ξ	Task delay occurrence threshold

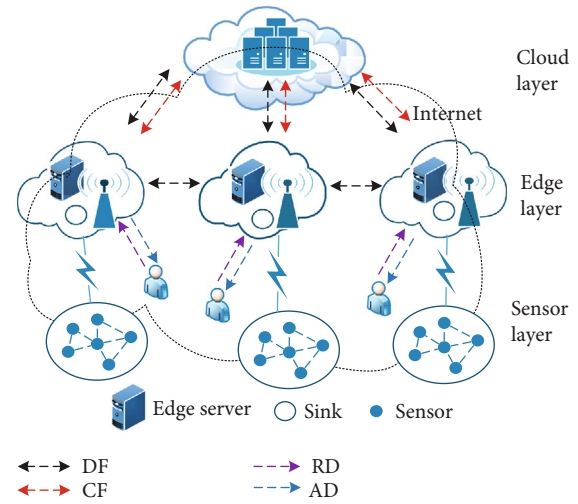


Figure 1: Edge-cloud collaborative computing framework.

edge-cloud collaborative computing can be applied in scenarios such as smart transportation, telemedicine, and environmental monitoring. The cloud layer includes a cloud service center, which is composed of some hardware such as homogeneous or heterogeneous computing, storage, network, and other hardware. The cloud service center uses virtualization, software defined network, redundancy, and other technologies to supply high-performance, highly reliable, and scalable resources to support a wide range of on-demand services for users. Control flow (CF) is generated between the cloud layer and edge layer. The cloud service center predicts the user task, and pushes the task resources to the edge servers through the CF in advance according to the prediction result. Task resources include software and software-dependent data required for task operation. The cloud service center monitors the task processing and resource utilization of the edge servers in real-time, summarizes the task processing and resource utilization of each edge server, and then distributes them to other edge servers through the CF.

The edge layer is composed of edge servers with limited resources, road side units (RSU), and sinks. The edge layer collects data from the sensor network in real-time. The edge server can aggregate and process data and provide users with real-time, fast, diverse, and flexible network applications. According to the current resource utilization and scheduling strategy, the edge server decides to execute the user tasks separately in the local edge servers, or subdivide and schedule the tasks to other edge servers and/or cloud service center for collaborative execution. Data flow (DF) is generated by executing tasks between edge-edge and edge-cloud. The edge server receives and loads the task resources from the cloud service center through the CF, and pre-start the environment required for task operation. The edge server uploads the task processing status and the resource usage status of computing, storage, and network to the cloud service center through the CF.

The sensor layer includes a sensor network, which is composed of a large number of sensors of IoT. Sensors can be deployed in environments such as smart transportation, telemedicine, and environmental monitoring. The sensor collects various required data in real-time and masters the status information of the monitoring area (object). The data collected by the sensors are uploaded to the edge layer in time. The uploaded data at the sensor layer can be analyzed and processed by the edge layer and cloud layer.

In edge-cloud IoT, users can initiate service-oriented task requests through mobile devices, computers, connected cars, and smart terminals, and send request data (RD) to the edge server. The requested task is executed by the local edge server alone or in collaboration with other edge servers and/or cloud service center, answer data (AD) will be returned to the client from the local edge server, other edge servers, and cloud service center. Take the application of edge-cloud collaborative computing in smart transportation as an example. Sensors in each area monitor passing vehicles and control traffic information, and upload the traffic information to the edge servers in each area in time. The edge server processes the data and uploads the summarized traffic information to the traffic command center (cloud service center). When a user initiates a task request, for example, the task is to obtain a regional traffic condition map. The edge server closest to the user receives the user's request. The edge server can perform tasks by itself, or dispatch to other adjacent edge servers and/or cloud service center to perform tasks cooperatively. Finally, the traffic condition map of a certain area is obtained, and the task result is returned to the user. The execution of user tasks requires the cloud service center to predict user tasks and push the task resources required to perform tasks to the edge server in advance. Task resources include software (program for drawing traffic conditions) and software data dependency (basic map).

3.2. Edge-Cloud Collaborative Computing Modeling.

$$q_u^j = l_u^j \left(\mu_u^e \cdot \omega_{u,e}^j / D_u^e + \mu_e^e \cdot \omega_{e,e}^j / D_{u,e}^e + \mu_e^c \cdot \omega_{e,c}^j / D_{u,e}^c\right). \quad (1)$$

Definition 1. Edge-cloud collaborative computing model (EC3M). EC3M is a six-tuple model, denoted as M_{EC3}, $M_{\mathrm{EC3}} = (U, J, E, c, O, \theta)$. U is the user set, which is composed of $n(u)$ independent users, $U = \{u_0, u_1, \cdots, u_{n(u)-1}\}$. Users do not interfere with each other, and the various tasks submitted by users have a time-series correlation, so the number and types of tasks can be predicted. J is the task set, which is composed of $n(j)$ service-oriented network applications (i.e., tasks), $J = \{j_0, j_1, \cdots, j_{n(j)-1}\}$. The task j is expressed as $j = \{\delta_j, \varepsilon_j, \Delta_j\}$. δ_j is the computing resources required for task j execution, which is quantified as the CPU computing power required for each task, i.e., GHz/task. ε_j is the storage resources required for task j execution. Δ_j is the task type of task j. $\Delta_j = 1$ indicates that task j is a time-sensitive task, and $\Delta_j = 0$ indicates that task j is a time-insensitive task. Each task type includes a variety of different tasks to meet the needs of various users. The task can be subdivided into several subtasks. E is the edge server set, which contains $n(e)$ geographically dispersed edge servers, $E = \{e_0, e_1, \cdots, e_{n(e)-1}\}$. The edge server $e = \{\delta_e, \varepsilon_e\}$ has limited hardware resources. δ_e and ε_e represent the computing resources and storage resources of the edge server e, respectively. The edge server is limited by hardware resources and can only load task resources required for some tasks simultaneously. The edge servers can subdivide the tasks and execute tasks locally or dispatch them to remote execution based on scheduling decisions. c is the cloud service center, which has massive computing, storage, network, and other hardware resources, and can load and run task resources for all tasks. The cloud service center manages and monitors the edge servers, effectively predicts the user tasks, and pushes the appropriate task resources to the relevant edge servers. O is the optimization objective of edge-cloud collaborative computing, which is quantified by both QoS and ESS (denoted as Q and S), $O = \{\max(Q), \max(S)\}$. θ is the optimization algorithm for resource deployment and task scheduling.

Definition 2. User's quality of service (QoS). QoS mainly focuses on the service experience and quality of user in the

edge-cloud collaborative computing environment. User tasks are executed locally in the edge servers that receive the tasks and the shorter the response time of user task requests, the higher the QoS. The latter's coefficient is stated in (1).

The QoS's coefficient is related to the task size and task execution. The task size is quantified as task execution time. $w^j_{u,e}$, $w^j_{e,e}$, and $w^j_{e,c}$ represent the proportion of task j executed locally by the receiving edge server, dispatched by the receiving edge server to other edge servers, and dispatched by the receiving edge server to cloud service center, respectively. Their corresponding task execution preference weight coefficients are μ^e_u, μ^e_e, μ^c_e, and there is $\mu^e_u > \mu^e_e > \mu^c_e > 0$, $\mu^e_u + \mu^e_e + \mu^c_e = 1$. The distance coefficient between nodes considers the sending and receiving of tasks and it is related to the distance degree between nodes. D^e_u is the distance coefficient between nodes that the task locally executed at the receiving edge server, $D^e_u = d(u, e) + d(e, u)$. $D^e_{u,e}$ is the distance coefficient between nodes that the task is dispatched by the local edge server to other edge servers, $D^e_{u,e} = d(u, e) + d(e, e) + d(e, u)$. $D^c_{u,e}$ is the distance coefficient between nodes that the task is dispatched by the local edge server to the cloud service center, $D^c_{u,e} = d(u, e) + d(e, c) + d(c, u)$. The distance degree $d(x, y)$ is related to the minimum bandwidth, cumulative delay, and reliability of the links between nodes x and y. The value of distance degree is equal to the cumulative delay divided by product of reliability and minimum bandwidth. If the minimum bandwidth is larger, the cumulative delay is smaller, and the reliability is higher, then the distance degree value is smaller. The objective function of QoS is described in (2).

$$Q = \sum_{u=0}^{u(n)-1} \sum_{j=0}^{j(n)-1} k^j_u \left(f^j_e \cdot q^j_u + \left(1 - f^j_e\right) \cdot \mu_0 \cdot l^j_u \cdot D^c_u \right) \cdot \begin{cases} \max\ (Q) \\ \text{s.t.} f^j_e = \{0, 1\}, \forall e \in E, j \in J\ . \\ k^j_u = \{0, 1\}, \forall u \in U, j \in J \end{cases} \tag{2}$$

$k^j_u = 1$ indicates that user u has a request for task j. $f^j_e = 1$ implies that the edge server e has the task resources required for task j to run. D^c_u is the distance coefficient between nodes of the task submitted by the user to the cloud service center, $D^c_u = d(u, c) + d(c, u)$. μ_0 is the non-preference weight coefficient, $\mu_0 = -\mu_e$. It can be seen from (2) that the edge servers have the resources required for the task operation and the larger the proportion of tasks executed locally, the larger the Q value, that is, the higher the user service quality of edge-cloud collaborative computing.

$$r^j_u = l^j_u \left(\mu^e_u \cdot \omega^j_{u,e} \cdot \beta^e_u + \mu^e_e \cdot \omega^j_{e,e} \cdot \beta^e_e + \mu^c_e \cdot \omega^j_{e,c} \cdot \beta^c_e \right). \tag{3}$$

Definition 3. Effect of system service (ESS). ESS mainly concentrates on the system service revenue and system service consumption of service providers in the edge-cloud collaborative computing. The system service revenue coefficient is stated in (3).

The system service revenue coefficient is associated with the task size and task revenue. β^e_u is the revenue index of the tasks performed on the local edge servers. β^e_e is the revenue index of the edge servers dispatching the received tasks to the neighbor edge servers. β^c_e is the revenue index of the edge servers dispatching the received tasks to the cloud service center. The objective function of ESS is shown in (4).

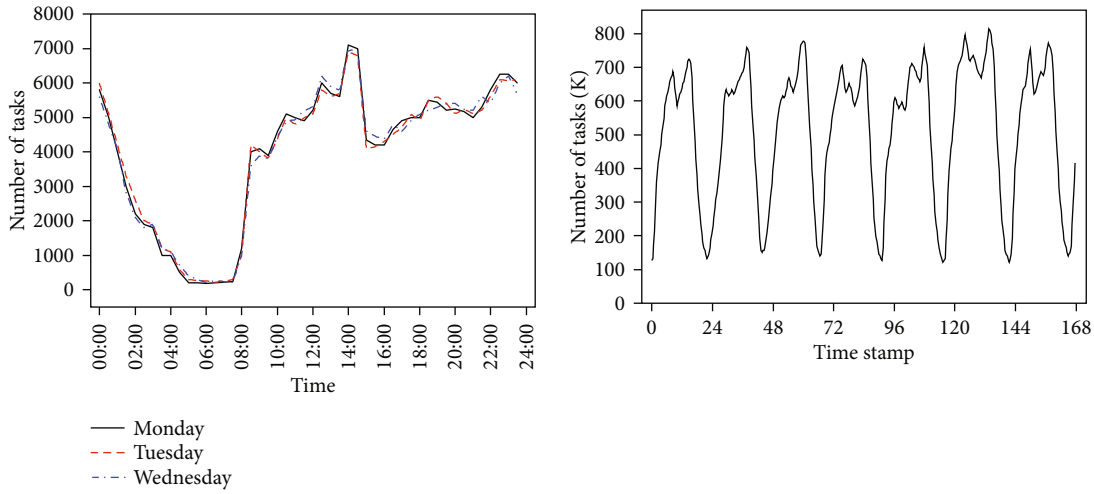
$$S = \sum_{u=0}^{u(n)-1} \sum_{j=0}^{j(n)-1} k^j_u \left(f^j_e \cdot r^j_u + \left(1 - f^j_e\right) \mu_0 \cdot l^j_u \cdot \beta^c_e \right) - \left(\tau_c \cdot \partial_c + \sum_{i=0}^{e(n)-1} \tau^i_e \cdot \partial^i_e \right) \begin{cases} \max\ (S) \\ \text{s.t.} f^j_e = \{0, 1\}, \forall e \in E, j \in J\ . \\ k^j_u = \{0, 1\}, \forall u \in U, j \in J \end{cases} \tag{4}$$

In (4), the system service consumption in the statistical period is the product of task execution time equivalent and average service energy consumption coefficient (denoted as τ and ∂). τ_c and τ_e represent the task execution time equivalent of the cloud service center and edge servers, respectively. ∂_c and ∂_e represent the average service energy consumption coefficient of cloud service center and edge servers, respectively. The energy consumption coefficient depends on the hardware/software costs, and the system operation and maintenance costs. The former includes the hardware/software purchase costs and the depreciation costs. The latter gets involved in the power consumption of equipment, the energy consumption of air conditioning and refrigeration, and the management and service costs. From the comparative analysis of single quantity, ∂_c is much larger than ∂_e. The higher the system service revenue and the lower the system service consumption, the larger the S value, that is, the higher the system service effect of edge-cloud collaborative computing.

In the edge-cloud collaborative computing model, the objective optimization of QoS and ESS is involved in resource deployment and task scheduling. (1) It needs to foretell the type and quantity of user tasks, reasonably push the task resources to the edge servers, and efficiently use the computing, storage, network, and other resources of the edge servers. (2) It also should optimize task scheduling, improve the QoS, and strengthen the ESS through the task collaborative processing of local edge servers, other edge servers, and cloud service center.

4. Task Prediction

Through long-term monitoring of user tasks, from a local perspective, the change of user tasks is a dynamic and random process, and a trend of change can be seen explicitly or implicitly. The task change and time have a strong correlation. From a global perspective, user tasks are



(a) External network interface of university data center

(b) Cluster server of e-commerce company

Figure 2: Task request statistics.

autocorrelated with time such as days, weeks, months, and years. Based on the task change rules of the horizontal and vertical time dimensions, the task changing trend can be found through prediction. Figure 2(a) shows the task request statistics chart of the network external network interface of a university's data center. Figure 2(b) shows the statistics of the number of tasks executed in the cluster server of an e-commerce platform. Both figures use hours as the statistical unit, and the number of tasks changes with time. By observing the number of tasks listed in Figure 2(a) for 3 consecutive days and Figure 2(b) for 7 consecutive days, it is found that the changing trend of daily tasks in the same period is similar, and the overall shows the regular periodic fluctuations. This provides a reference basis for user task prediction in edge-cloud IoT.

According to the overlapping trend of strong periodicity, medium trend, and weak randomness of user tasks in edge-cloud collaborative computing, the user tasks can be comprehensively predicted from the horizontal and vertical time dimensions in the cloud service center. Through statistics and analysis of the past time-series data, the change of tasks can be inferred. The prediction model based on the two-dimensional time series is indicated in (5).

$$p_t = \lambda v_t + (1-\lambda)\eta_t + z_t. \tag{5}$$

p_t is the task prediction result of the next time series. v_t shows the current time series value of the vertical time dimension, which is the periodic statistics part, specifically the mean value of the same time series in different periods. v_t is the mean value of the same time series in different periods. η_t is the current time series value of the horizontal time dimension, which is the trend prediction part. z_t is the random noise part. λ represents the adjustment factor of two-dimensional time series, and its value is between 0 and 1.

Theorem 4. *The task of trend prediction based on the horizontal time dimension in edge-cloud collaborative computing is* $\eta_t = (\eta_{t-1} + \eta_{t-2} + \cdots + \eta_{t-n}/n) + (n + 1/2)\varphi$.

Proof. According to the statistical analysis of practical data, the number of tasks increases or decreases linearly from a local perspective. The change in the number of tasks at time t can be expressed by a linear equation, as shown in (6). Thus, the number of tasks with time $t - n$ can be obtained, as shown in (7).

$$\eta_t = \varphi\chi_t + a, \tag{6}$$

$$\eta_{t-n} = \varphi\chi_{t-n} + a. \tag{7}$$

According to the moving average prediction method, the number of tasks at time t is related to the number of tasks in the previous n time slots, as shown in (8).

$$\bar{\bar{\eta}}_t = \frac{\eta_{t-1} + \eta_{t-2} + \cdots + \eta_{t-n}}{n}. \tag{8}$$

Combining formula (7), and calculating $\bar{\bar{\eta}}_t$ to obtain (9).

$$\begin{aligned}\bar{\bar{\eta}}_t &= \frac{\varphi\chi_{t-1} + a + \varphi\chi_{t-2} + a + \cdots + \varphi\chi_{t-n} + a}{n} \\ &= \frac{\varphi(\chi_{t-1} + \chi_{t-2} + \cdots + \chi_{t-n}) + na}{n} \\ &= \frac{\varphi(n\chi_t - n(n+1)/2) + na}{n} \\ &= \varphi\chi_t - \varphi(n+1)/2 + a = \varphi\chi_t + a - \varphi(n+1)/2.\end{aligned} \tag{9}$$

According to formula (6), the number of tasks $\eta_t = \varphi\chi_t + a$ at time t is substituted into formula (9), and (10) is obtained.

$$\bar{\bar{\eta}}_t = \eta_t - \varphi(n+1)/2. \tag{10}$$

There is a delay deviation of $\varphi(n+1)/2$ between $\bar{\bar{\eta}}_t$ (the number of tasks predicted by the moving average method) and η_t (the actual number of tasks), which needs to be corrected. Therefore, the task of trend prediction based on the horizontal time dimension is $\eta_t = \bar{\bar{\eta}}_t + \varphi(n+1)/2$, which is (11). The φ value can be calculated by a linear regression formula, $\varphi = \sum \chi_i \sum \eta_i - n\bar{\chi}\bar{\eta} / \sum \chi_i^2 - n\bar{\chi}^2$. It should be pointed out that if the number of tasks is absolutely stationary in the statistical interval, its $\varphi = 0$.

$$\eta_t = \frac{\eta_{t-1} + \eta_{t-2} + \cdots + \eta_{t-n}}{n} + \frac{n+1}{2}\varphi \tag{11}$$

□

Thus, the task prediction based on the two-dimensional time series is gotten, as stated in (12).

$$p_t = \lambda v_t + (1-\lambda)\left(\frac{\eta_{t-1} + \eta_{t-2} + \cdots + \eta_{t-n}}{n} + \frac{\eta+1}{2}\varphi\right) + z_t. \tag{12}$$

5. Resource Deployment and Task Scheduling

In this section, we introduce the resource deployment and task scheduling optimization of edge-cloud collaborative computing. Firstly, the cloud service center pushes the task resources to the appropriate edge servers according to the task prediction result to improve the local execution rate of user tasks. Secondly, the edge servers optimize the task scheduling through Pareto improvement to enhance both QoS and ESS.

5.1. Resource Deployment Algorithm with Task Prediction (RDAP). In the edge-cloud collaborative computing environment, the hardware resources of the edge servers are limited, and the tasks they perform are also limited. The cloud service center needs to monitor the task processing status of the edge layer in real-time, use task prediction based on two-dimensional time series to obtain the changing trend of task type and quantity, and push the task resources to the edge servers. Task running needs to occupy hardware resources, this paper mainly considers computing resources (δ) and storage resources (ε) [42]. The maximum available resource of the edge server e is h_e, $h_e = \{\delta_e^{\max}, \varepsilon_e^{\max}\}$. The currently available resource of the edge server e is b_e, $b_e = \{\delta_e^{\text{cur}}, \varepsilon_e^{\text{cur}}\}$. The resource consumption of task j is $g_j = \{\delta_j, \varepsilon_j\}$. Considering the limited number of tasks performed by the edge servers, the cloud service center needs to classify and aggregate the prediction tasks, and control the number of task resources pushed to the edge servers, keeping the task load of the edge servers in a reasonable range. The RDAP is shown in Algorithm 1.

The resource deployment with task prediction is performed in the cloud service center. Tasks $j \in J$ are classified into two types: time-sensitive task ($\Delta_j = 1$) and time-insensitive task ($\Delta_j = 0$) (Line 1). According to the monitoring data, the current and maximum available resources of each edge server are calculated (Lines 2-3). Formula (12) is used to predict tasks based on the two-dimensional time series (Line 4). The tasks of each edge server are classified and aggregated in line with the prediction result, arranging them in descending order by the occurrence frequency (Line 5). This can reduce the push quantity of task resources, decrease the resource occupation of the edge servers, and improve the hit rate of user tasks under the resource shortage. Based on the task classification and aggregation result, the resource deployment of time-sensitive tasks is first considered, and then the resource deployment of time-insensitive tasks is taken into account when the edge servers have surplus resources available, which is conducive to ensuring the user's quality of service (Lines 6-23). After obtaining the task resources deployed by each edge server, the currently available resources of each edge server are updated. To decrease the task delay, the delay threshold is determined to make sure that the proportion of available resources (including computing resources and storage resources) of the edge servers is higher than the task delay occurrence threshold ξ, so as to control the number of task resources pushed to the edge servers. Finally, the edge server task resource deployment set X is returned, and the cloud service center pushes the task resources to the edge servers (Line 25). The time complexity of the RDAP is $O(n(e) \times n(j)(\text{lb}(n(j)) + 1))$.

5.2. Task Scheduling Algorithm with Pareto Improvement (TSAP). The task scheduling of the edge-cloud collaborative computing should be oriented to users and service providers, and the QoS and ESS ought to be considered comprehensively. The single objective optimization of task scheduling cannot ensure that the other objective is also optimal. It is necessary to weigh two objectives for comprehensive optimization. For example, task scheduling to nodes with strong computing capabilities can reduce the response time of tasks and improve QoS, but it is easy to increase system service consumption and reduce ESS. It is necessary to consider the interests of users and service operators to weigh QoS and ESS. Therefore, the Pareto improvement in the field of economics is introduced, and the task scheduling scheme of edge-cloud collaborative computing is obtained by seeking Pareto improvement for both QoS and ESS. The TSAP is described in Algorithm 2.

The task scheduling with Pareto improvement is performed in the edge servers. New tasks are received and added to the task set J. If multiple users initiate task requests at the same time period, the tasks are sorted in ascending order (Lines 1-6). Pareto improvement of task scheduling is involved two stages (Lines 7-21). The first stage is the objective optimization of the QoS (Lines 8-13). The m-group of task scheduling schemes in the first stage is solved based on the random greedy approximation algorithm. Formula (2) is used to calculate the Q value in each group, and select the scheduling scheme with the highest Q value in each group. The objective curve of QoS can be obtained from these schemes set. The second stage is the objective optimization of the ESS (Lines 14-19). Based on

Input: E: edge server set; J: task set; H: edge server maximum available resource set; B: edge server current available resource set.
Output: X: edge server task resource deployment set.

classify all j into time-sensitive or time-insensitive task, $\exists j \in J, \Delta_j = \{1, 0\}$;
for each $e \in E$ **do**
  calculate h_e and b_e, $\exists h_e = \{\delta_e^{\max}, \varepsilon_e^{\max}\}$, $b_e = \{\delta_e^{\text{cur}}, \varepsilon_e^{\text{cur}}\}$;
  predict p_t by formula (12);
  sort, aggregate, and rank tasks for J in descending order of frequency;
  for each $j \in J$ **do**
    if $f_e^j = 1 \cap \Delta_j = 1$ **then**
      if $\min\{(\delta_e^{\text{cur}} - \delta_j/\delta_e^{\max}), (\varepsilon_e^{\text{cur}} - \varepsilon_j/\varepsilon_e^{\max})\} > \xi$ **then**
        $\delta_e^{\text{cur}} \longleftarrow \delta_e^{\text{cur}} - \delta_j$, $\varepsilon_e^{\text{cur}} \longleftarrow \varepsilon_e^{\text{cur}} - \varepsilon_j$;
        update $X \longleftarrow X_e^j$;
      end if
    end if
  end for
  if $\min\{\delta_e^{\text{cur}}/\delta_e^{\max}, \varepsilon_e^{\text{cur}}/\varepsilon_e^{\max}\} > \xi$ **then**
    for each $j \in J$ **do**
      if $f_e^j = 1 \cap \Delta_j = 0$ **then**
        if $\min\{(\delta_e^{\text{cur}} - \delta_j/\delta_e^{\max}), (\varepsilon_e^{\text{cur}} - \varepsilon_j/\varepsilon_e^{\max})\} > \xi$ **then**
          $\delta_e^{\text{cur}} \longleftarrow \delta_e^{\text{cur}} - \delta_j$, $\varepsilon_e^{\text{cur}} \longleftarrow \varepsilon_e^{\text{cur}} - \varepsilon_j$;
          update $X \longleftarrow X_e^j$;
        end if
      end if
    end for
  end if
end for
return X

Algorithm 1: RDAP.

Input: E: edge server set; J: task set; c: cloud service center; μ_u^e, μ_e^e, μ_e^c; β_u^e, β_e^e, β_e^c; ∂_c, ∂_e.
Output: Y: task scheduling scheme set.

new j received
  $J \longleftarrow J + j$;
end if
if count$(J) \neq \varnothing$ **then**
  sort J in ascending order;
end if
for each $j \in J$ **do**
  for each $E \cup C$ **do**
    if $f_e^j = 1, \exists j \in J, e \in E$ **then**
      calculate Q according to formula (2);
      select $Y_j^1[m] \subseteq Y^1$, $Y_j^1[m] \succ \max(Q)$;
    end if
  end for
  for each $E \cup C$ **do**
    if $f_e^j = 1, \exists j \in J, e \in E$ **then**
      calculate E according to formula (4);
      select $Y_j^2[m] \subseteq Y^2$, $Y_j^2[m] \succ \max(S)$;
    end if
  end for
  $Y \longleftarrow Y_j^1[m] \cap Y_j^2[m]$;
end for
return Y

Algorithm 2: TSAP.

the random greedy approximation algorithm, the m-group task scheduling schemes in the second stage are solved. The S value in each group is calculated by using formula (4), and the scheduling scheme with the highest S value in each group is selected. The objective curve of ESS can be obtained by these schemes set. The task scheduling schemes obtained in two stages are gradually compared by Pareto improvement, and the tangent point or any intersection point corresponding to the QoS and ESS objective curves are selected. The tangent point is the only optimal scheme for the two objectives, and the intersection point is any co-optimal scheme for the two objectives, so as to obtain the optimal scheduling scheme for each task (Line 20). Finally, the task scheduling scheme set Y is returned (Line 22). The time complexity of the TSAP is $O(n(j) \times (\mathrm{lb}(n(j)) + (n(e) + 1) \times m))$.

6. Experimental Evaluation

In this section, we build an edge-cloud collaborative computing experimental environment, set experimental parameters, determine evaluation indexes, and comprehensively evaluate resource deployment and task scheduling.

6.1. Experimental Environment and Parameters. The hardware platform of the edge-cloud collaborative computing experiment is an x86 server with Intel e5-2620v4 CPU, 64GB ECC RAM, and 3 × 2TB STA hard disk. The software platform relies on the CentOS 8.0 x86_64 operating system, and uses python 3.8 to build an edge-cloud collaborative computing simulation environment. In addition, OriginPro 2017 software is used for data analysis and post drawing. The parameters of the experimental environment are shown in Table 2. In the edge-cloud collaborative computing environment, the user task type, task execution time, CPU resource utilization, and RAM resource utilization are set by referring to the data set published by Alibaba Cloud [42–44]. User tasks have time series correlation, and the overall distribution is Zipf [45–47]. The frequency of tasks is inversely proportional to the rank of task popularity. In the experiment, the default value of Zipf distribution α parameter is 1.0 [48, 49]. The maximum available resources of the cloud service center are 10 K PE CPU, 10 TB RAM, and 10 PB disk. The hardware resource parameters of the edge servers are shown in Table 3. The benchmark algorithms for task scheduling include BAO (the benchmark task scheduling algorithm with OREO) [50] and BAF (the benchmark task scheduling algorithm with FIFO) [51, 52]. Task scheduling evaluation metrics include ATCT, OSSE, and TTDR.

6.2. Experimental Results and Discussion

6.2.1. Resource Deployment. In the edge-cloud collaborative computing environment, resource deployment with task prediction is evaluated and analyzed. The day is taken as the statistical cycle, each cycle is divided into 24 time periods, and each time period is 0.5 hours. In the experiment, the cloud service center predicts the tasks of the edge servers based on the two-dimensional time series, and the value of time slot n is 10. According to the experimental data

Table 2: Experimental environment parameters.

Parameters	Value
Types of tasks	300
Number of edge servers	15
δ_j/GHz	[0.1-0.3]
ε_j/MB	[20-80]
δ_e/GHz	[2.5-3.6]
μ_u^e, μ_e^e, μ_e^c	0.48, 0.31, 0.21
β_u^e, β_e^e, β_e^c	0.39, 0.36, 0.25
ξ	0.20

Table 3: Hardware resource parameters of edge servers.

Edge server number	CPU/PE	RAM/GB	Disk/GB
E01-E03	2 ~ 4	4 ~ 8	500
E04-E06	4 ~ 6	4 ~ 8	500
E07-E09	4 ~ 6	8 ~ 16	1000
E10-E12	4 ~ 8	8 ~ 16	1000
E13-E15	6 ~ 16	16 ~ 32	1000

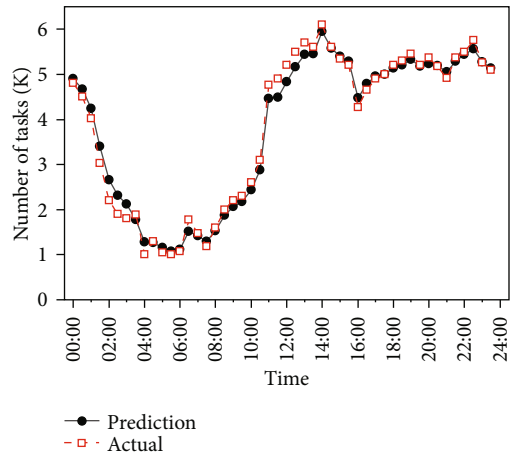


Figure 3: Prediction of user tasks.

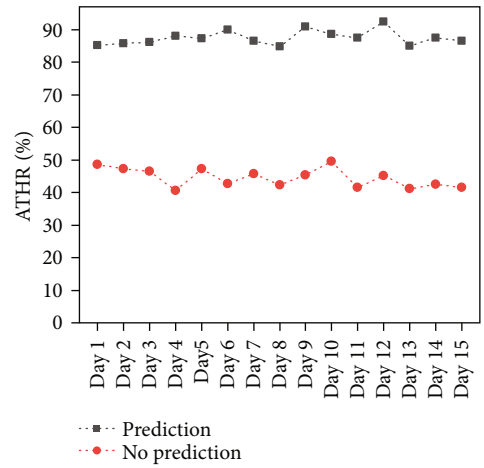


Figure 4: Average task hit rate of edge servers.

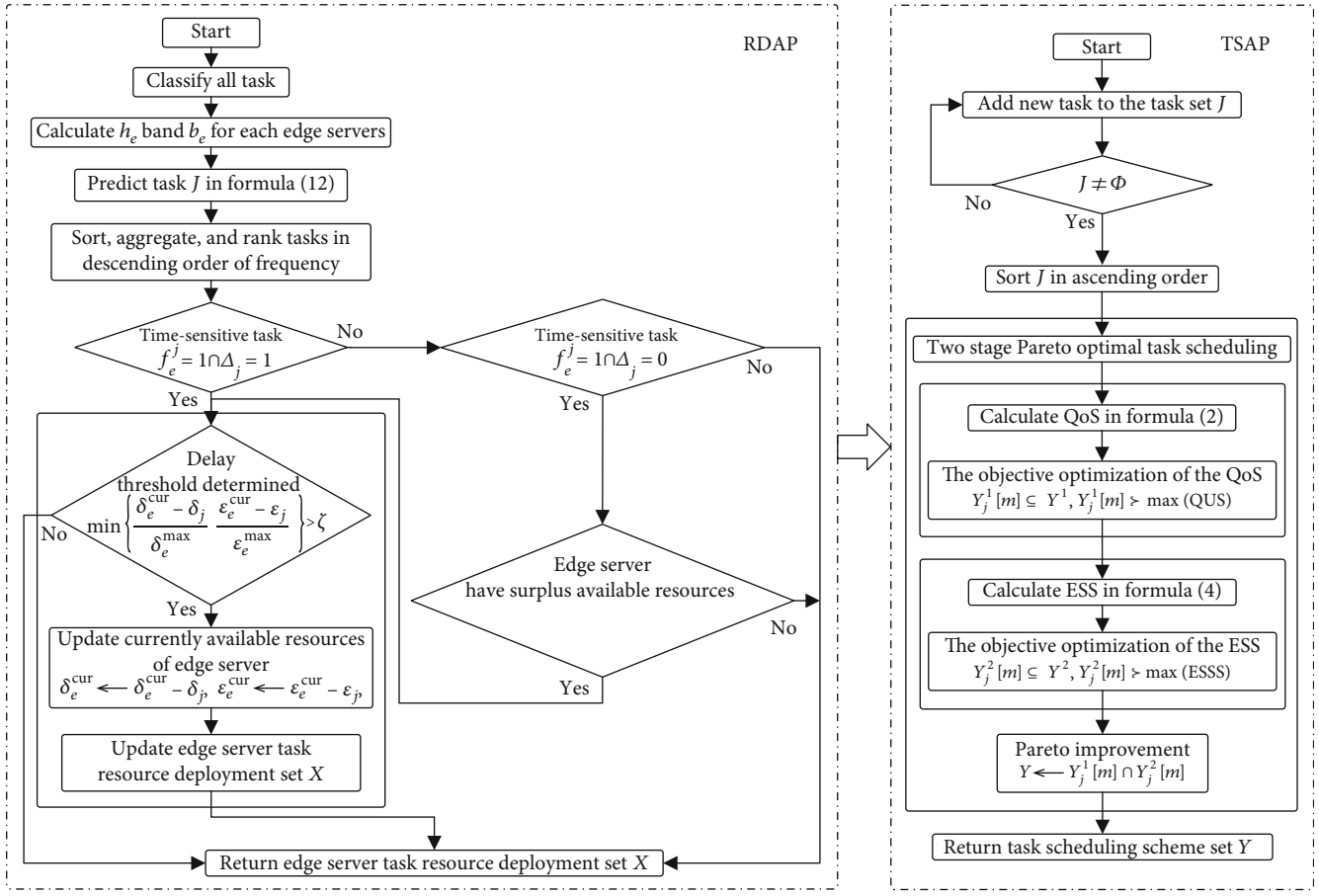


Figure 5: RDAP-TSAP.

of previous task prediction, the adjustment factor λ of the two-dimensional time series is set up to 0.27, and the random noise is ignored. The predicted result is compared with the actual number of tasks processed by the edge servers. As shown in Figure 3, the actual value of user tasks changes greatly, and the predicted value of tasks changes relatively gently. In general, the task prediction based on the two-dimensional time series is close to the actual value, the deviation of the average task number is less than 5%, and the accuracy of task prediction is high. The result can guide the cloud service center to optimize the task resource deployment of the edge servers.

The average task hit rate (ATHR) is used to evaluate the effect of resource deployment with task prediction. ATHR refers to the average value of the local execution proportion of tasks of each edge server. The proportion of local task execution of the edge server is the ratio of the edge server receiving user task requests and executing tasks locally (because it has the task resources required for task operation). The high ATHR value indicates that the cloud service center has a high accuracy rate of pushing the task resources based on the predicted result. The high rate of local execution of tasks in the edge servers is conducive to improving both QoS and ESS. In the edge-cloud collaborative computing environment, the cloud service center can effectively predict user tasks based on the two-dimensional time series, classify and aggregate them according to the prediction result, and optimize the task resources pushed to each edge server. It can improve the local execution rate of user tasks on the edge servers, and reduce the passive application of task resources to the cloud service center because the edge servers do not have the resources required for task operation. Through 15 consecutive days of experiments, using task prediction based on the two-dimensional time series to deploy the task resources required by the edge servers, the average task hit rate of the edge servers is very high. As shown in Figure 4, its ATHR value reaches 85.79% ~ 92.39%, which is much higher than the average task hit rate of less than 50% for resource deployment without task prediction.

6.2.2. Task Scheduling. We evaluate and analyze the task scheduling performance of the TSAP, RDAP-TSAP (combining RDAP and TSAP, as shown in Figure 5), BAO, and BAF from the three aspects of ATCT, OSSE, and TTDR. In the edge-cloud collaborative computing environment, to compare different task scheduling performances, the m-group task scheduling schemes of TSAP consider the values of 3, 5, 7, and 9, respectively.

(1) Average Task Completion Time (ATCT). The ATCT is the ratio of the total completion time of all user tasks to

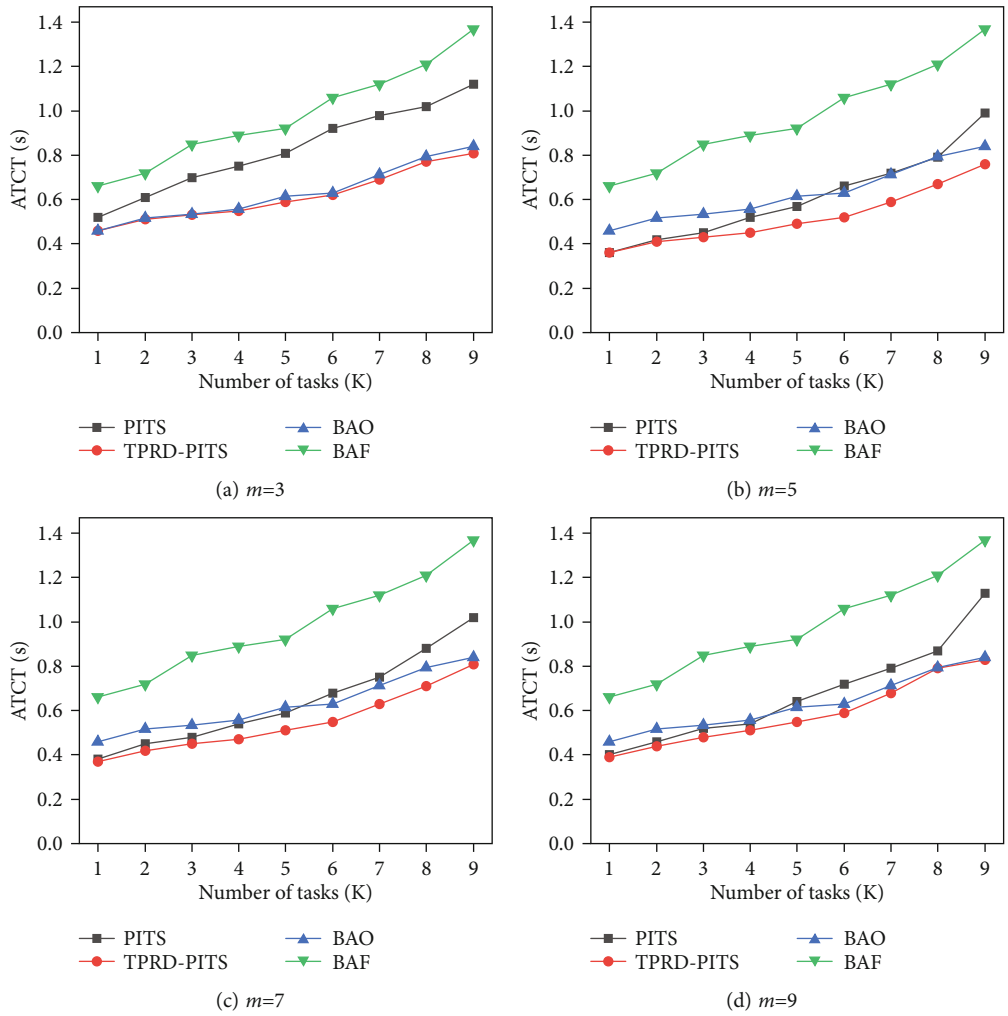


Figure 6: Average task completion time under different tasks.

the number of user tasks during the statistical period. The ATCT mainly considers the user service experience. The smaller the ATCT value, the shorter the average completion time of all tasks and the higher the quality of user experience. In the edge-cloud collaborative computing environment, the Zipf distribution α parameter of the user task is set to the default value, and the ATCT of the TSAP, RDAP-TSAP, BAO, and BAF algorithms are compared and analyzed by changing the task scale. As shown in Figure 6, with the increase of the number of user tasks, the ATCT values of the four algorithms increase and show an approximately linear trend. The ATCT values of RDAP-TSAP, BAO, and TSAP algorithms are significantly lower than that of BAF algorithm. The TSAP algorithm effectively reduces the ATCT by Pareto improvement on the two objectives of QoS and ESS, and its ATCT value is reduced by 29.69% on average compared with the BAF algorithm. The BAO algorithm jointly optimizes task caching and task offloading, and its ATCT value is reduced by 35.68% on average compared with the BAF algorithm. The RDAP-TSAP algorithm predicts the number of user task types, improves the accuracy of resources required by the cloud service center to push tasks to edge servers, and further reduces the ATCT by Pareto improvement of QoS and ESS. The ATCT value of RDAP-TSAP algorithm is 42.07% lower than the BAF algorithm, and 9.94% lower than the BAO algorithm. In addition, the TSAP and RDAP-TSAP algorithms have the best ATCT values when m =5. It shows that the smaller m value leads to the single task scheduling scheme, and the higher m value will increase the calculation cost of task scheduling itself, which will improve the ATCT.

Considering the influence of task Zipf distribution α parameter, the ATCT of the TSAP, RDAP-TSAP, BAO, and BAF algorithms will be further evaluated under certain user tasks. The number of user tasks is set up to 5k in the edge-cloud collaborative computing environment. As shown in Figure 7, as the value of the Zipf distribution α parameter increases, user tasks become more and more concentrated in the popular task types, and the ATCT values of the four scheduling algorithms decrease in varying degrees. Moreover, the TSAP and RDAP-TSAP algorithms have the best ATCT value when the m-group of task scheduling schemes is 5. Overall analysis, the RDAP-TSAP algorithm predicts the type and number of tasks, improves the accuracy of task resources push, increases the average task hit rate; and then integrates both QoS and ESS for Pareto improvement to

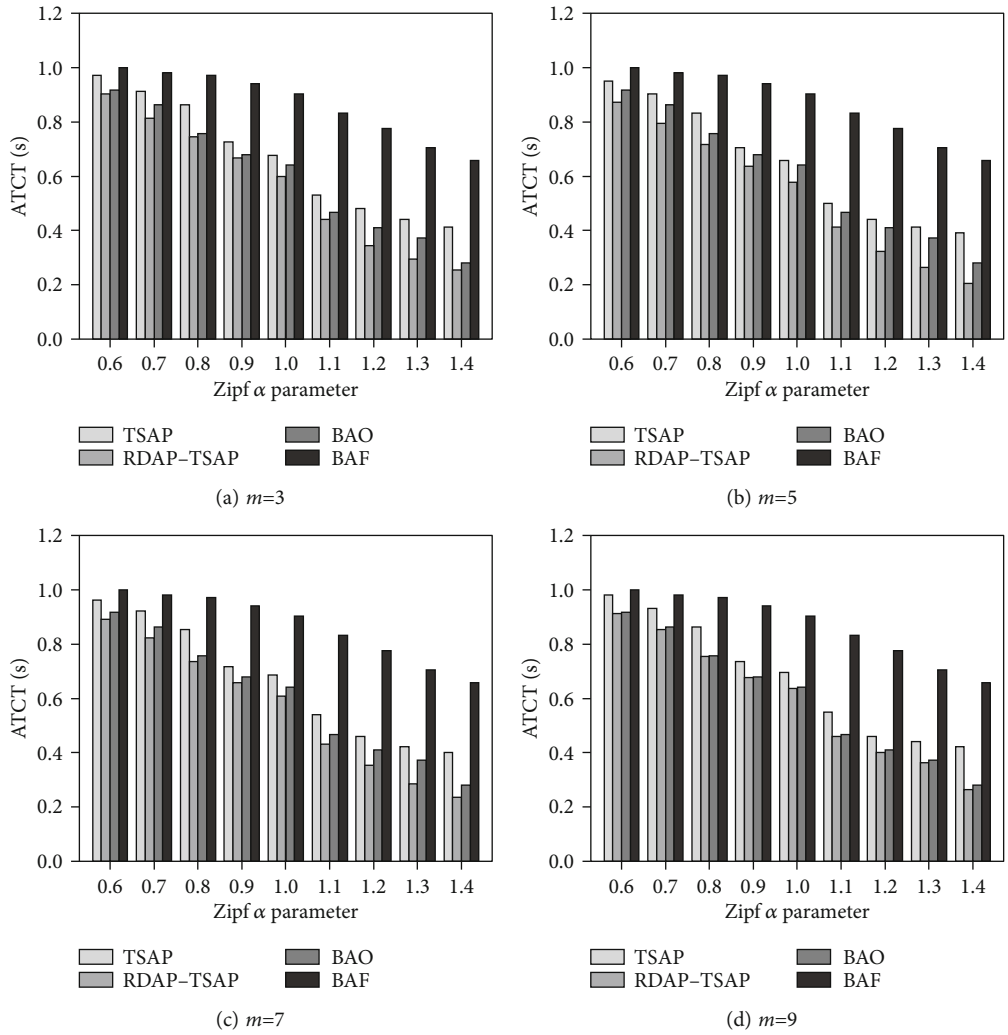


Figure 7: Average task completion time under different Zipf distribution α parameters.

optimize task scheduling. Its ATCT value is the best, with an average overall reduction of 34.94% compared with the BAF algorithm and 6.22% compared with the BAO algorithm. The TSAP algorithm benefits from two-stage Pareto improvement of QoS and ESS. In the absence of task prediction to optimize resource deployment, its ATCT value is overall reduced by 23.23% on average compared with the BAF algorithm.

(2) Overall System Service Effect (OSSE). The OSSE is the difference between system service revenue and system service consumption of the cloud service center and edge servers in the statistical period, which is the ESS value of normalization. The OSSE mainly considers the interests of service providers. The higher the system service revenue of providers and the lower their system service consumption, the higher its OSSE value. In the edge-cloud collaborative computing environment, the user tasks are Zipf distribution with default α parameter. The OSSE of the TSAP, RDAP-TSAP, BAO, and BAF algorithms are evaluated by gradually increasing the number of user tasks. As shown in Figure 8, the OSSE values of the four algorithms increase with the number of user tasks, showing an approximate logarithmic growth. Compared with the BA algorithm, the OSSE values of the RDAP-TSAP, BAO, and TSAP algorithms are significantly higher. Among them, the TSAP algorithm optimizes task scheduling through Pareto improvement of both QoS and ESS objectives, and improves the overall service effectiveness of the cloud service center and edge servers. Its OSSE value overall increases by 24.69% on average. The BAO algorithm jointly optimizes task caching and task offloading, and its OSSE value increases by 36.78% on average. The RDAP-TSAP algorithm predicts based on the type and number of tasks in the cloud service center, effectively pushes the task resources, improves the average task hit rate, and optimizes task scheduling based on the objectives of QoS and ESS at the edge servers to further increase the OSSE value. Its OSSE value overall increases by 49.55% on average. In addition, compared with the BAO algorithm, the OSSE value of the RDAP-TSAP algorithm has an overall increase of 10.43% on average. Moreover, similar to the previous ATCT analysis, the OSSE values of the TSAP and RDAP-TSAP algorithms are optimal when m=5. It indicates that a small value of m leads to a single task scheduling scheme

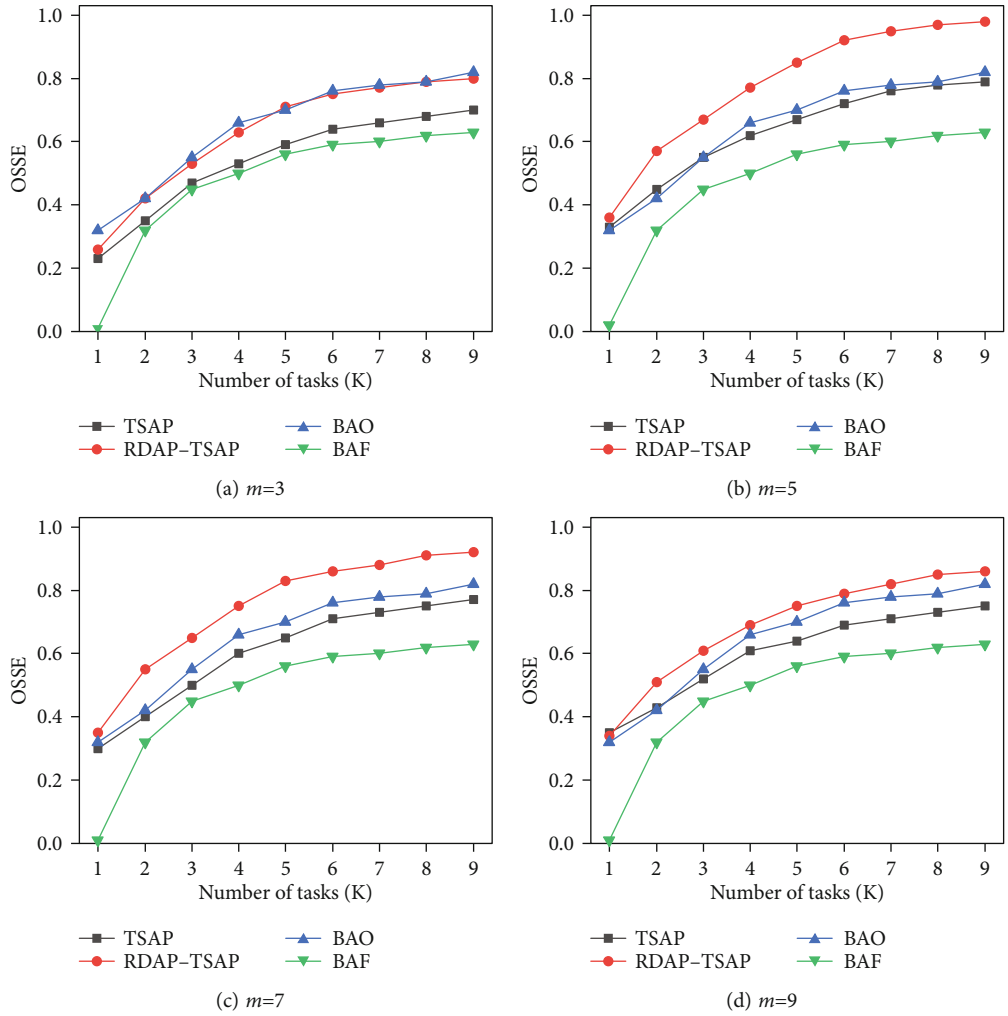


Figure 8: Overall system service effect under different tasks.

and is under-optimized, and a large value of m increases the computational overhead of task scheduling, which will reduce the OSSE.

Considering the influence of task Zipf distribution α parameter, the OSSE of the TSAP, RDAP-TSAP, BAO, and BAF algorithms will be further evaluated under certain user tasks. In Figure 9, the number of user tasks is 5k in the edge-cloud collaborative computing environment. As the value of the Zipf distribution α parameter increases, the user tasks gradually tend to be more popular task types and decrease the number of task resources required by edge servers, and the OSSE values of the four algorithms are greatly improved. And similar to the previous ATCT analysis, the OSSE values of the TSAP and RDAP-TSAP algorithms are optimal when the value of the m-group task scheduling schemes is 5. In general, the RDAP-TSAP algorithm predicts the tasks, improves the accuracy of task resources required to push edge servers, and increases the average task hit rate. Moreover, through the two-stage comprehensive optimization considering the QoS and ESS, the OSSE value of RDAP-TSAP algorithm is the best. Its OSSE value is 39.54% higher than the BAF algorithm, and 11.91% higher than the BAO algorithm. In the absence of task prediction, the TSAP algorithm uses Pareto optimization of the QoS and ESS, and its OSSE value is overall increased by 22.62% on average compared to the BAF algorithm.

(3) Total Task Delay Rate (TTDR). The TTDR is the proportion of tasks whose task response time exceeds 3 times the average response time of similar tasks in the statistical time period. The TTDR affects the user's quality of service and the effect of system service [53–55]. The low TTDR value can help reduce the average task completion time of users and enhance the QoS, and also help reduce the system energy consumption and improve the ESS. In the edge-cloud collaborative computing environment, user tasks are Zipf distribution with default α parameter, and the value of the m-group task scheduling schemes is 5. The TTDR of the TSAP, RDAP-TSAP, BAO, and BAF algorithms are evaluated by gradually increasing the number of user tasks. As shown in Figure 10, with the number of user tasks increasing, the TTDR values of the four algorithms increase. In general, the RDAP-TSAP algorithm is the best, the BAO and TSAP algorithm is the second, and the BAF algorithm is the worst. The RDAP-TSAP algorithm predicts tasks, classifies and aggregates tasks to reduce the number of task

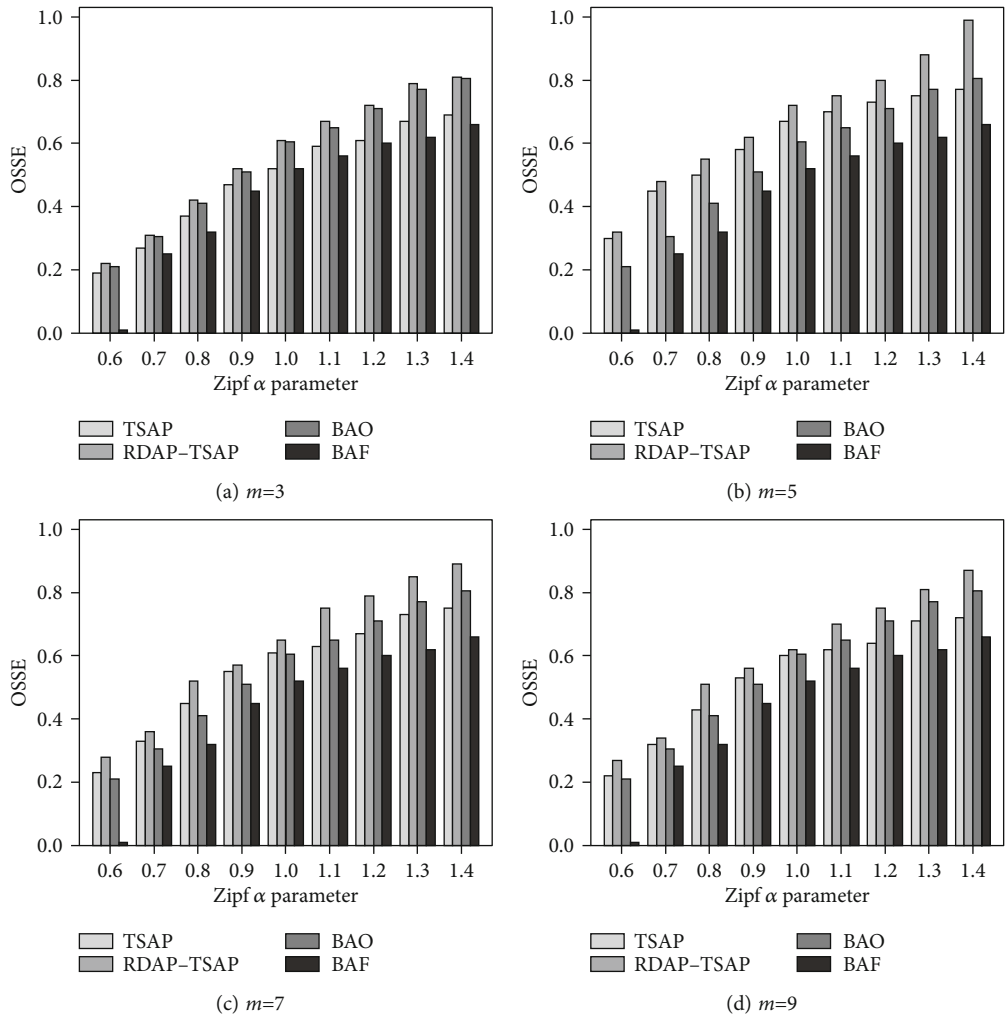


Figure 9: Overall system service effect under different Zipf distribution α parameters.

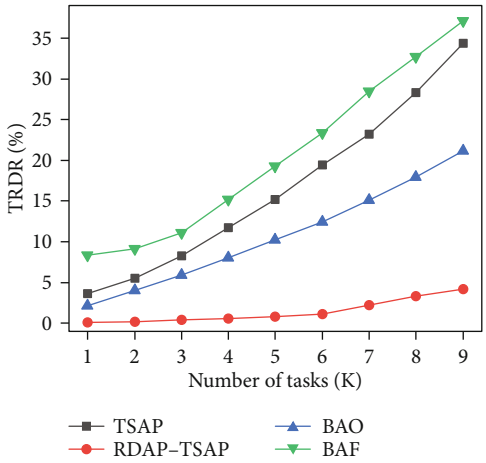


Figure 10: Total task delay rate under different tasks.

resources pushed to the edge servers, determines the delay threshold during resource deployment, and considers the available resources of the edge servers to reduce the probability of task delay. As the number of tasks increases, the total task delay rate increases slowly, and the TTDR value is controlled between 0.12% and 4.17%. The BAF, TSAP, and BAO algorithms lack task prediction in task scheduling and do not handle possible task delays. Their TTDR values are high, and the total task delay rate increases linearly with the increase of tasks. The BAF algorithm has the highest TTDR value, ranging from 8.37% to 37.12%. Because the TSAP algorithm performs Pareto improvement for task scheduling by integrating the objectives of QoS and ESS, it can reduce the average service completion time of tasks, and its TTDR value is relatively low, ranging from 3.67% to 34.39%. The BAO algorithm jointly optimizes task caching and task offloading, and its TTDR value is 2.17% to 21.19%.

Considering the influence of task Zipf distribution α parameter, the TTDR of the TSAP, RDAP-TSAP, BAO, and BAF algorithms is further evaluated under the condition of unchanged user tasks. In the edge-cloud collaborative computing environment, the number of user tasks is set up to 5k, and the value of the m-group task scheduling schemes is 5. The changes in their TTDR are shown in Figure 11. As the value of Zipf distribution α parameter increases, user

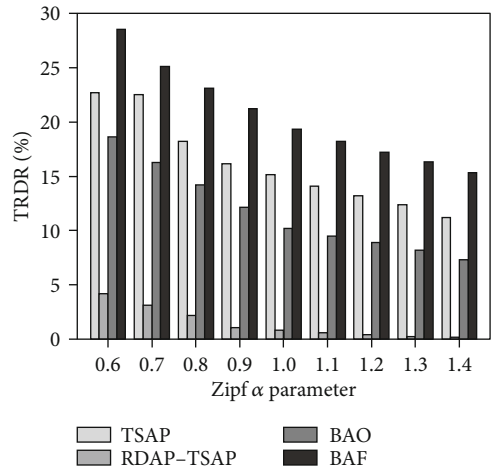


Figure 11: Total task delay rate under different Zipf distribution α parameters.

tasks are more and more concentrated on task types with high popularity, and the task resource types required by edge servers are reduced. In general, the TTDR of the TSAP, RDAP-TSAP, BAO, and BAF algorithms has decreased to varying degrees. The RDAP-TSAP algorithm predicts, classifies, and aggregates user tasks to reduce the task resources required to push to the edge servers. During resource deployment, the delay threshold is judged to effectively reduce the occurrence of task delays, and then the Pareto optimizes task scheduling to make its TTDR always at a low level, with values ranging from 0.15% to 4.17%, which is an average reduction of 19.05% compared to the BAF algorithm, and an average reduction of 10.29% compared to the BAO algorithm. The TSAP algorithm lacks user task prediction and does not optimize task resource deployment of edge servers. However, it integrates the QoS and ESS for Pareto improvement during user task scheduling, and its TTDR value is relatively high, ranging from 11.22% to 22.71%, which is 4.28% lower than the BAF algorithm on average. The BAO algorithm jointly optimizes task caching and task offloading, and its TTDR value is 7.29% to 18.61%, which is an average reduction of 8.76% compared to the BAF algorithm. The TTDR value of the BAF algorithm is the highest, ranging from 15.31% to 28.51%, with an average value of 20.47%.

7. Conclusion

In this paper, we focused on resource deployment and task scheduling optimization using task prediction in edge-cloud collaborative computing for users and service providers. We presented an edge-cloud collaborative computing framework for edge-cloud IoT, and an approach for task prediction based on the two-dimensional time series. Then, for service-oriented tasks, we described our proposed RDAP. Using our approach, user tasks can be predicted based on the two-dimensional time series in the cloud service center, user tasks classified and aggregated, and the resources required for task operation pushed to the edge servers to improve average task hit rate and reduce server resource overhead. We also described our designed TSAP. In our approach, at the edge servers, the random greedy approximation algorithm is used to make Pareto improvement to both QoS and ESS, and the tangent point or intersection point of the two objective curves is used to optimize the task scheduling. Finally, the experimental evaluation shows that the RDAP-TSAP algorithm combining RDAP and TSAP realizes the comprehensive optimization of user's quality of service and effect of system service. Based on improving the average user task hit rate, the RDAP-TSAP algorithm has better ATCT, OSSE, and TTDR values than the TSAP, BAF, and BAO algorithms.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Key Program of the National Natural Science Foundation of China (61632009), Hunan Provincial Natural Science Foundation of China (2019JJ70057), Natural Science Foundation of Guangdong Province (2017A030308006), National Key Research and Development Program of China (2020YFB1005804), and Fundamental Research Funds for the Central Universities of Central South University (2018zzts180). The work of Kim-Kwang Raymond Choo is supported only by the Cloud Technology Endowed Professorship.

References

[1] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[2] X. Zhou, X. Yang, J. Ma, and K. Wang, "Energy efficient smart routing based on link correlation mining for wireless edge computing in IoT," *IEEE Internet of Things Journal*, 2021.

[3] Y. Xu, X. Yan, Y. Wu, Y. Hu, W. Liang, and J. Zhang, "Hierarchical bidirectional RNN for safety-enhanced B5G heterogeneous networks," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 4, pp. 2946–2957, 2021.

[4] J. Zhu, D. S. Chan, P. M. M. Suryanarayana, R. Natarajan, and H. Hu, "Improving web sites performance using edge servers in fog computing architecture," in *2013 IEEE Seventh International Symposium on Service-Oriented System Engineering*, pp. 320–323, San Francisco, CA, USA, 2013.

[5] Y. Zhang and Y. Zhou, "Transparent computing: spatio-temporal extension on von neumann architecture for cloud services," *Journal of Tsinghua Science and Technology*, vol. 18, no. 1, pp. 10–21, 2013.

[6] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[7] T. Wang, Y. Mei, W. Jia, X. Zheng, G. Wang, and M. Xie, "Edge-based differential privacy computing for sensor-cloud systems," *Journal of Parallel and Distributed Computing*, vol. 136, pp. 75–85, 2020.

[8] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 1, no. 2, pp. 89–103, 2015.

[9] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–38, 2021.

[10] J. Zhang, Y. Wang, M. Long, J. Wang, and H. Wang, "Predictive recurrent networks for seasonal spatiotemporal data with applications to urban computing," *Chinese Journal of Computers*, vol. 43, no. 2, pp. 286–302, 2020.

[11] X. Zhu, Y. Luo, A. Liu, W. Tang, and M. Z. A. Bhuiyan, "A deep learning-based mobile crowdsensing scheme by predicting vehicle mobility," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4648–4659, 2021.

[12] Y. Sun, X. Yin, J. Jiang et al., "CS2P: improving video bitrate selection and adaptation with data-driven throughput prediction," in *Proceedings of the 2016 ACM SIGCOMM Conference*, pp. 272–285, Florianopolis, Brazil, 2016.

[13] X. Zhou, X. Xu, W. Liang et al., "Intelligent small object detection for digital twin in smart manufacturing with industrial cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 2, pp. 1377–1386, 2022.

[14] X. Zhu, Y. Luo, A. Liu, M. Z. A. Bhuiyan, and S. Zhang, "Multiagent deep reinforcement learning for vehicular computation Offloading in IoT," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9763–9773, 2021.

[15] T. Wang, Y. Lu, J. Wang, H. Dai, X. Zheng, and W. Jia, "EIHDP: edge-intelligent hierarchical dynamic pricing based on cloud-edge-client collaboration for IoT systems," *IEEE Transactions on Computers*, vol. 70, no. 8, pp. 1285–1298, 2021.

[16] X. Zhou, X. Xu, W. Liang, Z. Zeng, and Z. Yan, "Deep-learning-enhanced multitarget detection for end-edge-cloud surveillance in smart IoT," *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 12588–12596, 2021.

[17] X. Yan, Y. Xu, X. Xing, B. Cui, Z. Guo, and T. Guo, "Trustworthy network anomaly detection based on an adaptive learning rate and momentum in IIoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6182–6192, 2020.

[18] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[19] H. Mao, M. Schwarzkoopf, S. B. Venkatakrishnan, Z. Meng, and M. Alizadeh, "Learning scheduling algorithms for data processing clusters," in *SIGCOMM '19: Proceedings of the ACM Special Interest Group on Data Communication*, pp. 270–288, Beijing, China, 2019.

[20] V. Jalaparti, P. Bodik, I. Menache, S. Rao, K. Makarychev, and M. Caesar, "Network-aware scheduling for data-parallel jobs: plan when you can," *ACM SIGCOMM Computer Communication Review*, vol. 45, pp. 407–420, 2015.

[21] X. Ren, G. Ananthanarayanan, A. Wierman, and M. Y. Hopper, "Decentralized speculation-aware cluster scheduling at scale," in *SIGCOMM '15: Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, pp. 379–392, London, UK, 2015.

[22] A. Alashaikh, E. Alanazi, and A. Al-Fuqaha, "A survey on the use of preferences for virtual machine placement in cloud data centers," *ACM Computing Surveys*, vol. 54, no. 5, pp. 1–39, 2022.

[23] Y. Xu, J. Ren, Y. Zhang, C. Zhang, B. Shen, and Y. Zhang, "Blockchain empowered arbitrable data auditing scheme for network storage as a service," *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 289–300, 2020.

[24] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 144–153, Dallas, TX, USA, 2019.

[25] A. Chung, J. Park, and G. Robert, "Stratus: cost-aware container scheduling in the public cloud," in *SoCC '18: Proceedings of the ACM Symposium on Cloud Computing*, pp. 121–134, New York, NY, USA, 2018.

[26] X. Liu, J. Li, Z. Yang, and Z. Li, "A task collaborative execution policy in mobile cloud computing," *Chinese Journal of Computers*, vol. 40, no. 2, pp. 364–377, 2017.

[27] R. Nishtala, V. Petrucci, P. Carpenter, and M. Sjlander, "Twig: Multi-agent task management for colocated latency-critical cloud services," in *2020 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pp. 167–169, San Diego, CA, USA, 2020.

[28] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795–2808, 2016.

[29] T. G. Rodrigues, K. Suto, H. Nishiyama, and N. Kato, "Hybrid method for minimizing service delay in edge cloud computing through vm migration and transmission power control," *IEEE Transactions on Computers*, vol. 66, no. 5, pp. 810–819, 2017.

[30] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3590–3605, 2016.

[31] M. Chen and Y. Hao, "Task offloading for mobile edge computing in software defined ultra-dense network," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 587–597, 2018.

[32] J. He, D. Zhang, Y. Zhou, and Y. Zhang, "A truthful online mechanism for collaborative computation offloading in mobile edge computing," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 7, pp. 4832–4841, 2020.

[33] C. Wang, C. Liang, F. R. Yu, Q. Chen, and L. Tang, "Computation offloading and resource allocation in wireless cellular networks with mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 16, no. 8, pp. 4924–4938, 2017.

[34] T. Wang, M. Bhuiyan, G. Wang, L. Qi, J. Wu, and T. Hayajneh, "Preserving balance between privacy and data integrity in edge-assisted internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2679–2689, 2020.

[35] X. Zhou, W. Liang, S. Shimizu, J. Ma, and Q. Jin, "Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5790–5798, 2021.

[36] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: partial computation o_oading using dynamic voltage scaling," *IEEE Transactions on Communications*, vol. 64, no. 10, pp. 4268–4282, 2016.

[37] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. S. Quek, "O_oading in mobile edge computing: task allocation and computational frequency scaling," *IEEE Transactions on Communications*, vol. 65, no. 8, pp. 3571–3584, 2017.

[38] J. Zhang, X. Hu, Z. Ning et al., "Energy-latency tradeoff for energy-aware offloading in mobile edge computing networks," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2633–2645, 2018.

[39] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 1784–1797, 2018.

[40] Y. Ding, C. Liu, X. Zhou, Z. Liu, and Z. Tang, "A code-oriented partitioning computation offloading strategy for multiple users and multiple mobile edge computing servers," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4800–4810, 2020.

[41] Y. Xu, C. Zhang, G. Wang, Z. Qin, and Q. Zeng, "A blockchain-enabled deduplicatable data auditing mechanism for network storage services," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1421–1432, 2021.

[42] Z. Ge, J. Wang, C. Jiang et al., "Analysis of resource utilization of co-located clusters," *Chinese Journal of Computers*, vol. 43, no. 6, pp. 1103–1122, 2020.

[43] Aliyun, "Aliyun edge node service," (2021), http://www.aliyun.com/product/ens/.

[44] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[45] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: evidence and implications," in *IEEE INFOCOM '99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320)*, pp. 126–134, New York, NY, USA, 1999.

[46] X. Yan, Y. Xu, B. Cui, S. Zhang, T. Guo, and C. Li, "Learning URL embedding for malicious website detection," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6673–6681, 2020.

[47] Y. Xu, Q. Zeng, G. Wang, C. Zhang, J. Ren, and Y. Zhang, "An efficient privacy-enhanced attribute-based access control mechanism," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 5, pp. 1–10, 2020.

[48] X. Wang, Z. Wang, and F. Li, "Cache location selected algorithm for information-centric networking," *Journal of national university of defense technology*, vol. 41, no. 1, pp. 152–160, 2019.

[49] X. Zhou, W. Liang, K. Wang, R. Huang, and Q. Jin, "Academic influence aware and multidimensional network analysis for research collaboration navigation based on scholarly big data," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 1, pp. 246–257, 2021.

[50] J. Xu, L. Chen, and P. Zhou, "Joint service caching and task offloading for mobile edge computing in dense networks," in *IEEE INFOCOM 2018*, pp. 207–215, New York, NY, USA, 2018.

[51] M. Su, G. Wang, and R. Li, "Multidimensional qos cloud computing resource scheduling method based on stakeholder perspective," *The Journal of Communication*, vol. 40, no. 6, pp. 103–115, 2019.

[52] Y. Xu, C. Zhang, Q. Zeng, G. Wang, J. Ren, and Y. Zhang, "Blockchain-enabled accountability mechanism against information leakage in vertical industry services," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1202–1213, 2021.

[53] Q. Lu, X. Xu, L. Bass, L. Zhu, and W. Zhang, "A tail-tolerant cloud API wrapper," *IEEE Software*, vol. 32, no. 1, pp. 76–82, 2015.

[54] X. Zhou, Y. Li, and W. Liang, "CNN-RNN based intelligent recommendation for online medical pre-diagnosis support," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 3, pp. 912–921, 2021.

[55] C. Zhang, Y. Xu, Y. Hu, J. Wu, J. Ren, and Y. Zhang, "A blockchain-based multi-cloud storage data auditing scheme to locate faults," *Computing*, 2021.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 1493768, 16 pages
https://doi.org/10.1155/2022/1493768

WILEY | Hindawi

Research Article

Public Integrity Auditing of Shared Encrypted Data within Cloud Storage Group

Chunxia Han and Linjie Wang

School of Data Science, Tongren University, Tongren, Guizhou 554300, China

Correspondence should be addressed to Linjie Wang; wanglinjie_66@hotmail.com

Received 1 November 2021; Revised 26 November 2021; Accepted 25 January 2022; Published 25 February 2022

Academic Editor: Yingjie Wang

The pandemic of COVID-19 has posed a severe challenge to the traditional on-site centralized development projects; people therefore have to share data in a group by the cloud storage server and develop projects at home. The cloud server is untrustworthy, although it supplies the powerful computing capability and abundant storage space; so far wide research has been proposed to verify data integrity. Therefore, how to leverage the cloud server and ensure the integrity of the data (especially the encrypted data) stored on the remote cloud devices remains an issue for the clients. To address this issue, we utilize the technique of homomorphic hash function to implement reencryption ciphertext blocks and introduce a certificateless signature scheme for the integrity verification of encrypted data shared within a group. A detailed challenge-and-response game represents that the proposed scheme can preserve encrypted data blocks integrity against the internal/external attacker and malicious cloud service servers. We give the theoretical and experimental performance analysis of the scheme and exhibit that the scheme is efficient and practical.

1. Introduction

To date, cloud storage service has been an efficient paradigm for storing and sharing data information and a cooperation platform for staff to collaborate in many companies. Once a project manager can upload tasks to the server, each project participant can access, download, and modify the corresponding files through the network without any geographical restriction. Especially with the travel restrictions caused by the COVID-19 pandemic, people can only stay in their own homes and work through the Internet; thereby the use of cloud services for task cooperation and sharing has become particularly important. In the real world, Dropbox for Business [1], TortoiseSVN [2], and Google Drive already have become cloud service platforms for employees to share and collaborate online.

However, the prerequisite for this type of application to facilitate many company staff to work together is whether the cloud server provider (CSP) can make sure that the data is retained intact. In the field of cloud services, there is a multitude of inevitable internal and external attacks [3], as a slice of examples, the failure of software or hardware, illegal access, and deliberate deletion or corruption of the outsourced data, resulting in unreliable cloud services. Owing to the existence of these above attacks, the integrity of data is destroyed, which will inevitably reduce the availability and storage significance of data. This paper focuses on the integrity of data stored in the cloud.

For decades, to address the integrity verification of data, an army of studies on remote data integrity checking have been proposed by papers [4–16], and these schemes give efficient approaches to verify the integrity of outsourced data on cloud server without downloading them. However, all the above solutions are focused on the integrity auditing for individual data without involving the situation of sharing data in a group. How to verify the integrity of the shared data in a group is an interesting and essential task in the cloud server, which is also another item for the cloud service.

Remote data integrity verification is a technology that, for the data stored in the virtual cloud server, there is no

need to download the entire file locally to check the integrity of data. When a project with data attached to it is uploaded to a cloud server and shared among multiple engineers, some new challenges emerge and these challenges cannot be solved well with existing individual data integrity verification solutions. According to the above scenario, the project with data attached is divided into blocks and sent to the engineers of the project group, and different engineers will output different block tags in the same block. When a block is modified by the engineers of the project group, the new block tags will be regenerated. In a project group, all engineers will either online or offline compile their tasks, but no matter which kind they are required to store the results of the day's tasks in the cloud server and generate block tags for checking. To ensure that each engineer is honest to compile the project with the given data, the project manager must act as a verifier to verify the integrity of the data from time to time. When the verifier wants to audit the integrity of raw data, it needs to aggregate all tags with the engineer's identity information. The process of verification is more complicated and brings a significant volume of calculation [17–24]; these protocols thereby are not valid for the case of data sharing in a group.

When the data is shared among the engineers in the project group, also other challenges appear where some engineers in the group maybe withdraw from the group due to some special circumstances, such as being transferred to another project group or misbehaving. As a result of the above situations, the tags generated by the revoked user are invalid and need to be renewed by the other legitimate members. In addition, the data in the shared group also needs to be updated frequently, which also leads the tags to be changed constantly. For security reasons, if the identity of an engineer is revoked, all data as well as the corresponding tags, which belong to the revoked members previously, still have to be renewed by the existing user in the group. According to common sense, when an engineer exits the project, then its task will be transferred to other engineers, and its identity information in the project will be revoked; that is, its public/private key for participating in the project will become invalid. Considering the fact that the shared data is not stored on local devices, the traditional way is to download all data previously generated by the revoked engineer and ask an existing engineer to renew the tags and finally upload the new tags to the cloud server again. This operation can safely transfer the task to the engineer existing in the task, but it may significantly increase the existing engineer communication cost and calculation resources, especially when a considerable volume of the blocks needs to frequently change and update. To overcome the above drawbacks, the execution of the verification operations should be outsourced to CSP instead of execution by the existing engineers. Besides, integrity verification of shared data can be verified not only by the members of the shared data group but also by everyone who wants to leverage the data blocks in the cloud service. As a result, it is of tremendous significance that the scheme to be proposed can meet the public verification with the help of CSP.

At present, plenty of integrity verification schemes for shared data in this group have been put forward. Most of them [25–29] focus on the PKI technology based on the trustworthiness of certificate authority (CA), where it is difficult to find a trusted CA. Others are identity-based [28, 30] remote data integrity verification protocols, which rely on the private key generator (PKG) to generate all private keys. However, this approach suffers from a key escrow problem. Therefore, how to efficiently verify the integrity of outsourced data in a shared group by a public verifier and transfer the revoked members' data to existing members without downloading the data from the cloud service, as well as solving the key escrow and certificate management issues, is a challenging task.

Reviewing the existing protocol solutions, we mainly focus on the integrity verification for the encrypted shared data in a group. In this paper, we assume that there is an encrypted business project, which is divided into numerous encrypted subprojects, and it needs plenty of engineers to participate in development. A project manager, who invites the engineers to a temporary project group, takes charge of the system parameters and encrypts the raw project. Then the project blocks encrypted with public keys of specified members in the group are uploaded to the cloud service so that the engineers within the group can modify and upload subprojects compiled online or offline. If this is a big project with plenty of engineers in the project group, there are some issues to be addressed efficiently, for example, the integrity verification after legitimate changes to subprojects under development, the members revocation problem, and the entry of new members.

1.1. Contributions. To overcome the disadvantages of previous schemes and address the aforementioned issues, we propose a new remote data possession checking scheme for encrypted shared data group. The contributions of the proposed scheme are presented as follows:

(i) We propose a new remote data possession checking scheme to audit encrypted shared data in each group, in which the certificateless public key system is utilized as an underlying encryption mechanism, and the homomorphism hash approach is used to regenerate the ciphertext to improve the efficiency of member revocation scheme.

(ii) We then construct a public auditing scheme for verifying the integrity of encrypted data in the cloud service provider based on the corresponding certificateless authentication tag aggregation.

(iii) We design a ciphertext conversion scheme which leverages a homomorphic hash function to convert the ciphertext of the revoked member into the ciphertext of the existing member. The scheme has been implemented and the results are more efficient compared to state-of-the-art protocols.

(iv) We have proven the security of the proposed scheme which is based on the stability of CDH and DL assumptions by simulating a challenge-and-

response game involving two players: a challenger and an adversary.

1.2. Organization. The rest of this paper is organized as follows. Section 2 discusses the prior work done in verifying the integrity of group shared data. Section 3 introduces the preliminaries and Section 4 defines the problem statement which includes system model, design goals, outline of the scheme, and the secure model. The detailed construction of our scheme is presented in Section 5. The proposed scheme is simulated and a challenge-and-response secure model is formalized in Section 6, and we assess the efficiency of the proposed scheme in Section 7 based on the computation cost of tag generation and verification, communication cost analysis, and vocation analysis compared with the existing schemes. The conclusion of this paper is presented in Section 8.

2. Related Work

Since Deswarte et al. [4] first proposed a scheme for checking the integrity of data stored on remote virtual cloud servers; so far, a number of auditing schemes have been proposed. Among the proposed schemes, they can be generally divided into two directions: Provable Data Possession (PDP) [5] and Proofs of Retrievability (POR) [23].

The PDP scheme is proposed by Ateniese et al. [5] based on RSA signature and sampling strategies to improve the efficiency of integrity checking without retrieving the whole file. However, there is a limitation for this scheme; that is, it is only suitable for the auditing of static data files. To overcome this limitation, Ateniese et al. [31] presented the revised version of PDP based on symmetric encryption to efficiently address the dynamic checking issues instead of handling the insertion operation. Erway et al. [18] gave a dynamic provable data possession (DPDP) scheme, which supports full data dynamic operations to solve the insertion operation and improve the verification efficiency by leveraging the authenticated skip list.

To support fully dynamic data, Wang et al. [32], Erway et al. [18], and Zhu et al. [33] successively proposed schemes to construct auditing mechanisms supporting fully dynamic data, respectively. To realize public verification and dynamic data operation, Liu et al. [34] gave a dynamic public auditing scheme based on the Merkle Hash Tree (MHT), in which the block tags are generated by the data owners, and this incurs the increase of communication and calculation cost. To overcome this drawback, a scheme [35] to solve the heavy calculation burden on the data owner side at the expense of data owner's privacy has been proposed, in which both tag generation and integrity verification are implemented by the cloud server. The issue of privacy-preserving in public auditing has been addressed, in which the data blocks are blinded by a data owner before generating signatures by the third party [36]. In another related research, to avoid the certificate management problem of PKI, some PDP schemes based on Identity-Based Signature (IBS) [37, 38] were proposed. The major problem of IBS is the key escrow, which is solved by a certificateless-based signatures PDP scheme [39].

Similar to PDP, the POR is another approach introduced by Juels et al. [23] to audit the integrity of remote data stored on the cloud service. An improved POR scheme is given by Wang et al. [10] to authenticate block tags, in which a security proof is revised in their previous work. Based on the previous works of Erway et al. [18] and Ateniese et al. [5], a generic framework DPOR is proposed by Etemad et al. [40] to store call updated information in the logs. Apart from the aforementioned protocols, some other publicly verifiable protocols are published. Hao et al. [41] gave a public verification without including a TPA, and Shen et al. [42] solved the loss of private key for auditing issue. Wu et al. [43] introduced a time encapsulated POR protocol that could check the integrity of data and timestamp by verifier.

All schemes mentioned above mainly devote themselves to verifying the integrity of individual data. Since Wang et al. [44] proposed a scheme for auditing the integrity of data shared in a group in 2012, a succession of verification schemes for sharing data in a group have been proposed [7, 26–29, 45]. Among these schemes, [26, 27, 29, 45] represented PDP schemes for group data based on the signatures, respectively, and all of these schemes are more or less deficient in efficiency and revocation. To solve the multiuser modification problem of blocks, [28] based on PKI mechanism proposed a PDP scheme of polynomial authentication tags, which led to a heavy burden of certificate management. Recently, Li et al. [7] based on certificateless mechanism proposed a public integrity checking of group shared data on cloud storage, which changes the data tags of the revoked member into the existing member's tags. However, the data within the group in the scheme is all plaintext, which cannot satisfy the situation that the shared data in the group is ciphertext. According to all references mentioned above, although there are numerous schemes that can solve the problems of user adding and revocation in a shared group, on the premise of integrity auditing, there is no verification research on the integrity of encrypted data in a shared group. Therefore, we devote to designing a scheme for the integrity verification of encrypted data group in cloud service, which not only satisfies the member addition and revocation but also decreases the computational burden of challenge proof on the client side with the help of CSP.

3. Preliminaries

3.1. Bilinear Maps. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups of prime order p, and let g be a generator of $\mathbb{G}_1$. A bilinear map $e: \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$ has the following properties:

(1) Computability: there exists an efficient algorithm to compute map e.

(2) Bilinearity: for all $u, v \in \mathbb{G}_1$ and $a, b \in Z_p$, $e(u^a, v^b) = e(u, v)^{ab}$.

(3) Nondegeneracy: there exists a point g such that $e(g, g) \neq 1$.

3.2. Complexity Assumptions. In our scheme, the security is based on the following security assumptions.

Definition 1 (Computational Diffie-Hellman (CHE) Problem). Let $a, b \in Z_p^*$; given the tuple $g, g^a, g^b \in \mathbb{G}_1$ as input, output $g^{ab} \in \mathbb{G}_1$.

Assumption 2 (Computational Diffie-Hellman (CHE)). For any probabilistic polynomial time (PPT) algorithm $\mathcal{A}$, $\Pr[\mathcal{A}(g, g^a, g^b) \longrightarrow g^{ab}]$ is negligible, where $g, g^a, g^b \leftarrow \mathbb{G}_1$.

Definition 3 (Discrete Logarithm (DL) Problem). Let g be a generator of $\mathbb{G}_1$; given the tuple (g, g^a) as input, output $a \in Z_p^*$.

Assumption 4 (Discrete Logarithm (DL)). For any probabilistic polynomial time (PPT) algorithm $\mathcal{A}$, $\Pr[\mathcal{A}(g, g^a \longrightarrow (a))]$ is negligible, where $a \leftarrow Z_p^*$.

3.3. Homomorphic Hash Function. For a finite field F_n and a multiplicative group Z_p of order p, a family of homomorphic hash functions are a collection $\mathcal{H} = \{h_i : F_n \longrightarrow Z_q\}$, where i is the index yielded by an efficient algorithm. A homomorphic hash function [46] consists of the following properties:

(1) One way: given $\mathbf{x} \in F^n$ and an index i, there is no polynomial time adversary which can find a $h_i^{-1}(\mathbf{x})$.

(2) Collision resistance: given an index i, it is hard (computationally infeasible) to find two vectors $\mathbf{x}, \mathbf{y} \in F^n (\mathbf{x} \neq \mathbf{y})$ for which $h_i(\mathbf{x}) = h_i(\mathbf{y})$.

(3) Homomorphism: given an index i and any $\mathbf{x}, \mathbf{y} \in F^n (\mathbf{x} \neq \mathbf{y})$ $h_i(\mathbf{x} \circ \mathbf{y}) = h_i(\mathbf{x}) \circ h_i(\mathbf{y})$, "$\circ$" is either a "$\cdot$" or a "+."

4. Problem Statement

In this section, we show the system model and secure model and illustrate the design goals and the outline of our proposed scheme.

4.1. System Model. Similar to [7, 27, 29], we combine the cloud architecture with an example of sharing and developing encrypted files by the staffs of a company that are in the same group or department. The system model consists of three major entities: project group (i.e., members involved in the project), cloud service provider (CSP), and public verifier, and the relationship and the interaction situation among them are represented in Figure 1.

Project group consists of a volume of project members and a project manager that rents the cloud service platform. In the given example, a project manager is the original owner of the project file and takes charge of dividing the file into encrypted blocks, system parameters generation, member joining/revocation, and sharing the blocks in the project group through a cloud service provider. All project members can access, download, and modify the specified, encrypted data blocks.

Cloud service provider offers a wealth of storage services and powerful computing abilities by charging a certain fee. Referring to the research in [7], CSP can honestly implement the scheme but may try to gain the content of stored files and return an incorrect result to the verifier to get some extra benefits. Therefore, we assume that the CSP is semitrusted, encrypting all file blocks stored in the CSP, and generate tags corresponding to the project members.

The verifier can be any member of the project group that checks the integrity of encrypted data blocks kept in the CSP. Once a verifier sends an integrity auditing request, the CSP generates and returns the verification information. The verifier then checks the correctness of the auditing proof and reports the verification result.

4.2. Design Goals. To efficiently and securely verify shared encrypted data with a volume of members in a project group, our proposed scheme should be designed to achieve the following properties:

(i) Correctness: Based on the challenged proof generation, the verifier is able to correctly detect the integrity of challenging blocks.

(ii) Unforgeability: Only the specified member in the project group can yield valid verification information on the encrypted data blocks.

(iii) Identity privacy: During the integrity of auditing, the CSP cannot distinguish the identity of tag generator on each randomly picked block in the shared project group.

(iv) Tag-updating: When the identity of some members in the project group is revoked or new members are added, the corresponding ciphertext tags should be updated efficiently and securely.

(v) Verifiability: Random verifier is able to verify the integrity of ciphertext attached tags by the challenged proof calculated by the CSP.

4.3. Outline of the Scheme. The scheme consists of eight steps:

(1) $Setup(1^\kappa) \longrightarrow (params, msk)$: Taking a security parameter κ as input, the project manager implements this step and outputs the master key msk and all system parameters $params$.

(2) PartialKeyGen (ID_i, params, msk) $\longrightarrow$ (D_i): Taking the member's identity ID_i, the master key msk, and the parameters $params$ as input, the project manager executes this step and outputs member u_i's partial key D_i.

(3) KeyGen$(D_i, params) \longrightarrow (ssk_i, spk_i)$: Taking the member's partial key D_i and the parameters $params$ as input, the project member runs this step and returns pairing private/public key (ssk_i, spk_i).

(4) Encrypt$(m_i, spk_i) \longrightarrow (\sigma_i)$: Taking the file blocks m_i and member's public key spk_i as input, the project

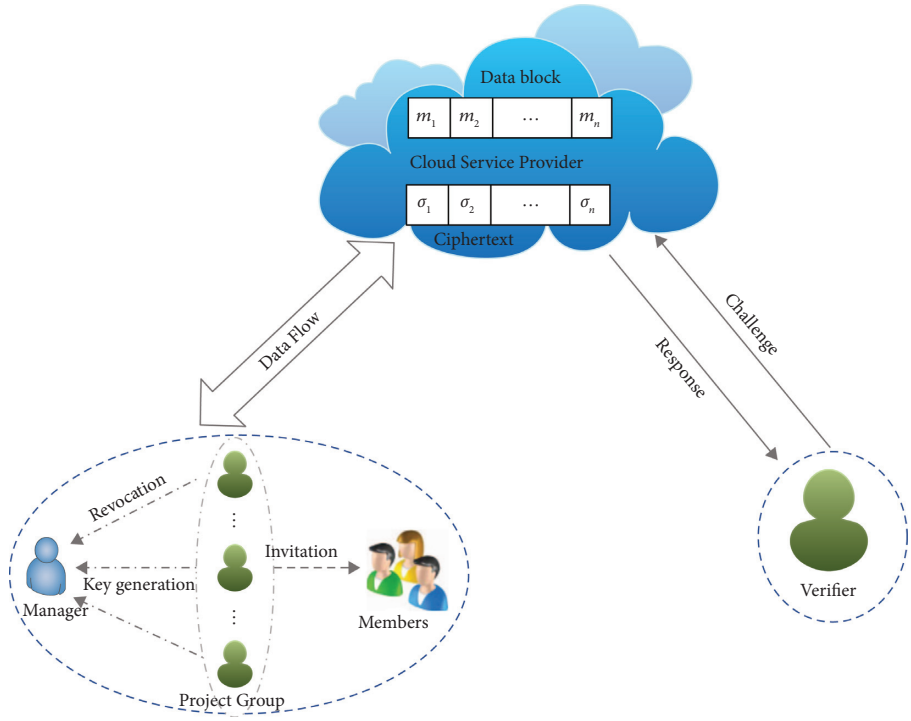


Figure 1: System model.

manager executes this step and generates a ciphertext σ_i.

(5) $TagGen(\sigma_i, ssk_i) \longrightarrow (T_i)$: Taking the ciphertext σ_i and a private ssk_i as input, the project member leverages this step to generate a tag T_i attached to the ciphertext block σ_i when uploading to CSP.

(6) Challenge $(c) \longrightarrow (chal, F_{i\,d})$: Taking the count of challenged block c as input, the verifier runs this step to output a challenge information *chal* appending the block name $F_{i\,d}$ for the integrity querying of the data file.

(7) ProofGen $(chal, F^*, \{T_i | i \in n\}) \longrightarrow P$: Taking the challenge information *chal*, the challenged encrypted block set F^*, and tag set $\{T_i | i \in n\}$ as input, CSP runs this step and responds with the integrity proof P.

(8) Verify Proof $(P, chal, params) \longrightarrow 0, 1$: Taking the integrity proof P, the challenge information *chal*, and the public parameters *params* as input, the verifier implements this step and returns 1 if result P passes the verification; otherwise, it returns 0.

Note that, in addition to the above steps, there are two other steps: *JoinGen* and*RevGen*. Step *JoinGen* is executed by the project members, which invites some other members who are not in the project group, and step *RevGen* is also implemented by the project members, and the procedure is divided into two scenarios depending on whether the revoked member has invited members to participate in the project group.

4.4. Secure Model. Since the certificateless cryptography [47] is the underlay of our new scheme, and referring to the security model of data integrity auditing protocols represented in papers [30, 39, 48], we consider the security requirement and adversary model of the encrypted shared scheme against a fully-adaptive chosen ciphertext attacker (IND-CCA) [47, 49] which involves a challenger $\mathscr{C}$ and four types of adversaries, namely ,$\mathscr{A}_1$, $\mathscr{A}_2$, $\mathscr{A}_3$, and$\mathscr{A}_4$. Among the four adversary types, although adversaries $\mathscr{A}_1$ and$\mathscr{A}_2$ both execute the tag-forge attacks, they have different attack capabilities. Type $\mathscr{A}_3$ implements the ciphertext integrity proof attack to cheat the verifier, and type $\mathscr{A}_4$ tries to generate a forgery of regenerated ciphertext and passes the verification of ciphertext receiver. We give the four following games to illustrate the security model in detail.

4.4.1. Setup. There are two parties, adversary $\mathscr{A}_i$ for $\mathscr{A}_i$ and challenger $\mathscr{C}$ that keeps the private keys and the master key security and sends the public system parameters to$\mathscr{A}_i$. Challenger $\mathscr{C}$ interacts with adversary $\mathscr{A}_j$ for $j \in \{1, 2\}$ in this game. In order to generate a forgery of tag in a security game, $\mathscr{A}_j$ needs to execute the following different queries: hash query, key query, public key query, public key replacement, encryption query, and tag query.

4.4.2. Common Queries. $\mathscr{A}_j$ gives the polynomial times different queries to $\mathscr{C}$ which responds to the following queries:

(1) Hash query is adaptively made by $\mathscr{A}_j$ and the hash values are responded by $\mathscr{C}$.

(2) Key query is adaptively run by $\mathscr{A}_j$ to submit different target identity ID (first running the step PartialKeyGen by$\mathscr{C}$ for ID if necessary) to $\mathscr{C}$ for

querying the key, and then a key pairing for ID is responded to $\mathcal{A}_j$ by $\mathcal{C}$ performing the step *KeyGen*.

(3) Encrypt query is adaptively implemented by $\mathcal{A}_j$ to submit different plaintext m to $\mathcal{C}$, and then a ciphertext σ with both the randomly picked value r and the public key spk_{ID} is responded to $\mathcal{A}_j$ by $\mathcal{C}$.

(4) Tag query is adaptively executed by $\mathcal{A}_j$ to query the tag of any ciphertext block σ with the corresponding member's identity ID, and then a tag Tag is returned by $\mathcal{C}$ running the step *TagGen*.

4.4.3. Game1. In this game, adversary $\mathcal{A}_1$ not only executes the Common Queries above but also runs the following specialized queries:

(1) Partial key query: is adaptively implemented by $\mathcal{A}_1$ to submit different target identity ID to $\mathcal{C}$, and then the partial key for the ID is responded to $\mathcal{A}_1$ by $\mathcal{C}$ running the step PartialKeyGen.

(2) Public key replacement: According to the assumed capability of $\mathcal{A}_1$, it can replace the public key $ssk_{I\,D}$ of any ID with random value $ssk_{ID}^{'}$ multiple times if necessary.

4.4.4. Forgery. Eventually, there are two scenarios on the forgery tag Tag' output by adversary $\mathcal{A}_1$. One is that $\mathcal{A}_1$ outputs a forgery tag Tag' for the ciphertext block σ encrypted by the public key$ssk_{I\,D}$, and the tag is generated with the public key $ssk_{ID'}$ and the identity ID'. The other one is that $\mathcal{A}_1$ outputs a forgery tag Tag' for ciphertext block σ' encrypted by public key $ssk_{ID'}$, and the tag is generated with the public key $ssk_{ID'}$ and the identity ID'. In both scenarios, such adversary $\mathcal{A}_1$ does not have access to the master key of system, but it can request the public key and has the capability to replace the member's public key and make the tag queries for all identities of its random choice. However, if $\mathcal{A}_1$ wants to win the game, there are several natural restrictions on adversary $\mathcal{A}_1$ as discussed below:

(1) $\mathcal{A}_1$ cannot request a query on the private key for identity ID' at any time.

(2) $\mathcal{A}_1$ cannot both query the partial key for ID' and substitute the public key of identity ID' at the same time.

(3) $\mathcal{A}_1$ cannot make a tag query for the encrypted target data block σ' with the identityID' and the public key $ssk_{ID'}$.

(4) In addition to the above limitations, $\mathcal{A}_1$ can forge a valid tag for the encrypted data block σ' with the identity ID' and the public key $ssk_{ID'}$, and it also can forge a valid tag for the ciphertext block σ encrypted with the legitimate public key $ssk_{I\,D}$, where the tag is generated by the replaced public key $ssk_{ID'}$ and the identity ID'.

4.4.5. Game2. In this game, adversary $\mathcal{A}_2$ only executes the Common Queries above and then forges the tag Tag' for the ciphertext σ' with the identity ID'.

4.4.6. Forgery. In this process of forgery, adversary $\mathcal{A}_2$ is unable to replace the member's public key, but it has the capability to access the master key of system. However, there are also two scenarios on the forgery tag Tag' output by adversary $\mathcal{A}_2$. One is that $\mathcal{A}_2$ outputs a forgery tag Tag' for ciphertext block σ encrypted by the public key $ssk_{I\,D}$, and the tag is generated with the public key $ssk_{ID'}$ and the identity ID'. The other one is that $\mathcal{A}_2$ outputs a forgery tag Tag' for ciphertext block σ' encrypted by public key $ssk_{ID'}$, and the tag is generated with the public key $ssk_{ID'}$ and the identity ID'. In addition, if $\mathcal{A}_2$ wants to win the game, it is subject to the following restrictions:

(1) $\mathcal{A}_2$ can neither query the private key nor replace the public key for ID' at any point.

(2) $\mathcal{A}_2$ cannot make a tag query for the encrypted target data block σ' with the identity ID'.

(3) In addition to the above limitations, $\mathcal{A}_2$ can forge a valid tag Tag' for the encrypted data block σ' with the identityID', as well as the legitimate public key $ssk_{I\,D}$.

Definition 5. The scheme is semantically secure against the single tag forged attack of the ciphertext block if adversary $\mathcal{A}_1$ or $\mathcal{A}_2$ in polynomial probability time has a negligible advantage to win *Game* 1 and *Game* 2.

4.4.7. Game3. In terms of Definition 5, an adversary cannot forge a legitimate label for a single ciphertext block without accessing the right private key. In this game, we consider that adversary $\mathcal{A}_3$ that acts as the untrusted CSP in the system attempts to persuade the verifier to pass the integrity verification of corrupted data. Inspired by [7], challenger $\mathcal{C}$ plays two roles, that is, the honest CSP and an integrity checker, and the operation of *Game* 3 is executed as follows:

Tag query: The target tuple (ID, m) is adaptively selected by $\mathcal{A}_3$ and sent to $\mathcal{C}$, which responds with the querying tag which is generated with the ciphertext σ and the identity ID by the step *TagGen*.

Challenge: Challenger $\mathcal{C}$, which acts as the verifier, generates and sends a random challenge information *chal* to $\mathcal{A}_3$, which is requested to respond with the corresponding data possession proof P for *chal*.

Forgery: Once receiving the challenge information *chal*, $\mathcal{A}_3$ acts as the CSP, generates a proof P, and responds to$\mathcal{C}$. The premise for $\mathcal{A}_3$ to win the game is that the miscalculated block information in proof P can pass the integrity verification successfully.

Definition 6. The scheme is semantically secure against forging the integrity proof on incorrect data if adversary $\mathcal{A}_3$ in polynomial probability time has a negligible advantage to win *Game* 3.

4.4.8. Game4. In this game, the specified member acts as adversary $\mathcal{A}_4$ that interacts with challenger $\mathcal{C}$. Here, the revoked member and CSP are regarded as the trusted parties.

If the reencrypted data has been corrupted, $\mathcal{A}_4$ tries to cheat the verifier that the tag generated by reencrypted data can pass the integrity verification. In terms of Definitions 5 and 6, we know that any adversary cannot pass the tag verification on a single block without accessing the private key and correct data. Therefore, the focus of this game is on whether adversary $\mathcal{A}_4$ can forge the integrity proof of reencrypted data to pass the verification. Inspired by [7, 29], challenger $\mathcal{C}$ plays two roles, that is, the honest CSP and a revoked member, and the operation of *Game* 4 is executed as follows:

Reencrypt key query: $\mathcal{A}_4$ adaptively picks an identity ID and submits it to challenger $\mathcal{C}$ for querying the reencrypting key of ID. $\mathcal{C}$ runs the reencrypting key subroutine in step *RevGen* and returns the reencrypting key $r_{\mathcal{A}\leftrightarrow\mathcal{C}}$.

Tag query: The target tuple (σ, ID) is adaptively selected by $\mathcal{A}_4$ and sent to $\mathcal{C}$ for querying the tag for the reencrypting ciphertext σ'. According to the step *RevGen*, $\mathcal{C}$ responds the tag generated by ciphertext σ' and ID to adversary$\mathcal{A}_4$.

4.4.9. Forgery. Eventually, $\mathcal{A}_4$ outputs a forgery tag Tag' for the target ciphertext σ with the identity ID.

Definition 7. The scheme is semantically secure against forging the integrity proof without both correct identity and reencryption key if adversary $\mathcal{A}_4$ in polynomial probability time has a negligible advantage to win the aforementioned *Game* 4.

5. Our Scheme

Without loss of generality, there is a project manager named u_0 in the group which is in charge of the generation of system parameters and other engineers' partial secret keys. Suppose that the project F is divided into n project blocks as the following $M = (m_1, m_2, \cdots, m_n)$. u_0 invites z engineers in one group to execute this project, and each engineer u_i has a unique identity represented as ID_i for $1 \le i \le z$. In order to keep the security of each block, all project blocks stored on the CSP should be encrypted by the public key of the corresponding developing engineer; namely, $C^* = (\sigma_1, \sigma_2, \cdots, \sigma_n)$, in which σ_i represents the ciphertext of ith subproject m_i. The scheme consists of the following steps:

Setup (1^κ): u_0 takes as input a security parameterκ and outputs the public parameters including two multiplicative cyclic groups $\mathbb{G}_1$ and $\mathbb{G}_2$ and a bilinear map e: $\mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$, where the orders of $\mathbb{G}_1$ and $\mathbb{G}_2$ are both the big prime q and g is a generator of $\mathbb{G}_1$. It sets two collision-resistant hash functions H_1: $\{0,1\}^* \longrightarrow \mathbb{G}_1^*$ andH_2: $\{0,1\}^* \longrightarrow \mathbb{G}_1^*$. Two pseudorandom generators π and ϕ are selected, where π: $Z_q^* \times \{1, 2, \cdots, n\} \longrightarrow Z_q^*$ and ϕ: $Z_q^* \times \{1, 2, \cdots, n\} \longrightarrow \{1, 2, \cdots, n\}$ are used to generate the pseudorandom value and pseudorandom permutation, respectively. u_0 selects a master key $msk = s \in Z_q^*$ and calculates the public key $P_0 = g^s$. All the parameters params $= (q, g, \mathbb{G}_1, \mathbb{G}_2, e, P_0, H_1, H_2, \phi, \pi)$ are published.

PartialKeyGen: When receiving the identity ID_i of the participant engineer u_i, the project manager u_0 returns $D_i = H_1(ID_i)^s$ by secure channel as the partial private key of u_i.

KeyGen: ID_i randomly selects secret value $x_i \in Z_q^*$ as a partial private key and constructs ID_i's private/public key pairs (ssk_i, spk_i) as $ssk_i = \langle D_i^{x_i}, x_i \rangle$ and $spk_i = \langle X_i, Y_i \rangle$, where $X_i = g^{x_i}$ and $Y_i = {P_0}^{x_i}$.

Encrypt: The project manager u_0 randomly picks a value $r_i \in Z_q^*$ and encrypts l project blocks (engineer u_i could be assigned l project blocks)$\{m_{i_1}, \cdots, m_{i_l}\}$ into$\{\sigma_{i_1}, \cdots, \sigma_{i_l}\}$ leveraging u_i's public key spk_i and uploads them to the CSP, where $m_{i_j} \in Z_q$, $\sigma_{i_j} = \langle \sigma_{i_j 1}, \sigma_{i_j 2} \rangle$ for$\sigma_{i_j 1} = m_{i_j} \oplus H_2(e(H_1(ID_i), Y_i)^{r_i}))$, $\sigma_{i_j 2} = g^{r_i})$, and $j \in (1, l)$. Note that u_0 only executes this step once, encrypting the corresponding project blocks by utilizing every engineer's public key, and later this step is mainly implemented by the engineers involved in the project.

TagGen: Since each block has a unique file name F_{id}, a tag will be generated for all the encrypting blocks of file F. Suppose that the project engineer u_i wants to generate tags for each uploaded encrypted block σ_i'. It randomly picks a parameter $r_i' \in Z_q^* (1 \le i \le z)$, first encrypts its developed project blocks ${}_{mi_k}{}'(1 \le k \le l)$ into ${}_{\sigma i_k}{}' = \langle {}_{\sigma i_k}1', {}_{\sigma i_k}2' \rangle$, and then generates tags $T_{i_k} = (H_2(\omega_{i_k}) \cdot {}_{\sigma i_k}2')^{x_i} \cdot {D_i}^{\sigma i_k 1'}$ for the ciphertexts, where$\omega_{i_k} = F_{id_k} \| n \| i_k$. Then, project engineer u_i uploads $\{{}_{\sigma i_k}{}', T_{i_k} | k \in (1, l)\}$ to the CSP. Then, the CSP utilizes the public parameters and the data provided by the user to construct an (1) to check the correctness of all u_i's tags:

$$e\left(\prod_{k=1}^{l} T_{i_k}, g\right) = e\left((\sigma_{i2}')^l \cdot \prod_{k=1}^{l}\left(H_2\left(\omega_{i_k}\right)\right), X_i\right) \cdot e\left((H_1(ID_i))^{\sum_{k=1}^{l} {}_{\sigma i_k}1'}, P_0\right), \quad (1)$$

where $\sigma_{i2}' = g^{x_i}$.

Challenge: Anyone as a verifier can check the integrity of group data stored in CSP. The verifier randomly picks the challenged block count $c (1 \le c \le n)$ and two values$k_1, k_2 \in Z_q^*$. Then the challenged information $chal = (c, k_1, k_2)$ appending the file name F_{id} is sent to CSP.

ProofGen: Once the CSP receives $chal = (c, k_1, k_2)$, the challenge information set $I = \{(v_i', a_i)\}$ is calculated, in which $a_i = \pi(k_1, i)$ is the regenerated parameter by the pseudorandom generatorπ, and the subset $\{v_i' | i \in (1, c)\}$ (for $v_i' = \phi(k_2, i)$) of $\{1, 2, \cdots, n\}$ is a new index permutation of challenge block regenerated by the pseudorandom generatorϕ. Without loss of generality, suppose that the challenge block set consists of encrypted blocks $\sigma_{v_1'}, \sigma_{v_2'}, \cdots, \sigma_{v_c'}$ and let$C$ denote $\{\sigma_{v_1'}, \sigma_{v_2'}, \cdots, \sigma_{v_c'}\}$. Let the challenge block subsets $C_{l_1} = \{\sigma_{v_1}, \cdots, \sigma_{v_j}\}$, $C_{l_2} = \{\sigma_{v_{j+1}}, \cdots, \sigma_{v_t}\}$ $\cdots$, $C_{l_{z'}} =$

$\{\sigma_{v_{u+1}}, \cdots, \sigma_{v_c}\}$ belong to engineers u_{l_1}, u_{l_2}, $\cdots$, $u_{l_{z'}}$, respectively, where the permutation $\{v_1, \cdots, v_j, \cdots, v_{u+1}, \cdots, v_c\}$ is the rearrangement of permutation $\{v_1', v_2', \cdots, v_c'\}$. We can obtain $C = C_{l_1} \cup C_{l_2} \cup \cdots \cup C_{l_{z'}}$ and $C_{l_k} \cap C_{l_{k'}} = \varnothing$ for $k \neq k'$, where the set $\{l_1, l_2, \cdots, l_{z'}\}$ is the subset of permutation$\{1, 2, \cdots, z\}$ and $|C_{l_1}| + |C_{l_2}| + \cdots + |C_{l_{z'}}| = c$. The CSP calculates two sets $T = \{\overline{T}_1, \cdots, \overline{T}_{z'}\}^z$ and $F = \{\overline{F}_1, \cdots, \overline{F}_{z'}\}$, where $\overline{T}_k = \prod_{v_j \in C_{l_k}} T_{v_j}{}^{a_j}$ and $\overline{F}_k = \sum_{v_j \in C_{l_k}} a_j \, \sigma v_j 1'$. Finally, the proof $P = (T, F)$ is sent to the verifier.

VerifyProof: Upon receiving proof P, the verifier utilizes the precalculated values set $\{(v_i', a_i)\}$ to generate a set of challenge blocks. According to the tag generation rules, the verifier obtains ω_{v_j} to generate all the tags $T_{v_i}{}^{a_i}$ of participating challenge blocks. Then it takes all above proof information as input to check whether (2) holds, where $l_k' = |C_{l_k}|$:

$$e\left(\prod_{k=1}^{z'} \overline{T}_k, g\right) = \prod_{k=1}^{z'} \left(e\left(\prod_{v_j \in C_{l_k}} \left(H_2\left(\omega_{v_j}\right) \cdot \left(\sigma v_j 2'\right)^{l_k'} \right), X_{l_k}{}^{a_j} \right) \cdot e\left(\prod_{k=1}^{z'} H_1\left(ID_{l_k}\right)^{\overline{F}_k}, P_0 \right) \right). \tag{2}$$

If this equation holds, it outputs either 1 ("*accept*") or 0 ("*reject*"). The correctness of this scheme can be checked by the following equality:

$$\begin{aligned} e\left(\prod_{k=1}^{z'} \overline{T}_k, g\right) = \prod_{k=1}^{z'} e(T_k, g) = \prod_{k=1}^{z'} e\left(\prod_{v_j \in C_k} T_{v_j}{}^{a_j}, g \right) &= \prod_{k=1}^{z'} e\left(\prod_{v_j \in C_{l_k}} \left(\left(H_2\left(\omega_{v_j}\right) \cdot {}_{\sigma v_j 2}{}' \right)^{x_{v_j}} \cdot D_{v_j}{}^{\sigma i_k 1'} \right)^{a_j}, g \right) \\ &= \prod_{k=1}^{z'} \left(e\left(\prod_{v_j \in C_{l_k}} \left(H_2\left(\omega_{v_j}\right)^{x_{v_j} a_j} \cdot {}_{\sigma v_j 2}{}'^{\, x_{v_j} a_j} \right), g \right) \right. \\ &\quad \cdot e\left(\prod_{v_j \in C_{l_k}} D_{v_j}{}^{\sigma i_k 1' a_j}, g \right)^{l_k' = (C_{l_k})} \prod_{k=1}^{z'} e\left(\prod_{v_j \in C_{l_k}} \left(H_2\left(\omega_{v_j}\right) \cdot {}_{\sigma v_j 2}{}'^{\, l_k'} \right), g^{x_{v_j} a_j} \right) \\ &\quad \left. \cdot e\left(H_1\left(ID_{v_j}\right)^{\sum \sigma i_k 1' a_j}, P_0 \right) \right) \\ &= \prod_{k=1}^{z'} \left(e\left(\prod_{v_j \in C_{l_k}} \left(H_2\left(\omega_{v_j}\right) \cdot \left(\sigma v_j 2'\right)^{l_k'} \right), X_{l_k}{}^{a_j} \right) \right) \cdot e\left(\prod_{k=1}^{z'} H_1\left(ID_{l_k}\right)^{\overline{F}_k}, P_0 \right). \end{aligned} \tag{3}$$

5.1. Invite to Join. If engineer u_i invites another engineer u_j to participate in its subproject, u_i first sends an identity concatenation $ID_j \| ID_i$ to u_0, and then u_0 responds a partial private key $D_{ji} = H_1(ID_j || ID_i)^s$ to u_j by secure channel. u_j randomly chooses $x_{ji} \in Z_q{}^*$ to generate its secure key $ssk_{ji} = \langle D_{ji}{}^{x_{ji}}, x_{ji} \rangle$ and public key $spk_{ji} = \langle X_{ji}, Y_{ji} \rangle$, where $X_{ji} = g^{x_{ji}}$ and $Y_{ji} = P_0{}^{x_{ji}}$.

While u_j successfully joins the project and wants to editu_i's some block (named ${}_{mi_k}{}'$), the specified file ciphertext block ${}_{\sigma i_k 1}{}'$ needs to converted to a block encrypted by u_j's public key. The reencryption key $r_{i \leftrightarrow j}$ is generated and the ciphertext ${}_{\sigma i_k 1}{}'$ turns into ${}_{\sigma j i_k 1}{}'$ which is encrypted by u_j's private key and parameter r_i randomly picked by u_0. Note that the reencryption key $r_{i \leftrightarrow j}$ is bidirectional; that is, it can be utilized to transfer the ciphertext from u_j to u_0 and vice versa.

JoinGen$(ID_i, ID_j) \longrightarrow (r_{i \leftrightarrow j})$: When receiving the identity ID_j, u_i calculates the reencryption key $r_{i \leftrightarrow j} = H_2(e(H_1(ID_i), Y_i)^{r_i} \oplus e(H_1(ID_j \| ID_i), Y_{ji})^{r_i})$ and sends it tou_j. Then u_j calculates ciphertext ${}_{\sigma j i_k 1}{}'$ for block ${}_{mi_k}{}'$ as ${}_{\sigma j i_k 1}{}' = {}_{\sigma i_k 1}{}' \oplus r_{i \leftrightarrow j} = {}_{mi_k}{}' \oplus H_2 \ (e(H_1(ID_i), Y_i)^{r_i}) \oplus H_2(e(H_1(ID_i), Y_i)^{r_i} \oplus e \ (H_1(ID_j \| ID_i), Y_{ji})^{r_i}) = {}_{mi_j}{}' \oplus H_2(e(H_1 \ (ID_i), Y_i)^{r_i}) \oplus (H_2(e(H_1(ID_i), \ Y_i))^{r_i}) \oplus H_2(e(H_1(ID_j \| ID_i), Y_{ji})^{r_i}) = {}_{mi_k}{}' \oplus H_2(e(H_1(ID_j \| ID_i), Y_{ji})^{r_i}) = {}_{mi_k}{}' \oplus H_2(e(D_{ji}{}^{x_{ji}}, g^{r_i}))$..

5.2. Revoke a Participant. Once an engineer u_i leaves the project, the project manager u_0 should claim the private/public key pairings of u_0 to be invalid. At the same time, the contents consisting of the ciphertext, tags, and so forth of the file block associated with the revoked engineer u_i are also changed. Otherwise, there are some secure risks on the ciphertext and tags which are executed by u_i; thereby the integrity of the ciphertext cannot be checked either. In this process, there are two situations to be considered: one is that the revoked user u_i has invited engineers in the project, and,

in this case, any inviter (named u_j) can be required to replace the tag and ciphertext of the revoked user. In the other case, u_i does not invite any users to participate in the project, so the project manager needs to convert the ciphertext and tags for u_i. We represent the detailed implementation as follows:

RevGen: Assume that u_i is the revoked project member, and u_j is the member who continues the project in place of u_i. In this section, CSP is used to check the correctness of regenerated tags by u_j. In addition, suppose that u_i, u_j, and CSP are all online simultaneously during this procedure.

Without loss of generality, let member u_j as a specified recipient take charge of all blocks of the revoked engineeru_i. u_i utilizes the step *JoinGen* to regenerate the ciphertext of u_j, and the block tag is yielded by u_j.

(1) u_i calculates $r_{i\hookrightarrow j}$ and sends it to u_j, where $r_{i\hookrightarrow j} = H_2(e(H_1(ID_i), Y_i)^{r_i} \oplus e(H_1(ID_j) \cdot Y_j)^{r_i}))$.

(2) Leveraging the key $r_{i\hookrightarrow j}$, u_j calculates ${}_{\sigma j_{k1}}{}' = r_{i\hookrightarrow j} \oplus {}_{\sigma i_{k1}}{}' = {}_{mi_k}{}' \oplus (H_2 e(H_1(ID_j), Y_j)^{r_i})$ and ${}_{\sigma j_{k2}}{}' = {}_{\sigma i_{k2}}{}' = g^{r_i}$ and publishes the regenerated ciphertext ${}_{\sigma j_k}{}' = \langle {}_{\sigma j_{k1}}{}', {}_{\sigma j_{k2}}{}' \rangle$ of block ${}_{mi_k}{}'$ for $1 \le k \le l$. Then, u_j calculates the tag $T_{j_k} = (H_2(\omega_{j_k}) \cdot {}_{\sigma j_{k2}}{}')^{x_j} \cdot D_j{}^{\sigma j_{k1}}$ to CSP for$\omega_{j_k} = F_{id_k} \| n \| i_k$.

(3) While receiving the tuple $({}_{\sigma j_k}{}', T_{j_k})$, CSP verifies (1) to ensure the validity of the tags.

6. Security Proof

In this section, we give the secure proof of the proposed scheme via the following properties.

6.1. Security Analysis

Theorem 1. *If a polynomial probability time adversary $\mathcal{A}_1$ has an advantagee to win Game* 1 *described in Section 4.4 within time t after executing the most q_{H_1} Hash-1 queries, q_K key queries, q_R Public Key Replace, q_E encryption queries, and q_{H_2} Hash-2 queries and requesting at most q_T times tag queries, then there exists a (ε', t)-simulator $\mathcal{B}$ that can address the CDH problem with $(\varepsilon' \ge \varepsilon/((1 + q_p + q_T) \cdot e))$ $t' \le t + (q_{H_1} + q_p + 3q_K + 3q_{H_2} + 2q_T t_e + 3t_m q_T + q_R + q_E)$, where one exponentiation costs time t_e on$\mathbb{G}_1$, one scalar multiplication operation costs time t_m in$\mathbb{G}_1$, and e is the base of natural logarithm.*

Proof. On input (g, g^a, g^b) in $\mathbb{G}_1$, if adversary $\mathcal{A}_1$ is able to forge a tag with the identity$I\ D$ and the replaced public key in *Game* 1, then algorithm $\mathcal{B}$ has capability to address CDH problem; that is, it can calculateg^{ab}. Given g, g^a, and g^b, simulator $\mathcal{B}$ simulates each step of interaction with $\mathcal{A}$ as follows:

Setup. $\mathcal{A}_1$ launches a query-respond game. $\mathcal{B}$ sets $P_0 = g^a$ with the master key a which is security picked and then outputs and returns the system parameters $params = (q, g, \mathbb{G}_1, \mathbb{G}_2, e, P_0, H_1, H_2, \phi, \pi)$ to $\mathcal{A}_1$.

Hash-1 Query. $\mathcal{A}_1$ adaptively requests the Hash-1 query results for any identity ID^* in terms of its capability. In order to facilitate the management of all the query results, $\mathcal{B}$ establishes a tuple list $L_1 = \{(I\ D, r, Q, \tau)\}$ to record all query data. If a certain ID^* has been recorded in the list, $\mathcal{B}$ directly returns its corresponding tuple (ID^*, r^*, Q^*, τ^*) to $\mathcal{A}_1$. Otherwise, $\mathcal{B}$ selects a random value $r^* \in Z_q{}^*$ and tosses a coin $\tau \in \{0, 1\}$. Assume that the coin represents 1 with a probability of γ, and vice versa, $1 - \gamma$. If τ shows 0, $\mathcal{B}$ sets $Q^* = H_1(ID^*) = g^{r^*} \in \mathbb{G}_1$; if τ shows 1, $\mathcal{B}$ sets$Q^* = H_1(ID^*) = (g^b)^{r^*} \in \mathbb{G}_1$. Then the result Q^* is returned to $\mathcal{A}_1$ and the tuple $(ID^*, Q^*, *, *)$ is inserted to list L_1, where the symbol $*$ indicates that the position is empty and has no value, which may be generated in a subsequent query.

Partial key query. In order to obtain the partial key of any identity ID^*, $\mathcal{A}_1$ adaptively implements partial key query. $\mathcal{B}$ firstly checks whetherID^* corresponding tuple (ID^*, r^*, Q^*, τ^*) exists in L_1. If not, $\mathcal{B}$ executes the Hash-1 query and inserts the result in L_1. Notably, another new tuple list L_2 is established by $\mathcal{B}$ to manage the newly queried data during this process, where $L_2 = \{(ID, D_{ID}, spk_{I\ D}, ssk_{I\ D}, \sigma_{I\ D})\}$. If τ shows 1 in L_1, $\mathcal{B}$ returns$\perp$ for ID^* and then records the tuple value $(ID^*, \perp, *, *, *)$ in L_2. Otherwise, $\mathcal{B}$ responds the partial key query as follows.

(1) If ID^* is stored in listL_2, $\mathcal{B}$ checks whether the location of D_{ID^*} is a symbol$\perp$ or not. If it is not $\perp$, $\mathcal{B}$ returns it directly to $\mathcal{A}$. Otherwise, $\mathcal{B}$ reexecutes the coin tossing step in Hash-1 query. When the coin tosses $\tau = 0$, $\mathcal{B}$ returns the value $D_{ID^*} = (Q^*)^a = g^{ar^*}$ to $\mathcal{A}_1$ and then updates the values Q^*, τ in L_1, andD_{ID^*} in L_2 on the corresponding identity ID^*, respectively; otherwise, $\tau = 1$, and $\mathcal{B}$ aborts.

(2) If ID^* is not stored in list L_2, $\mathcal{B}$ determines the value of D_{ID^*} according to τ in list L_1. If $\tau = 0$, $\mathcal{B}$ returns $D_{ID^*} = g^{ar^*}$ to $\mathcal{A}_1$; otherwise, $\tau = 1$, and $\mathcal{B}$ aborts.

Note that the tuple in L_1 and L_2 has such a characteristic: the value of τ in the tuple $(I\ D, Q, r, \tau)$ of L_1 corresponding to the tuple $(I\ D, \perp, *, *, *)$ of L_2 is 1; the value of τ in the tuple $(I\ D, Q, r, \tau)$ of L_1 corresponding to the tuple $(I\ D, D_{ID^*}, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*})$ inL_2 is 0.

Key query. $\mathcal{A}_1$ adaptively requests the key query for any identityID^*. $\mathcal{B}$ searches list L_2 for the tuple $(I\ D, D_{ID^*}, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*})$.

(1) If the tuple $(I\ D, D_{ID^*}, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*})$ is stored in L_2, $\mathcal{B}$ first checks whether the position of D_{ID^*} is $\perp$ or not. If$D_{ID^*} = \perp$, $\mathcal{B}$ turns to execute Hash-1 query and partial key query in turn. Otherwise, $\mathcal{B}$ checks whether the position of spk_{ID^*} in this tuple is the symbol $*$. If $spk_{ID^*} = *$, $\mathcal{B}$ selects $x_{ID^*} \in Z_q{}^*$ at random and sets$ssk_{ID^*} = \langle D_{ID^*}{}^{x_{ID^*}}, x_{ID^*} \rangle$ and $spk_{ID^*} = \langle X_{ID^*}, Y_{ID^*} \rangle = \langle g^{x_{ID^*}}, P_0{}^{x_{ID^*}} \rangle$. $\mathcal{B}$ updates the tuple $(I\ D, D_{ID^*}, *, *, *)$ into L_2 and sends the key pairing (spk_{ID^*}, ssk_{ID^*}) to $\mathcal{A}_1$.

(2) If the tuple $(I\ D, D_{ID^*}, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*})$ is not stored in L_2, $\mathcal{B}$ turns to execute Hash-1 query and partial key query in turn. Once the value ofD_{ID^*} has been obtained after the Hash-1 query and partial key query, $\mathcal{B}$ randomly selects$x_{ID^*} \in Z_q{}^*$ and sets

$ssk_{ID^*} = \langle D_{ID^*}{}^{x_{ID^*}}, x_{ID^*} \rangle$ and $spk_{ID^*} = \langle X_{ID^*}, Y_{ID^*} \rangle = \langle g^{x_{ID^*}}, P_0{}^{x_{ID^*}} \rangle$. $\mathcal{B}$ inserts the tuple $(ID, D_{ID^*}, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*})$ into L_2 and returns (spk_{ID^*}, ssk_{ID^*}) to $\mathcal{A}_1$. On the other hand, if $D_{ID^*} = \perp$, the tuple $(ID, \perp, *, *, *)$ is inserted intoL_2, and $\mathcal{B}$ aborts.

Public Key Replace. According to the assumption, adversary $\mathcal{A}_1$ has capability to replace the public key. $\mathcal{A}_1$ adaptively implements the Public Key Replace for the target memberID^* with the substitution public key $sp_{kID^*}{}'$.

(1) If the tuple $(ID^*, D_{ID^*}, spk_{ID^*}, ssk_{ID^*}, *)$ is stored inL_2, $\mathcal{B}$ modifies the tuple as $(ID^*, D_{ID^*}, sp_{kID^*}{}', *, *)$ in terms of $\mathcal{A}_1$'s request, where $sp_{kID^*}{}' = (_{XID^*}{}', _{YID^*}{}')$.

(2) If the tuple $(ID^*, D_{ID^*}, spk_{ID^*}, ssk_{ID^*}, *)$ is not contained inL_2, $\mathcal{B}$ adds the tuple $(ID^*, *, sp_{kID^*}{}', *, *)$ to L_2 in terms of$\mathcal{A}_1$'s request, where $sp_{kID^*}{}' = (_{XID^*}{}', _{YID^*}{}')$.

Hash-2 query. In order to facilitate the management of Hash-2 query, a list $L_3 = \{(V_{ID}, \sigma_{ID_2}, \omega_{ID}, \lambda, h)\}$ is still established by $\mathcal{B}$ to record the participating tuple. As required, $\mathcal{A}_1$ runs the Hash-2 query on the identity ID^*, partial public key Y_{ID^*}, and Hash-1 query Q^*. $\mathcal{B}$ randomly picks a tuple value (λ^*, h^*) and calculates $V_{ID^*} = H_2(e(Q^*, _{YID^*}{}')^{\lambda^*})$, $\sigma_{ID_2{}^*} = g^{\lambda^*}$, and $H_2(\omega_{ID^*}) = g^{h^*}$ and then sends $V_{ID^*}, \sigma_{ID_2{}^*}$ and $H_2(\omega_{ID^*})$ to $\mathcal{A}_1$. The new tuple $(V_{ID^*}, \sigma_{ID_2{}^*}, \omega_{ID^*}, \lambda^*, h^*)$ is added in L_3.

Encrypt query. For any plaintext m, $\mathcal{A}_1$ adaptively requests the encrypt query with identityID^*. $\mathcal{B}$ searches list L_2 for the tuple$(ID^*, D_{ID^*}, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*})$. If the tuple $(ID^*, D_{ID^*}, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*})$ is stored in L_2, $\mathcal{B}$ returns σ_{ID^*} directly to$\mathcal{A}_1$. Otherwise, whether the tuple is $(ID^*, \perp, *, *, *)$ or $(ID^*, *, sp_{kID^*}{}', *, *)$, $\mathcal{B}$ calculates the ciphertext $_{\sigma ID^*}{}'$ with the replaced key $sp_{kID^*}{}'$, in which $_{\sigma ID^*}{}' = (_{\sigma ID_1{}^*}{}', _{\sigma ID_2{}^*}{}')$ and$_{\sigma ID_1{}^*}{}' = m_{ID^*} \oplus V_{ID^*}$. Then, $\mathcal{B}$ updates the tuple $(ID, \perp, sp_{kID^*}{}', *, _{\sigma ID^*}{}')$ and $(ID^*, *, sp_{kID^*}{}', *, _{\sigma ID^*}{}')$ in L_2, respectively.

Tag query. $\mathcal{A}_1$ adaptively requests the tag on any identityID^* and plaintext block m_{ID^*} by submitting the result ofσ_{ID^*}. Based on the result of tossing the coin in L_1; if $\tau^* = 1$, $\mathcal{B}$ aborts. Otherwise, based on the values of $H_2(\omega_{ID^*})$ in L_3 and both D_{ID^*} andσ_{ID^*} inL_2, $\mathcal{B}$ generates the tag for the tuple $(D_{ID^*}, \sigma_{ID^*}, H_2(\omega_{ID^*}))$ by step $TagGen$ and returns it to $\mathcal{A}_1$.

Forgery. Eventually, a forgery tag T', which is relevant to plaintext m' on the identityID' with the public key $spk_{ID'}$, is forged by $\mathcal{A}_1$. If $\tau = 0$, $\mathcal{B}$ aborts. Otherwise, based on the aforementioned operations, $\mathcal{B}$ holds the following values: $H_1(ID') = (g^b)^{r'}$, $\sigma_{ID_2'} = g^{\lambda'}$, $P_0 = g^a$, and $H_2(\omega_{ID'}) = g^{h'}$, $\sigma_{ID_1'}$, and$X_{ID'}$, and then it can output $g^{ab} = (T'/X_{ID'}{}^{\lambda+h})^{(1/(r'\sigma_{ID'}))}$ solving the proposed CDH problem.

Analysis. Now, we analyze the probability that $\mathcal{B}$ can guess the correct query of the target data block by simulating operation. Similar to the analysis and proof of [48, 50], $\mathcal{B}$ only halts two queries on partial key query and tag query; therefore the probability of $\mathcal{B}$ implementing the queries is higher than $(1-\gamma)^{q_p+q_T}$. Assume that the probability of occurrence of output of the right value of g^{ab} for $\mathcal{B}$ is $\varepsilon \cdot \gamma \cdot (1-\gamma)^{q_p+q_T}$.

Let

$$\varepsilon' = \varepsilon \cdot \gamma \cdot (1-\gamma). \tag{4}$$

In order to find the minimum value of ε', let us take the derivatives of both sides of (4) with respect to γ:

$$\begin{aligned} \frac{d\varepsilon'}{d\gamma} &= \frac{d\left(\varepsilon \cdot \gamma \cdot (1-\gamma)^{q_p+q_T}\right)}{d\gamma} \\ &= \varepsilon \cdot (1-\gamma)^{q_p+q_T} - \varepsilon \cdot \gamma \cdot \left(q_p+q_T\right) \cdot (1-\gamma)^{q_p+q_T-1} \\ &= \varepsilon \cdot (1-\gamma)^{q_p+q_T-1} \cdot \left[1-\gamma \cdot \left(q_p+q_T+1\right)\right]. \end{aligned} \tag{5}$$

Replace $d\varepsilon'/d\gamma = 0$; that is,

$$\varepsilon \cdot (1-\gamma)^{q_p+q_T-1} \cdot \left[1-\gamma \cdot \left(q_p+q_T+1\right)\right]. \tag{6}$$

We can obtain $\gamma_{opt} = 1/(1+q_p+q_T)$. Thereby (4) becomes

$$\varepsilon' = \varepsilon \cdot \gamma \cdot (1-\gamma)^{q_p+q_T} \geq \varepsilon \cdot \frac{1}{1+q_p+q_T} \cdot \left(1-\frac{1}{1+q_p+q_T}\right)^{q_p+q_T}. \tag{7}$$

According to the formula $\lim_{n\longrightarrow\infty}(1+(1/n))^n = e$ for $n \in N$, equation (7) becomes

$$\varepsilon' \geq \varepsilon/\left(\left(1+q_p+q_T\right) \cdot e\right). \tag{8}$$

Further, simulator $\mathcal{B}$ can solve the CDH problem in polynomial time t' which satisfies $t' \leq t + (q_{H_1} + q_p + 3q_K + 3q_{H_2})t_e + (3t_m + 2t_e) \cdot q_T + q_R + q_E$. □

Theorem 2. *If a PPT adversary $\mathcal{A}_2$ has an advantageε to win Game 2 described in Section 4.4 within time t after implementing the mostq_{H_1} Hash-1 queries, q_K key queries, q_E encryption queries, and q_{H_2} Hash-2 queries and requesting at most q_T times tag queries, then there exists a (ε', t)-simulator $\mathcal{B}$ that can address the CDH problem with$\varepsilon' \geq (\varepsilon/((1+q_K+q_T+q_E) \cdot e))$, $t' \leq t + (q_{H_1} + q_p + 3q_K + 3q_{H_2}) \cdot t_e + (3t_m + 2t_e) \cdot q_T + q_R + q_E$, where one exponentiation costs time t_e on$\mathbb{G}_1$, one scalar multiplication operation costs time t_m in $\mathbb{G}_1$, and e is base of natural logarithm.*

Proof. On input (g, g^a, g^b) in $\mathbb{G}_1$, the CDH algorithm $\mathcal{B}$ has capability to simulate a data-integrity-verifying security game and output g^{ab} by interacting with adversary$\mathcal{A}_2$ as follows:

Setup. $\mathcal{B}$ chooses the master key$s$ at random and outputs the system parameters$params$. Then, both s and$params$ are returned to $\mathcal{A}_2$ by $\mathcal{B}$.

Hash-1 query. $\mathcal{A}_2$ requests the Hash-1 query results for any identityID^* in terms of its capability. A tuple list $L_1 = \{(ID, Q, r)\}$ is established to record all query data by $\mathcal{B}$. If a certain ID^* has been stored in L_1, $\mathcal{B}$ returns $(g^a)^{r^*}$ to $\mathcal{A}_2$.

Otherwise, $\mathcal{B}$ selects a random value $r^* \in Z_q^*$ and responds to $\mathcal{A}_2$ with $Q^* = (g^a)^{r^*}$ and then stores (ID^*, Q^*, r^*) in L_1.

Key query. According to the assumption that adversary $\mathcal{A}_2$ has an ability to access the master keys, it directly initiates the key query of the public/private key pairing (spk_{ID}, ssk_{ID}). A list $L_2 = \{(ID, spk_{ID}, ssk_{ID}, \sigma_{ID}, \tau)\}$ is established by $\mathcal{B}$ for recording the results of key query.

(1) If ID^* is not stored in list L_2, $\mathcal{B}$ picks a value of x^* at random and tosses a coin $\tau \in \{0, 1\}$. Let γ denote the probability of $\tau^* = 0$; thus $1 - \gamma$ represents the probability of $\tau^* = 1$. If $\tau^* = 1$, $\mathcal{B}$ sets $ssk_{ID^*} = \langle (Q^*)^{x^*}, x^* \rangle$ and $spk_{ID^*} = \langle X_{ID^*}, Y_{ID^*} \rangle = \langle (g^b)^{x^*}, g^{sx^*} \rangle$ and inserts $(ID^*, spk_{ID^*}, ssk_{ID^*}, *, \tau^*)$ into L_2 but halts and returns $\perp$. If $\tau^* = 0$, $\mathcal{B}$ sets $ssk_{ID^*} = \langle (Q^*)^{x^*}, x^* \rangle$ and $spk_{ID^*} = \langle X_{ID^*}, Y_{ID^*} \rangle = \langle g^{x^*}, g^{sx^*} \rangle$ and records $(ID^*, spk_{ID^*}, ssk_{ID^*}, *, \tau^*)$ into L_2 and then returns x^* to $\mathcal{A}_1$.

(2) If ID^* is stored in list L_2, $\mathcal{B}$ checks whether the value of τ^* is 1 or 0. If $\tau^* = 1$, $\mathcal{B}$ halts and returns $\perp$. Otherwise, assuming that ssk_{ID^*} is already in L_2, $\mathcal{B}$ directly returns it to $\mathcal{A}_2$.

Notably, since $\mathcal{A}_2$ can access the master key to get the private key, there is no partial key query.

Hash-2 query. As required, $\mathcal{A}_2$ runs the Hash-2 query for the target value ω_{ID^*}. In order to record the participating tuple, $\mathcal{B}$ establishes a list $L_3 = \{(V_{ID}, \sigma_{ID_2}, \omega_{ID}, \lambda, h)\}$ for Hash-2 query. If ω_{ID^*} is stored in L_3, $\mathcal{B}$ returns the value $H_2(\omega_{ID^*}) = g^{h^*}$ to $\mathcal{A}_2$. Otherwise, $\mathcal{B}$ randomly picks a tuple value (λ^*, h^*) and calculates $V_{ID^*} = H_2(e(Q^*, Y_{ID^*})^{\lambda^*})$, $\sigma_{ID_2^*} = g^{\lambda^*}$, and $H_2(\omega_{ID^*}) = g^{h^*}$ and then returns $H_2(\omega_{ID^*})$ to $\mathcal{A}_2$. The new tuple $(V_{ID^*}, \sigma_{ID_2^*}, \omega_{ID^*}, \lambda^*, h^*)$ is added in L_3.

Encrypt query. For any plaintext m, $\mathcal{A}_1$ adaptively requests the encrypt query with identity ID^*. $\mathcal{B}$ searches list L_2 for the tuple $(ID^*, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*}, \tau^*)$.

(1) If the tuple $(ID^*, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*}, \tau^*)$ cannot be found in list L_2, $\mathcal{B}$ first requests the Hash-1 query and key query until the tuple (ID^*, Q^*, r^*) and the tossing coin value τ^* become existent in L_1 and L_2, respectively. Then $\mathcal{B}$ calculates the ciphertext $\sigma_{ID_1^*} = m_{ID^*} \oplus V_{ID^*}$ and updates the tuple $(ID^*, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*}, \tau^*)$ in L_2, where $\sigma_{ID^*} = (\sigma_{ID_1^*}, \sigma_{ID_2^*})$. If $\tau^* = 1$, $\mathcal{B}$ halts and returns $\perp$; otherwise, $\mathcal{B}$ outputs σ_{ID^*} to $\mathcal{A}_1$.

(2) If the tuple $(ID^*, spk_{ID^*}, ssk_{ID^*}, \sigma_{ID^*}, \tau^*)$ is stored in L_2 and $\tau^* = 0$, then $\mathcal{B}$ directly returns σ_{ID^*} to $\mathcal{A}_1$; otherwise, $\mathcal{B}$ halts and returns $\perp$.

Tag query. For ciphertext σ_{ID^*} associated with plaintext m^*, adversary $\mathcal{A}_1$ adaptively performs the tag query with $(\omega_{ID^*}, \sigma_{ID^*}, ID^*)$. $\mathcal{B}$ first checks the value of τ^* in L_2; if $\tau^* = 1$, $\mathcal{B}$ halts and outputs $\perp$. Otherwise, based on the values of ω_{ID^*} in L_3 and σ_{ID^*} in L_2, $\mathcal{B}$ calculates D_{ID^*} to generate the tag for the tuple $(\omega_{ID^*}, \sigma_{ID^*}, ID^*)$ by step $TagGen$ and then returns it to $\mathcal{A}_2$.

Forgery. Eventually, a forgery tag T', which is relevant to plaintext m' on identity ID' with the private key $ssk_{ID'}$, is forged by $\mathcal{A}_2$. If $\tau = 0$, $\mathcal{B}$ halts and outputs $\perp$. Otherwise, based on the aforementioned operations, $\mathcal{B}$ holds the following values: $P_0 = g^s$, $H_1(ID') = g^{ar'}$, $H_2(\omega_{ID'}) = g^{h'}$, $X_{ID'} = g^{bx'}$, and $\sigma_{ID_1'} = g^{\lambda'}$, $\sigma_{ID_1'}$, and then it can output $g^{ab} = (T')^{1/((\lambda' + h')x'r'ssk_{ID_1'})}$ solving the proposed CDH problem.

Analysis. In this game, there are three times of aborting for $\mathcal{B}$ on key query, encrypt query, and tag query. Thereby, the probability of $\mathcal{B}$ implementing the queries for $\mathcal{A}_2$ without abortion is higher than $(1 - \gamma)^{q_K + q_T + q_E}$. Thus, the probability of occurrence of output of the right value of g^{ab} for $\mathcal{B}$ is $\varepsilon' \geq \varepsilon \cdot \gamma \cdot (1 - \gamma)^{q_K + q_T + q_E} \geq \varepsilon / ((1 + q_K + q_T + q_E) \cdot e)$. Running time of algorithm $\mathcal{B}$ generating the forgery tag is $t' \leq t + (2q_T + q_K)t_e + (2q_T + 2q_{H_1} + 4q_K + 3q_{H_2}) \cdot t_e + q_E$. □

Theorem 3. *As long as the DL assumption holds, the probability that adversary $\mathcal{A}_3$ wins Game3, that is, to forge the tag and pass the verification in the scheme, is computationally negligible.*

Proof. If $\mathcal{A}_3$ wants to win the game, it has to generate the forged integrity proof $P' = (T', F')$ according to the challenge information $chal = (c, k_1, k_2)$ and satisfy the following equations with the nonnegligible probability:

$$e\left(\prod_{k=1}^{z'} \overline{T}'_k, g\right) = \prod_{k=1}^{z'} \left(e\left(\prod_{v_j \in C_{l_k}} \left(\left(H_2\left(\omega_{v_j}\right) \right) \cdot \left(\sigma_{v_j 2}' \right)^{l_k'}, X_{l_k}{}^{a_j} \right) \right) \cdot e\left(\prod_{k=1}^{z'} H_1\left(ID_{l_k}\right)^{\overline{F}'_k}, P_0 \right), \right. \tag{9}$$

where z' denotes the count of the group member participating in the challenge and l_k' represents the number of encrypted data blocks participating in the challenge.

On the other hand, assuming that $P = (T, F)$ is also a set of legitimate integrity proofs generated according to challenge information $chal = (c, k_1, k_2)$, tuple P is also verified

using the above equation; that is, $e(\prod_{k=1}^{z'} \overline{T}_k, g) = \prod_{k=1}^{z'} (e(\prod_{v_j \in C_{l_k}} (H_2(\omega_{v_j}))^{l'_k}, X_{l_k}^{a_j})) \cdot e(\prod_{k=1}^{z'} H_1(ID_{l_k})^{\overline{F}_k}, P_0)$. According to the rules of Game3, the two different integrity proofs P' and P generated, respectively, by adversary $\mathcal{A}_3$ and the legitimate member on the same challenge information $chal = (c, k_1, k_2)$ have the following relationship: $\overline{T} = \overline{T}'$ and $\overline{F} \neq \overline{F}'$. According to the above inequality, we can get the same formula as that in [29]: $\prod_{k=1}^{z'} H_1(ID_k)^{\overline{F}'_k} \neq \prod_{k=1}^{z'} H_1(ID_k)^{\overline{F}_k}$. Then we can get $\prod_{k=1}^{z'} H_1(ID_k)^{(\overline{F}'_k - \overline{F}_k)} = 1$.

Randomly given $\alpha_k \in Z_q^*$ and h a generator of $\mathbb{G}_1$, $H_1(ID_k)$ can be denoted as $H_1(ID_k) = h^{\alpha_k}$. Then we can get an approach to solve the DL problem by turning above formula into $1 = h^{\sum_{k=1}^{z'} \alpha_k \Delta \overline{F}_k}$; that is, $\sum_{k=1}^{z'} \alpha_k (\overline{F}'_k - \overline{F}_k) = 0$. In terms of the assumption in the game, there must be at least one tuple $(\overline{F}'_k, \overline{F}_k)$ that satisfies $\overline{F}'_k \neq \overline{F}_k$, and therefore at least one of the corresponding α_k is 0. Based on the analysis of α_k, there is at least one component $\alpha_k = 0 (1 \leq k \leq z')$ in the vector $(\alpha_1, \alpha_2, \cdots, \alpha_{z'})$, so the count of vectors satisfying the condition is at most $q^{z'-1}$. Clearly, we can find that the probability of $\sum_{k=1}^{z'} \alpha_k (\overline{F}'_k - \overline{F}_k) = 0$ is less than $q^{z'-1}/q^z = 1/q$, which is negligible for a large prime q. We can find that the probability of solving the DL problem is a nonnegligible probability $1 - 1/q$; thereby adversary $\mathcal{A}_3$ wins Game3 at the negligible probability. □

Theorem 4. *The adversary cannot pass the integrity proof by leveraging forged ciphertext.*

Proof: . If $\mathcal{A}_4$ wants to win the game, it tries to generate the forged ciphertext $\sigma' = (\sigma_{ID'_1}, \sigma_{ID'_2})$ with the forgery identity ID' and legitimate ciphertext σ in the revoke a participant phase. Suppose that adversary $\mathcal{A}_4$ generates its parameters for its identity ID' through the aforementioned games, for example, $H_1(ID')$, the private key $ssk_{ID'} = \langle D_{ID'}^x, x' \rangle$, the public key $spk_{ID'} = \langle X_{ID'}, Y_{ID'} \rangle$, and $H_2(\omega_{ID'})$. If the tag generating by $\mathcal{A}_4$ utilizing these parameters still passes the integrity proof by CSP, then adversary $\mathcal{A}_4$ wins this game. Otherwise, it fails.

(1) Assume that the revoked member u_i calculates the reencryption key $r_{i \leftrightarrow j} = H_2(e(H_1(ID_i), Y_i)^{r_i} \oplus e(H_1(ID_j), Y_j)^{r_i})$ with $\mathcal{A}_4$'s identity ID' and its public key $spk_{ID'}$ and then returns $r_{i \leftrightarrow j}$ to $\mathcal{A}_4$.

(2) $\mathcal{A}_4$ calculates reencrypted ciphertext $\sigma_{ID'} = (\sigma_{ID'_1}, \sigma_{ID'_2})$, where $\sigma_{ID'_1} = \sigma_{i_1} \oplus r_{i \leftrightarrow j} = m_i \oplus H_2(e(H_1(ID'), Y_{ID'})^{r_i})$ and $\sigma_{ID'_2} = \sigma_{i_2} = g^{r_i}$. Then $\mathcal{A}_4$ randomly picks $x_{ID'}$ as the partial key and outputs the forgery tag $T_{ID'} = (H_2(\omega_{ID'}) \cdot \sigma_{ID'_2})^{x'} \cdot D_{ID'}^{\sigma_{ID'_1}}$.

Through the aforementioned operations, the forgery tag is generated by attacker $\mathcal{A}_4$. If the tag passes the integrity proof, the equation $e(T_{ID'}, g) = e(H_2(\omega_{ID'}) \cdot \sigma_{ID'2}, X_{ID'}) \cdot e(H_1(ID')^{\sigma_{ID'1}}, P_0)$ holds. However, Theorems 1 and 2 have pointed that the tag with the forged private/public key pairing has a negligible probability to win Game1 and Game2; thereby, without the real ciphertext reencrypted by the legitimate private key, the adversary could output the correct integrity proof only with negligible probability. □

7. Performance Analysis

In this section, we first show the computation and communication cost of our scheme by theory and then represent the experiment results of the scheme.

7.1. Computation Cost. In our scheme, the computation cost is mainly concentrated on those operations that are computationally complex and time-consuming, such as pairing operation, exponentiation operation, and multiplication operations. For the simplicity of presentation, we use symbols C_p, C_{exp}, C_{mul_1}, and C_{mul_2} to represent the cost of one pairing operation in $\mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$, one exponentiation operation in $\mathbb{G}_1$, one multiplication in group $\mathbb{G}_1$, and one multiplication in group $\mathbb{G}_2$, respectively. In addition, the computational overhead of some other operations (such as pseudorandom parameter selection, hash operation, addition, and pseudorandom permutation) is negligible, so these operations are not analyzed. Continue to leverage the above symbols, and let n, z, c, and z' denote the number of projects divided into subprojects, members participating in the project, the challenge subprojects, and the members involved in the challenge, respectively. According to the processes of *Encrypt* and *TagGen*, the computation costs for n data blocks are $z(C_p + 2C_{exp})$ and $n(2C_{mul_1} + 2C_{exp})$, respectively.

In the process of *ProofGen*, we ignore the generation operation cost of challenge information *chal* and focus on the proof information, so the computation cost of this procedure is $cC_{exp} + (c - z')C_{mul_1}$. In this scheme, we also give the computation cost on the revocation step *RevGen* for a member of the project, in which the revoked member costs $2(C_p + C_{exp})$, the specified recipient costs $2(C_{exp} + C_{mul_1})$, and the verification computation cost for the CSP is $(2l - 1)C_{mul_1} + 3C_p + 2C_{exp} + C_{mul_2}$. Table 1 shows the detailed comparison of computation cost and data blocks types among the scheme of papers [7, 39] and ours. As can be seen from Table 1, in step *TagGen*, our scheme is one more C_{mul_1} than [7] and much less than [39]. After all, the cost of [39] is related to the number of participants z'. The cost amount of the step *ProofGen* is the same as that of [7] but is less than that of paper [39]. In addition, the step *VerifyProof* is used to verify the correctness of proof information and its computation cost is $(z' + 2)C_p + (2c + z' - 2)C_{mul_1} + (c + 2z')C_{exp} + z'C_{mul_2}$, and the costs of [7, 39] are $3C_p + (2c + z') \cdot C_{exp} + (2c + z')C_{mul_1} + C_{mul_2}$ and $(z' + 2)C_p + (c + z')C_{exp} + (c + 2z')C_{mul_1} + z'C_{mul_2}$, respectively. Compared with the schemes in [7, 39], the cost of our scheme is slightly higher. The reason is that our scheme performs tag generation, verification, and update of ciphertext, and the computational cost is obviously higher than that in literature.

7.2. Communication Cost. In this scheme, the communication cost mainly arises from the challenge information generation phase and proof generation phase. To audit the integrity of the data stored in the cloud service, a verifier sends the challenge

Table 1: Comparison of computation cost.

Schemes	Tag generation	Proof generation	Verification proof	Data block type
Scheme in [39]	$(z'+1)\cdot(C_{exp}+C_{mul_1})$	$cC_{exp}+cC_{mul_1}$	$3C_p+(2c+z')C_{exp}+(2c+z')C_{mul_1}+C_{mul_2}$	Plaintext
Scheme in [7]	$2C_{exp}+C_{mul_1}$	$cC_{exp}+(c-z')C_{mul_1}$	$(z'+2)C_p+(c+z')C_{exp}+(c+2z')C_{mul_1}+z'C_{mul_2}$	Plaintext
Our scheme	$2C_{exp}+2C_{mul_1}$	$cC_{exp}+(c-z')C_{mul_1}$	$(z'+2)C_p+(c+2z')C_{exp}+(2c+z'-2)C_{mul_1}+z'C_{mul_2}$	Ciphertext

information (c, k_1, k_2) to the CSP, and then proof $P = (T, F)$ is returned to the verifier by CSP. The communication cost for an integrity proof challenge is $|n| + 2|q|$ bits, and the communication cost of proof information response is $2(c + z')|\mathbb{G}_1| + |n| + 2|q|$ bits, where $|q|$ is the element length in $\mathbb{Z}_q$ and $|n|$ is the length of the element in set $\{1, n\}$. In addition, the communication cost for the revocation phase is $(2(l+1)|\mathbb{G}_1|)$, where l is denotes the number of ciphertext data blocks owned by the revoked member.

7.3. Experimental Results. In this experiment, we utilized the Ubuntu Kylin 16.04 LTS (64-bit) operation system equipped with the VMware Workstation 10 with Intel Core i7-8700 3.2 GHz processor and 16 G RAM of the host computer with Win10 operation system using C language to simulate the scheme implementation environment. The Pairing Based Cryptography (PBC) [51] library (version 0.5.14) has been used to execute pairing steps and the Openssl library [52] (version 1.1.1k) is deployed to implement two hash (SHA 256) operations. For the choice of experimental parameters, we used the file params/a.param provided by PBC for type A pairing and constructed a 256-bit order elliptic curve [53] group of type A. To obtain more accurate results,all experiments were run 50 times to get an average.

The step *Encrypt* needs to execute two time-consuming calculations, namely, pairing operation and exponentiation on group $\mathbb{G}_1$, totally costing almost 602.024 ms for 100 members. We utilize the file with the size of 32 M for experimental demonstration, so the total number of blocks is 10^6 which is bounded by the order of the 256-bit group. Suppose that all blocks are averagely distributed to project members; thereby the number of members getting the blocks is 10^4. Figure 2 depicts the time cost result for the members varying from 1 to 100 to generate all ciphertext blocks. Through observation, it is found that the time consumption of encrypting operation is proportional to the number of users. It takes 5.83 ms for a single member to encrypt all its data blocks, while all members can accept the fact that it takes 602.02 ms to encrypt all data blocks. Moreover, the operation that all data blocks are clustered together and encrypted only occurs at the beginning of the project, in the distribution phase of the subproject.

Based on the cost of ciphertext generation, we now evaluate the cost of tag generation experimentally. We still leverage the 10^6 ciphertext blocks for the experiment. To carry out the experiment demonstration, we utilize the participant ciphertext number ranging from 10^5 to 10^6 with an increment of 10^5 for each test. From the experimental results in Figure 3,

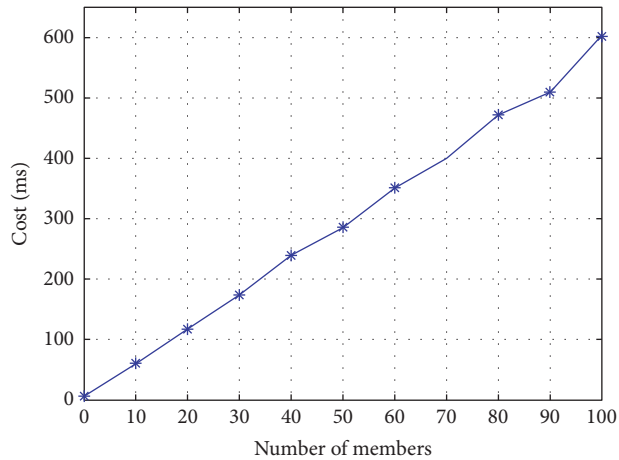


Figure 2: Computation cost of encryption.

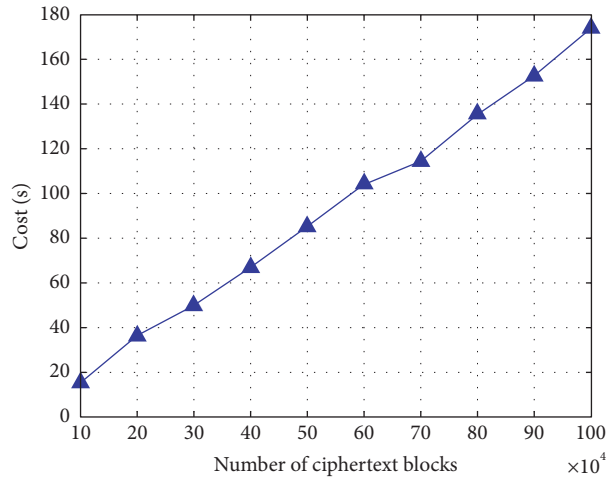


Figure 3: Computation cost of tag generation.

we can observe that the time consumption for tag generation is linear with the increase of the number of blocks, and it takes about 173.9 s to generate tags for all 10^6 blocks. As observing the proposed scheme, the entire *Encrypt* and *TagGen* processes are executed by only the project manager; thereby the project members just need to download the ciphertext and tags within the appointed time.

We set $|q| = 256$ bits, $|n| = 20$ bits, and $z' = 100$ as in previous work [39]. Based on the previous conclusions [5, 7, 39], if 1% of all blocks are corrupted, 460 challenge blocks picked randomly can achieve 99% error detection

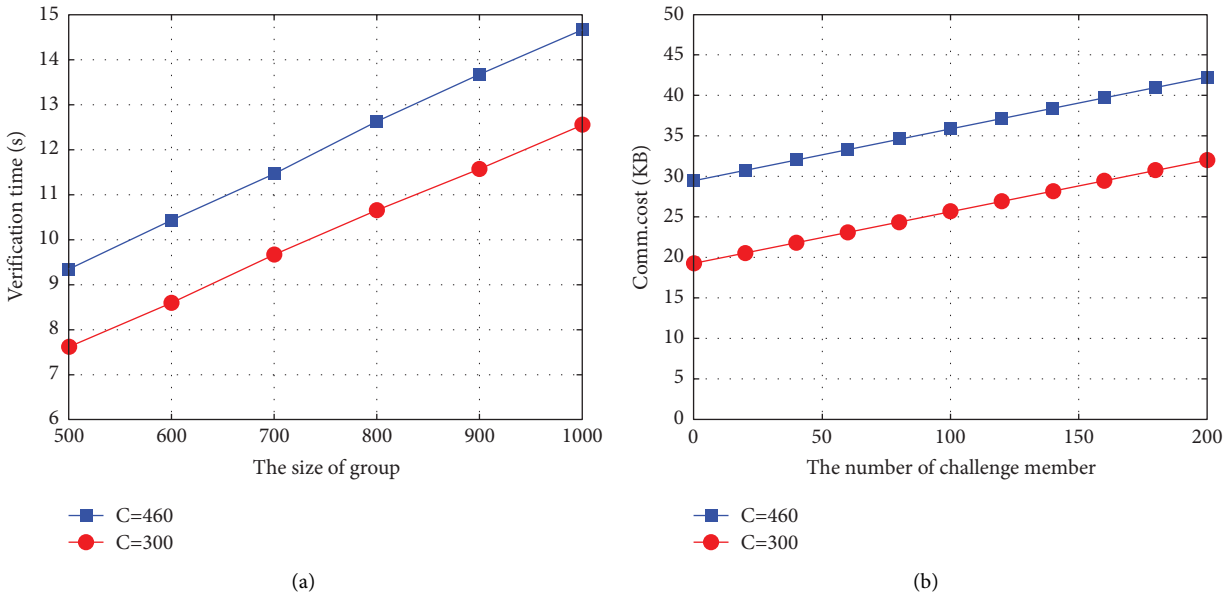


Figure 4: The computation cost of challenge operation. (a) The size of group. (b) The number of challenge members.

probability and 300 challenge blocks chosen at random can successfully achieve 95% malpractice detection probability. In Figure 4, we can see that when the size of project group varies from 500 to 1000 and the number of challenge members ranges from 0 to 200, our scheme can achieve an auditing task with the maximum verification time of 14.664 s and 42 KB by choosing $c = 460$.

8. Conclusion

In this paper, a remote encrypted data integrity auditing scheme stored on a cloud service provider is presented. This scheme addresses the integrity auditing issue for the encrypted data which is shared with numerous members of a group. In our scheme, the sponsor of a shared data group is the project manager who is responsible for the initialization of system parameters, the selection of partial private keys for project members, and the generation of original ciphertext blocks for subprojects. Meanwhile, with the help of certificateless signature idea, the synchronization change between the ciphertext block and the tag is realized, and the problem of auditing the integrity of the ciphertext block is transformed into an equation verification related to the tag. Therefore, based on the above two measures, the key escrow and certificate management in PKI naturally do not exist. With regard to the revocation of the member, our scheme utilizes the homomorphic hash function to transform the ciphertexts of the revoked members into the ciphertexts of the existing members without leaking the information of ciphertext. Finally, the protocol has been proven secure to satisfy adaptively selective the ciphertext attack assuming the stability of CDH and DL in bilinear pairing. From the results of the experiment, our scheme is efficient in both computation and communication cost and more secure in a shared group in cloud storage.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61772008, in part by the Science and Technology Major Support Program of Guizhou Province, China, under Grant 20183001, in part by the Key Program of the National Natural Science Union Foundation of China under Grant U1836205, in part by the Science and Technology Program of Guizhou Province under Grant ZK[2021]325, in part by the Science and Technology Program of Guiyang under Grant [2021]1–5, and in part by the Science and Technology Planning Project of Tongren Municipality under Grant [2020]78.

References

[1] Dropbox for business. [Online]. Available: https://www.dropbox.com/business.

[2] Tortoisesvn. [Online]. Available: https://tortoisesvn.net/.

[3] L. Shuib and E. Yadegaridehkordi, "Big data adoption: state of the art and research challenges," *Information Processing & Management*, vol. 56, no. 6, 2019.

[4] Y. Deswarte and A. Sa¨ıdane, "Remote integrity checking-how to trust files stored on untrusted servers," in *Integrity and Internal Control in Information Systems VI - IFIP TC11/WG11.5*, Springer, Lausanne, Switzerland, 2003.

[5] Giuseppe Ateniese, R. C. Burns, R. Curtmola et al., "Provable data possession at untrusted stores," in *Proceedings of the 2007*

ACM Conference on Computer and Communications Security, CCS 2007, P. F. Syverson, Ed., Alexandria, USA, October 2007.

[6] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 9, pp. 1717–1726, 2013.

[7] J. Li, H. Yan, and Y. Zhang, "Certificateless public integrity checking of group shared data on cloud storage," *IEEE Transactions on Services Computing*, vol. 14, no. 1, pp. 71–81, 2021.

[8] H. Wang, D. He, and S. Tang, "Identity-based proxy-oriented data uploading and remote data integrity checking in public cloud," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1165–1176, 2016.

[9] J. Li, W. Yao, Y. Zhang, H. Qian, and J. Han, "Flexible and fine-grained attribute-based data storage in cloud computing," *IEEE Transactions on Services Computing*, vol. 10, no. 5, pp. 785–796, 2017.

[10] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 5, pp. 847–859, 2011.

[11] H. Yan, J. Li, J. Han, and Y. Zhang, "A novel efficient remote data possession checking protocol in cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 78–88, 2017.

[12] J. Li, Y. Wang, Y. Zhang, and J. Han, "Full verifiability for outsourced decryption in attribute based encryption," *IEEE Transactions on Services Computing*, vol. 13, no. 3, pp. 478–487, 2020.

[13] J. Li, X. Lin, Y. Zhang, and J. Han, "Ksf-oabe: outsourced attribute-based encryption with keyword search function for cloud storage," *IEEE Transactions on Services Computing*, vol. 10, no. 5, pp. 715–725, 2016.

[14] Y. Yu, M. H. Au, G. Ateniese et al., "Identity-based remote data integrity checking with perfect data privacy preserving for cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 767–778, 2017.

[15] Y. Zhang, C. Xu, X. Liang, H. Li, Y. Mu, and X. Zhang, "Efficient public verification of data integrity for cloud storage systems from indistinguishability obfuscation," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 3, pp. 676–688, 2017.

[16] H. Wang, Q. Wu, B. Qin, and J. Domingo-Ferrer, "Identity-based remote data possession checking in public clouds," *IET Information Security*, vol. 8, no. 2, pp. 114–121, 2014.

[17] F. Sebé, J. Domingo-Ferrer, A. Mart´ınez-Ballesté, Y. Deswarte, and J. J. Quisquater, "Efficient remote data possession checking in critical information infrastructures," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 8, pp. 1034–1038, 2008.

[18] C. Christopher Erway, A. Küpc¸ü, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in *Proceedings of the 2009 ACM Conference on Computer and Communications Security, CCS 2009*, E. Al-Shaer, S. Jha, and A. D. Keromytis, Eds., pp. 213–222pp. 213–, Chicago, Illinois, USA, November 2009.

[19] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, W. Lou, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Transactions on Computers*, vol. 62, no. 2, pp. 362–375, 2013.

[20] Y. Yu, Y. Zhang, J. Ni, M. H. Au, L. Chen, and H. Liu, "Remote data possession checking with enhanced security for cloud storage," *Future Generation Computer Systems*, vol. 52, pp. 77–85, 2015.

[21] Y. Feng, G. Yang, and J. K. Liu, "A new public remote integrity checking scheme with user and data privacy," *International Journal of Applied Cryptography*, vol. 3, no. 3, pp. 196–209, 2017.

[22] H. Wang, "Identity-based distributed provable data possession in multicloud storage," *IEEE Transactions on Services Computing*, vol. 8, no. 2, pp. 328–340, 2015.

[23] A. Juels, S. Burton, and J. Kaliski Jr, "Pors: proofs of retrievability for large files," in *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007*, P. Ning, S. De Capitani di Vimercati, and P. F. Syverson, Eds., pp. 584–597, Alexandria, Virginia, USA, October 2007.

[24] K. D. Bowers, A. Juels, and A. Oprea, "HAIL: a high-availability and integrity layer for cloud storage," in *Proceedings of the 2009 ACM Conference on Computer and Communications Security, CCS 2009*, E. Al-Shaer, S. Jha, and A. D. Keromytis, Eds., Chicago, Illinois, USA, November 2009.

[25] B. Wang, B. Li, and H. Li, "Knox: privacy-preserving auditing for shared data with large groups in the cloud," in *Applied Cryptography and Network Security*, F. Bao, P. Samarati, and J. Zhou, Eds., vol. 7341, pp. 507–525, 2012.

[26] B. Wang, H. Li, and M. Li, "Privacy-preserving public auditing for shared cloud data supporting group dynamics," in *Proceedings of the IEEE International Conference on Communications, ICC*, Budapest, Hungary, June 2013.

[27] X. Liu, Y. Zhang, B. Wang, and J. Yan, "Mona: secure multi-owner data sharing for dynamic groups in the cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 6, pp. 1182–1191, 2013.

[28] J. Yuan and S. Yu, "Public integrity auditing for dynamic data sharing with multiuser modification," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 8, pp. 1717–1726, 2015.

[29] B. Wang, B. Li, and H. Li, "Panda: public auditing for shared data with efficient user revocation in the cloud," *IEEE Transactions on Services Computing*, vol. 8, no. 1, pp. 92–106, 2015.

[30] Y. Yu, Y. Mu, J. Ni, J. Deng, and K. Huang, "Identity privacy-preserving public auditing with dynamic group for secure mobile cloud storage," in *Proceedings of the Network and System Security - 8th International Conference, NSS 2014*, M. Ho Au, B. Carminati, and C.-C. Jay Kuo, Eds., pp. 28–40, Xi'an, China, October 2014.

[31] Giuseppe Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in *Proceedings of the 4th International ICST Conference on Security and Privacy in Communication Networks, SECURECOMM 2008*, L. Albert, P. Liu, and R. Molva, Eds., September 2008.

[32] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in *Proceedings of the Computer Security - ESORICS 2009, 14th European Symposium on Research in Computer Security*, pp. 355–370, Saint-Malo, France, September 2009.

[33] Y. Zhu, H. Wang, Z. Hu et al., "Dynamic audit services for integrity verification of outsourced storages in clouds," in *Proceedings of the 2011 ACM Symposium on Applied Computing (SAC)*, W. C. Chu, W. Eric Wong, M. J. Palakal, and C.-C. Hung, Eds., TaiChung, Taiwan, March 2011.

[34] L. Chang, J. Chen, L. T. Yang et al., "Authorized public auditing of dynamic big data storage on cloud with efficient

verifiable fine-grained updates," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 9, pp. 2234–2244, 2014.

[35] Li Jin, X. Tan, X. Chen, D. S. Wong, and F. X. Opor, "Enabling proof of retrievability in cloud computing with resource-constrained devices," *IEEE Transactions on Cloud Computation*, vol. 3, no. 2, pp. 195–205, 2015.

[36] Z. Yang, W. Wang, Y. Huang, and X. Li, "Privacy-preserving public auditing scheme for data confidentiality and accountability in cloud storage," *Chinese Journal of Electronics*, vol. 28, no. 1, pp. 179–187, 2019.

[37] N. Garg and S. Bawa, "Id-papc: identity based public auditing protocol for cloud computing," in *Proceedings of the 2017 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, pp. 14–17, IEEE, Dehradun, India, December 2017.

[38] R. S. Bali and N. Kumar, "Secure clustering for efficient data dissemination in vehicular cyber-physical systems," *Future Generation Computer Systems*, vol. 56, pp. 476–492, 2016.

[39] B. Wang, B. Li, H. Li, and F. Li, "Certificateless public auditing for data integrity in the cloud," in *Proceedings of the IEEE Conference on Communications and Network Security, CNS 2013*, pp. 136–144, National Harbor, MD, USA, October 2013.

[40] M. Etemad and A. Küpc¸ü, "Generic efficient dynamic proofs of retrievability," in *Proceedings of the 2016 ACM on Cloud Computing Security Workshop, CCSW 2016*, E. R. Weippl, S. Katzenbeisser, M. Payer, S. Mangard, E. Androulaki, and M. K. Reiter, Eds., ACM, Vienna, Austria, pp. 85–96, October 2016.

[41] H. Zhuo Hao, S. Sheng Zhong, and N. Nenghai Yu, "A privacy-preserving remote data integrity checking protocol with data dynamics and public verifiability," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 9, pp. 1432–1437, 2011.

[42] W. Shen, J. Qin, Y. Jia, H. Rong, J. Hu, and J. Ma, "Data integrity auditing without private key storage for secure cloud storage," *IEEE Transactions on Cloud Computing*, vol. 19, 2019.

[43] T. Wu, G. Yang, Y. Mu, F. Guo, R. H. Deng, and Deng, "Privacy-preserving proof of storage for the pay-as-you-go business model," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 2, pp. 563–575, 2021.

[44] B. Wang, B. Li, and H. Li, "Knox: privacy-preserving auditing for shared data with large groups in the cloud," in *Applied Cryptography and Network Security*, F. Bao, P. Samarati, and J. Zhou, Eds., vol. 7341, pp. 507–525, Springer, 2012.

[45] B. Wang, B. Li, and H. Li, "Oruta: privacy-preserving public auditing for shared data in the cloud," in *Proceedings of the 2012 IEEE Fifth International Conference on Cloud Computing*, R. Chang, Ed., Honolulu, HI, USA, June 2012.

[46] H. Yao, C. Wang, Bo Hai, and S. Zhu, "Homomorphic hash and blockchain based authentication key exchange protocol for strangers," in *Proceedings of the 2018 Sixth International Conference on Advanced Cloud and Big Data (CBD)*, pp. 243–248, IEEE, Lanzhou, China, August 2018.

[47] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security*, pp. 452–473, Springer, Daejeon, South Korea, December 2003.

[48] J. Li, H. Yan, and Y. Zhang, "Identity-based privacy preserving remote data integrity checking for cloud storage," *IEEE Systems Journal*, vol. 15, no. 1, pp. 577–585, 2021.

[49] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway, "Relations among notions of security for public-key encryption schemes," in *Proceedings of the 18th Annual International Cryptology Conference*, H. Krawczyk, Ed., Springer, Santa Barbara, California, USA, pp. 26–45, August 1998.

[50] N. Garg, S. Bawa, and N. Kumar, "An efficient data integrity auditing protocol for cloud computing," *Future Generation Computer Systems*, vol. 109, pp. 306–316, 2020.

[51] B. Lynn. Pairing-based Cryptography Library. https://crypto.stanford.edu/pbc/download.html.

[52] Toolkit Openssl Project. Openssl library. https://www.openssl.org/docs/manmaster/man7/crypto.html.

[53] D. R. L. Brown, "Sec 2: recommended elliptic curve domain parameters," *Standars for Efficient Cryptography*, vol. 20, 2010.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 5251228, 8 pages
https://doi.org/10.1155/2022/5251228

Research Article

Credit Evaluation of SMEs Based on GBDT-CNN-LR Hybrid Integrated Model

Lei Zhang[1,2] **and Qiankun Song**[2]

[1]*School of Economic and Management, Chongqing Jiaotong University, Chongqing 400074, China*
[2]*School of Mathematics and Statistics, Chongqing Jiaotong University, Chongqing 400074, China*

Correspondence should be addressed to Lei Zhang; zhangleicqjtu@163.com

Received 30 December 2021; Revised 19 January 2022; Accepted 21 January 2022; Published 11 February 2022

Academic Editor: Yingjie Wang

Under the background of the increasing demand for credit evaluation and risk prediction, the establishment of an effective credit evaluation model for small- and medium-sized enterprises has become a research hotspot. Based on previous studies, this paper proposes a two-layer feature extraction method based on Gradient Boosting Decision Tree (GBDT) and Convolutional Neural Network (CNN). First, based on the original features, GBDT is used to combine and automatically screen them, the missing values in the feature are processed, and the transformed high-dimensional sparse features are obtained. Then, CNN is used to extract features further, and finally, the logistic regression (LR) model is used to predict. In the simulation experiment, this paper takes a dataset of 14,366 small- and medium-sized enterprise credit evaluations as the analysis samples to verify the results. The results show that the GBDT-CNN-LR model has the best performance. The model also shows good generalization ability and stability in the reliability test.

1. Introduction

For the credit financing of small- and medium-sized enterprises, on the one hand, due to their small scale, high operating, and capital flow risks, financing channels and financing limits will be restricted; on the other hand, the high debt repayment risk and fraudulent behavior of small- and medium-sized enterprises will bring a huge risk of capital loss to the banking industry. How to address the problems of financing difficulties and high credit risks for small- and medium-sized enterprises caused by the asymmetry of information between the two parties to establish a high-precision credit evaluation and prediction model has become the focus of current research.

The SME credit evaluation based on artificial intelligence algorithms has high accuracy and fast speed, which are more often used in the bank credit evaluation business. At the same time, the requirements for the accuracy of the evaluation algorithm are also increasing. Scholars have done extensive research on machine learning algorithms for SME credit classification prediction, including statistical methods, single machine learning algorithms, integrated learning algorithms, and multimodel hybrid integrated learning algorithms [1–4]. Compared with credit evaluation methods based on machine learning algorithms, traditional statistical methods often require more complicated feature engineering in the early stage, which is not only inefficient, but the accuracy of the model is largely affected by the early feature engineering work. The data mining models of machine learning algorithms mainly include artificial neural networks [5–8], support vector machines [9–11], and decision trees [12, 13]. Huang et al. [14] compared the classification accuracy and applicability of several common neural network models. The empirical results show that the probabilistic neural network (PNN) has the lowest classification error rate. Uddin et al. [15] applied the random forest (RF) method to the robust modeling of credit default prediction, which has been proven as an efficient classifier than others. Wang et al. [16] selected appropriate indicators and used an improved SVM model for analysis to be able to

detect the credit risk of SMEs. Luo et al. [17] used a deep learning network and applied a deep belief network with Restricted Boltzmann Machines to credit scoring, which has higher accuracy than that of traditional logistic regression methods. Zhong et al. [18] compared the machine learning training effects of BP, ELM, I-ELM, and SVM, and the results showed that the effects of ELM and BP neural networks are better.

The characteristics of missing values, high dimensionality, and redundancy in the credit evaluation of small- and medium-sized enterprises make it difficult to find the optimal evaluation feature integration of the evaluation classifier, which is also a key factor that leads to the low accuracy of the current evaluation classification. In order to further enhance the evaluation effect, algorithm research based on hybrid integrated machine learning has been innovated and improved for the existing problems so that the integrated model is better than the original model in various evaluation indicators of the predicted results. The RS-PSO-SVM model [19] solves the problem of nonlinear modeling and multicollinearity, which has high accuracy and efficiency. It uses the PSO algorithm to optimize the SVM model parameters and to assess and classify corporate credit risks. Sun et al. [20] combined SMOTE and Bagging to propose the DTE-SBD model, which can not only dispose of the class imbalance problem of enterprise credit evaluation but also increase the diversity of base classifiers for DT ensemble. Ma [21] put forward a hybrid integrated method RS-Boosting based on boosting and random subspace sampling to predict corporate credit risk and verified the effectiveness and feasibility of the method through empirical comparisons. Arora and Kaur [22] used the Bolasso algorithm to select consistent and relevant features from the feature library and applied the generated candidate features to different classification algorithms such as the random forest. The results showed that the BS-RF algorithm has a good performance in the classification accuracy of credit evaluation.

The credit evaluation of SMEs has complex features and high redundancy, and the evaluation data often contain a lot of missing values. Therefore, when using machine learning methods for corporate credit evaluation, high requirements are often placed on the processing of missing data in the early stage, and good feature engineering is also required. However, most of the above models simply remove the redundant features in the metadata and put their subsets into one or several base models for training. However, they do not compare and verify the results of the selected subsets based on different base models. In addition, when the number of feature indicators in the dataset changes, the original model will no longer be applicable.

Aiming at the shortcomings of existing research, this paper proposes a hybrid ensemble model using the GBDT-CNN method for feature extraction to evaluate corporate credit. The model uses the GBDT-CNN method to extract the original data features, which can effectively deal with the missing values of the samples while reducing the difficulty of feature engineering, thereby reducing the assumption of the data missing mechanism and the dependence on the data distribution model, which also has better robustness to abnormal situations in the original data.

2. Enterprise Credit Evaluation Techniques and Procedures

2.1. GBDT Model. Gradient Boosting Decision Tree, based on the idea of Boosting and CART algorithm, is an iterative decision tree algorithm. Except that the first decision tree is generated using the original predictive index, the goal in each iteration is to minimize the loss function of the current learner, that is, to make the loss function always drop along its gradient. Through continuous iteration, it makes the final residual error close to 0. Then by adding up the results of all trees, we can get the final prediction results [23].

The credit risk identification of SMEs is an obvious binary classification problem, which predicts risks through a series of basic corporate information, stocks, capital, investment, income, and other indicators. Let y denote the credit behavior of the enterprise, $y = 1$ denote dishonesty behavior, and $y = 0$ denote nondishonesty behavior. $x = \{x^1, x^2, \ldots, x^K\}$ is a k-dimensional variable composed of a series of basic information of the enterprise. For a training dataset $T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$ containing N samples, the GDBT modeling process is as follows:

$$f_0(x) = \arg_c \min \sum_{i=1}^{N} L(y_i, c), \tag{1}$$

where $f_0(x)$ is the initial decision tree with only one root node, y_i is the i-th training data, c is the constant that minimizes the loss function $f(x)$, and $L(y_i, c)$ is the loss function.

In the GBDT model, different loss functions can be used for binary classification problems, but log-likelihood is generally used:

$$L(y, f(x)) = \log(1 + \exp(-yf(x))), \tag{2}$$

where f is the binary classification model to be solved.

Let the number of iterations be $m = 1, 2, \ldots, M$, and then the negative gradient of the i-th training sample is

$$r_{mi} = -\left[\frac{\partial L(y, f(x_i))}{\partial f(x_i)}\right]_{f(x)=f_{m-1}(x)} = \frac{y_i}{1 + \exp(yf(x_i))}. \tag{3}$$

According to all samples and their negative gradient directions (x_i, r_{mi}) $(i = 1, 2, \ldots, N)$, a decision tree T_m composed of J leaf nodes is obtained. The j-th leaf node area is R_{mj} $(j = 1, 2, \ldots, J)$, and the best fit value of each leaf node is

$$c_{mj} = \arg\min \sum_{c_{x_i} \in R_{mj}} \log(1 + \exp(y_i f_{m-1}(x_i) + c)). \tag{4}$$

The learners obtained in this round are

$$f_m(x) = f_{m-1}(x_i) + \sum_{i=1}^{N} \sum_{j=1}^{J} c_{mj} I \quad x_i \in R_{mj}, \tag{5}$$

where I is the indicative function of the i-th training sample in the j-th leaf node region and

$$I = \begin{cases} 1, & X_i \in R_{mj}, \\ 0, & X_i \notin R_{mj}. \end{cases} \tag{6}$$

After M rounds of iteration, the final decision model is

$$f(x) = f_M(x) = c + \sum_{m=1}^{M} \sum_{j=1}^{J} c_{mj} I \quad x \in R_{mj}. \tag{7}$$

According to the number of times the variable is selected as the split variable in the regression tree during the iteration process and the degree of improvement of the model during the split process, the importance of each variable can be obtained as

$$\begin{aligned} R_k^2 &= \frac{1}{M} \sum_{m=1}^{M} R_k^2(T_m), \\ R_k^2(T_m) &= \sum_{j=1}^{J} E_j^2 I_j\left(x^k\right), \end{aligned} \tag{8}$$

where T_m is the decision tree trained in the m-th iteration, $I_j(x^k)$ is the k-th variable x^k , which is selected as the indicator function of the j-th leaf node split variable in the decision tree T_m, E_j^2 denotes the improvement of the prediction result when the variable x^k is used as the leaf separate variable, and R_k^2 represents the importance value of the variable x^k in the decision tree.

2.2. CNN. Convolutional Neural Network (CNN) consists of one or more convolutional layers and a fully connected layer, which also includes associated weights layers and pooling layers. CNN's features such as local connection, weight sharing, and pooling processing can effectively reduce network complexity and decrease the number of training parameters. To some extent, they make the model have a certain degree of invariance to translation, distortion, and scaling. While maintaining strong robustness and fault tolerance, it is also easy to train and optimize the network structure [2, 7, 24].

Here, this paper will map the combined feature and feature classification automatically (searched by GBDT) to higher dimensions through the CNN to truly reflect the distribution of the data.

2.3. Logistic Regression. Logistic regression is used for classification problems. The decision boundary can be expressed as $w_1 x_1 + w_2 x_2 + b = 0$, assuming that a certain sample point satisfies the condition $h_w(x) = w_1 x_1 + w_2 x_2 + b > 0$. Then, the category is judged as 1. For the binary classification problem, the given dataset is as follows:

$$\begin{aligned} D = (x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N), \quad x_i \subseteq R^n, \\ y_i \in 0, 1, \quad i = 1, 2, \ldots, N. \end{aligned} \tag{9}$$

Because the value of $w_T x + b$ is continuous, it is used to fit the conditional probability $p(Y = 1|x)$. However, for $w \neq 0$, the value of $w_T x + b$ is R, and the probability of nonconformity ranges from 0 to 1, so we use a generalized linear model. The unit step function is as follows:

$$p(Y = 1 \mid x) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \quad z = w^T x + b. \\ 1, & z > 0, \end{cases} \tag{10}$$

The step function is not differentiable, and the log probability function is a commonly used substitute function:

$$y = \frac{1}{1 + e^{-(w^T x + b)}}. \tag{11}$$

Then, there are

$$\ln(\text{odds}) = \ln \frac{y}{1 - y}. \tag{12}$$

Regarding y as a class posterior probability estimation,

$$\begin{aligned} w^T x + b &= \ln \frac{P(Y = 1 \mid x)}{1 - P(Y = 1 \mid x)}, \\ P(Y = 1|x) &= \frac{1}{1 + e^{-(w^T x + b)}}. \end{aligned} \tag{13}$$

The output $Y = 1$ log odds is a model represented by a linear function of the input x, that is, a logistic regression model. The closer the value of $w_T x + b$ is to positive infinity, the closer the value of $P(Y = 1|x)$ probability is to 1. Therefore, logistic regression first fits the decision boundary and then establishes the probability link between this boundary and the classification, which gives the probability in the dichotomous case.

3. Enterprise Credit Evaluation Model GBDT-CNN-LR

The samples used by SMEs for credit evaluation often contain a large amount of missing data. The use of machine learning and other methods for credit evaluation has high requirements for the processing of missing data in the early stage. In addition, features of SMEs' credit evaluation have the characteristics of large number, complexity, and high redundancy. Traditional machine learning methods must be based on good feature engineering in the early stage. Therefore, finding the optimal evaluation feature set of the evaluation classifier is the key to improving the accuracy of the evaluation classification. Most of the existing missing value processing methods use certain approaches to fill in data artificially. It is necessary to assume that the dataset obeys a certain distribution model. However, in practical applications, the feature missing data are often intertwined. If the assumptions and models are unreasonable, they will affect the follow-up learning effect of the classifier.

According to the analysis above, if a method adopted can make full use of the information contained in the known dataset, there is no need for the bank and other financial

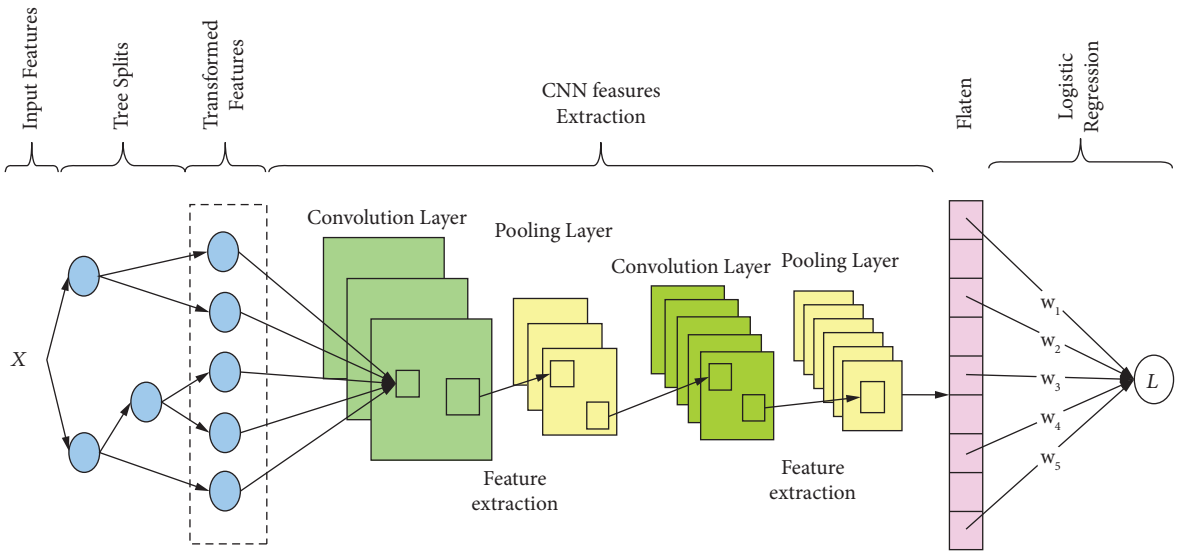


Figure 1: Frame diagram of GBDT-CNN-LR model.

institutions to process the missing data before they classify SMEs' credit, thereby reducing the assumption of the data missing mechanism and the dependence on the data distribution model. Thus, it improves the quality of the evaluation feature set in the evaluation classifier, thereby enhancing the classification accuracy. Therefore, this research mainly focuses on how to simplify the preliminary feature processing for enterprise credit data as much as possible so as to achieve the highest possible discrimination accuracy while realizing feature extraction and feature combination.

This problem can be considered from two aspects: first, compared with human feature engineering, whether the method adopted can reflect the information covered by the original data features so as to ensure the correct rate of subsequent classification of untrustworthy companies; second, whether the adopted method can better adapt to and deal with the outliers and missing values in the data, including whether it is sensitive to the data and whether it can still maintain high accuracy even in the case of massive data distribution.

Therefore, this paper proposes a method based on GBDT-CNN to extract the features of the original data. First of all, it is based on the idea of Boosting. In the GBDT feature generation part, except for the first decision tree generated by the original predictor, the goal of each subsequent iteration will minimize the loss function of the current learner; that is, the loss function always descends along its gradient, and the final residual error tends to 0 through continuous iterations. Finally, the prediction result can be obtained by combining the results of all the trees through a specific aggregation function. Different from the traditional model, this paper uses GBDT as a tool to automatically combine and filter the features of the original data, discover distinguishable features, and generate new feature combinations, thereby retaining the information contained by the original data. In addition, when the loss function is properly selected, GBDT has strong robustness to abnormal conditions in the original dataset and is not sensitive to hyperparameters. It can achieve good prediction accuracy without long-time parameter adjustments. Considering that the original dataset has two types, continuous and discrete values, GBDT can also handle them flexibly without preceding operations, which simplifies the complexity of early feature engineering.

In the GBDT model section, each original data sample will eventually fall on the leaf node of the tree, and after the One-Hot encoding is connected, the transformed high-dimensional sparse feature vector is obtained. This paper then uses CNN with Batch Normalization as a further feature extraction tool to find higher-dimensional features to improve classification accuracy. The specific implementation methods are as follows. First, this paper uses BN to standardize the input data of each layer of the network to ensure that the mean and variance of the input distribution are stable within a certain range. While alleviating the Internal Covariate Shift problem in the network, it also alleviates the disappearance of the gradient to a certain extent and accelerates the convergence of the model. Second, BN makes the network more robust to parameters and activation functions and reduces the complexity of training and tuning of the neural network model. Third, the BN training process uses the Mini Batch mean and variance as the overall sample statistics estimation and introduces random noise. To a certain extent, they have a regularization effect on the model and enhance the robustness of the model.

After extracting the characteristics of the original data through the GBDT-CNN method, the classification model is then used to identify and discriminate the untrustworthy enterprises. LR (logistic regression) is a kind of generalized linear model. The output is the weighted sum of the input features, and the final result is output by the Sigmoid function so that it lies between 0 and 1, which conforms to the meaning of probability. The credit evaluation of an enterprise is to conclude whether to lend or not after comprehensively inspecting various financial and operating indicators of the enterprise. Therefore, the logistic regression

Table 1: Comparison of evaluation indexes of different models.

Model	Accuracy	f1_score	Recall_score
GDBT-LR	0.9349	0.9565	0.9445
GDBT-CNN-LR	0.997	0.9782	0.9558
Random Forest Classifier	0.9491	0.9495	0.9448
Decision Tree Classifier	0.9357	0.9364	0.9354
Logistic regression	0.8177	0.8027	0.7327
SVM	0.5107	0.4439	0.386
MLP	0.5107	0.7404	0.6416
GaussianNB	0.5107	0.7647	0.9636
KNN	0.5107	0.7805	0.8456

model can be better applied to the problem of enterprise credit evaluation, and it is easy to explain the importance of each evaluation index to the final evaluation result.

Based on the analysis and discussion above, this paper aims to establish a GBDT-CNN-LR-based credit risk assessment model for SMEs. The frame diagram is shown in Figure 1.

For the use of integrated learning methods for enterprise credit evaluation, we need to consider two factors: (1) whether the model can effectively identify untrustworthy companies from the sample, that is, the accuracy requirements; (2) whether the weak learning model of the model can produce a difference, to avoid the degradation of the model effect, that is, the requirement of diversity. Regarding the first question, using the GBDT-LR model to solve the prediction of Facebook ad clicks in previous studies, the GBDT-LR model can better solve the prediction problem and achieve higher accuracy, which is sufficient to explain that the GBDT-CNN-LR model has a certain application basis, and it is possible to achieve certain recognition accuracy. For the second aspect, GBDT draws on the idea of Boosting in the training process. Every training reduces the residual of the previous training model so that the residual is reduced in the gradient direction, and each classification tree constructed reduced the error in the previous step. Thus, GBDT pays more attention to those samples with larger gradients. It can be considered that each classification decision tree constructed afterward only pays attention to some of its subsamples. Compared with the forecast of ad clicks, enterprise credit evaluation requires a higher accuracy rate. If the evaluation result is wrong, it may cause huge economic losses to the bank. In actual experiments, the traditional GBDT-LR model is still difficult to achieve the expected high accuracy rate. The accuracy rate of LR is limited by the previous feature engineering. Therefore, this paper proposes to use CNN on the basis of the feature vector generated by GBDT. The intention is to find higher-dimensional features as input data to improve the prediction accuracy of LR regression.

4. Experiments and Discussion

4.1. Datasets. The experimental dataset contains the credit records of 14,366 small- and medium-sized enterprises and 14 characteristics, including company stock price, foreign investment, registered capital, corporate assets, income,

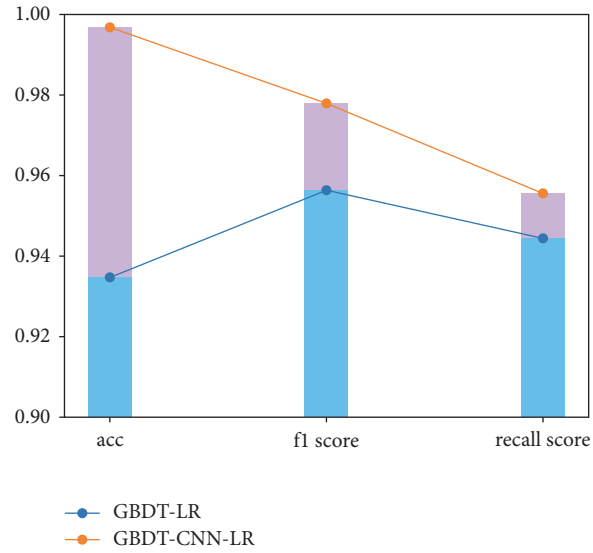


Figure 2: Before and after adding CNN convolution.

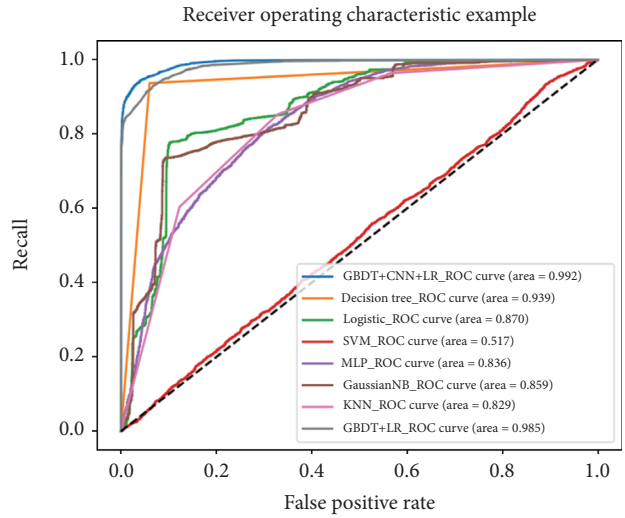


Figure 3: ROC_AUC curve.

expenses, liabilities, and taxation, which are selected as the credit evaluation indicators of small- and medium-sized enterprises.

4.2. Evaluation Index. The accuracy is used as the most important evaluation index, that is, the number of samples that are predicted correctly divided by the total number of samples, and the f1_score coefficient and recall_score are used as auxiliary evaluation indicators.

4.3. The Result of the Experiment. First of all, this paper conducts statistical analysis on the missing values of each feature in the sample set. Most of the features in the sample set used in this paper have 60% or more missing data, which verifies the universality of the problem that this paper aims to solve. Therefore, this paper uses the proposed GBDT-CNN model to search for the distribution and information

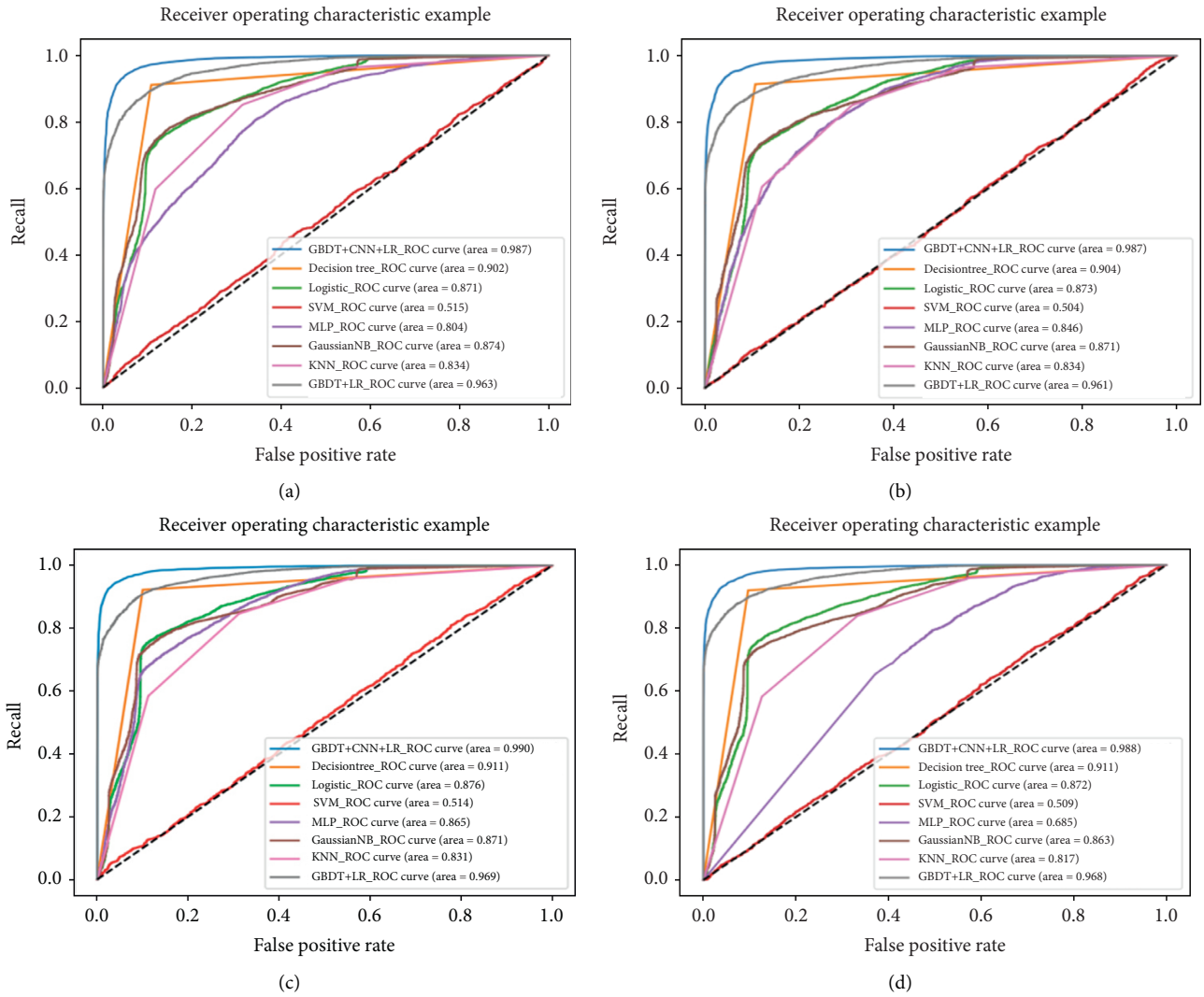


Figure 4: ROC_AUC graph of the subdataset. (a) Sample (a). (b) Sample (b). (c) Sample (c). (d) Sample (d).

of the data itself and automatically fill in the missing data. The new feature vector generated is substituted into the Logistic model as an input index to output the discrimination result.

First, compare the evaluation effects of the single model and the integrated model, and the results are shown in Table 1.

It can be seen from Table 1 that both the tree model and the logistic regression model can achieve better prediction accuracy, but the prediction accuracy rate of the SVM, MLP, NB, and KNN models is only 51.07%. The three evaluation indicators (accuracy, f1_score, and recall_score) of the model after adding CNN to extract features are higher than those of other models.

When CNN has not been added to models to extract features, the effects of random forest, decision tree, and GBDT are significantly better than those of the Logistic model. Since logistic regression is a linear model, random forest, decision tree, and GBDT are all nonlinear models. And they perform better than logistic regression on many nonlinear datasets and linear datasets. Therefore, the linearity of the Logistic model itself limits the predictive ability of the model to explain this phenomenon reasonably.

This paper uses the GBDT model to extract features and then adds the Logistic model for classification, and the prediction accuracy is 93.49%, which is worse than that of a single model such as random forest and decision tree. Therefore, this paper considers further optimization of the model. Since the features automatically filtered out by the GBDT model have high dimensionality and large sparseness, this paper first uses CNN to convolve and sum the features obtained by GBDT and move them from a highly sparse space to a reasonably sparse space, which not only satisfies the certain sparsity required by logistic regression but also maintains the difference between each feature.

The experiment shown in the following figure compares the evaluation effect of the GBDT-CNN-LR model with CNN and that without CNN.

It can be seen from Figure 2 that, after adding CNN to extract features, compared with the GBDT-LR model without adding CNN to extract features, the accuracy is

increased by 4.6%. In addition to the evaluation indicators above, the ROC_AUC curve can more accurately judge the performance of the GBDT-CNN-LR model by the AUC area. Therefore, this paper draws the ROC_AUC curve of different models. As shown in Figure 3, GBDT-CNN-LR's AUC area is 0.992, which is larger than the AUC area of other models. Therefore, it can be considered that the GBDT-CNN-LR model that joins CNN to extract features is reasonable and has higher prediction accuracy for evaluating the credit risk of small- and medium-sized enterprises.

The missing values of the sample data account for a relatively large amount, reaching 42.6% of the total dataset. Using GBDT-CNN to automatically fill missing values has achieved high prediction accuracy, but if the new data does not fit the sample model, the model is very likely to be unstable. Therefore, this paper tests the stability of the model.

The dataset is divided into 4 parts, and each dataset retains the same missing rate as the original dataset. Then, we train each small dataset and draw the corresponding ROC_AUC curve graph, compare the AUC area of the model, and judge the stability of the model. The results are shown in Figure 4.

The results show that the prediction accuracy of the support vector machine model is still poor, and the multilayer perceptron (MLP) fluctuates sharply. The reason may be that the neural network is more sensitive to data, there is too little data, or there are too many missing values. Thus, the training of a neural network has a large error. The AUC area of the GBDT-LR model without the CNN channel showed a downward trend of about 2%–3%, but the AUC area of the GBDT-CNN-LR model using the CNN channel almost did not decrease. Therefore, the GBDT-CNN-LR model can show good generalization ability and stability on both large datasets and small datasets. The GBDT-LR model without the CNN channel also has good generalization ability and stability, but they are lower than those of the GBDT-CNN-LR model numerically.

5. Conclusions

The application of SME credit evaluation based on artificial intelligence algorithms in the bank credit evaluation business is becoming more and more extensive; thus, the accuracy of the evaluation model and algorithm also puts forward higher requirements. This paper proposes the GBDT-CNN-LR evaluation model. The model first uses GBDT to automatically combine and filter the original data features, which can better deal with problems such as the concentration of missing indicator values, and obtain transformed high-dimensional sparse feature vectors. Then, on the basis of the feature vector generated by GBDT, CNN is used for further feature extraction, and finally, these higher-dimensional features are predicted by logistic regression. In the simulation experiment, compared with the Random Forest Classifier, Decision Tree Classifier, Logistic Regression, SVM, and other basic classification algorithms, it can be clearly seen that the accuracy of the GDBT-CNN-LR model is higher than other models. In addition, the model shows good generalization ability and stability in the reliability test, which can effectively reduce the risk of investment and provide reliable technical support for financial institutions, accordingly possessing far-reaching practical significance.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was partially supported by the Group Building Scientific Innovation Project for Universities in Chongqing (CXQT21021) and the Science and Technology Research Project of Chongqing Education Commission (KJQN202100712).

References

[1] Y. Zhu, C. Xie, G. J. Wang, and X. G. Yan, "Comparison of individual ensemble and integrated ensemble machine learning methods to predict China's SME credit risk in supply chain finance," *Neural Computing and Applications*, vol. 28, no. 1, pp. 41–50, 2017.

[2] Z. P. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[3] X. Cai, Y. Qian, Q. Bai, and W. Liu, "Exploration on the financing risks of enterprise supply chain using Back Propagation neural network," *Journal of Computational and Applied Mathematics*, vol. 367, Article ID 112457, 2020.

[4] Y. Wang, Y. Gao, Y. Li, and X. Tong, "A worker-selection incentive mechanism for optimizing platform-centric mobile crowdsourcing systems," *Computer Networks*, vol. 171, Article ID 107144, 2020.

[5] J. P. Bigus, *Ata Mining with Neural Networks: Solving Business Problems from Application Development to Decision Support*, McGraw-Hill, New York, NY, USA, 1996.

[6] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[7] Y. Wang, Z. Cai, Z.-H. Zhan, B. Zhao, X. Tong, and L. Qi, "Walrasian equilibrium-based multiobjective optimization for task allocation in mobile crowdsourcing," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 4, pp. 1033–1046, 2020.

[8] Z. Lu, Y. Wang, Y. Li, X. Tong, C. Mu, and C. Yu, "Data-driven many-objective crowd user selection for mobile crowdsourcing in industrial IoT," *IEEE Transactions on Industrial Informatics*, 2021.

[9] S. Andaryani, V. Nourani, A. T. Haghighi, and S. Keesstra, "Integration of hard and soft supervised machine learning for flood susceptibility mapping," *Journal of Environmental Management*, vol. 291, Article ID 112731, 2021.

[10] G. Kesavaraj and S. Sukumaran, "A study on classification techniques in data mining," in *Proceedings of the 2013 fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, pp. 1–7, IEEE, Tiruchengode, India, 2013.

[11] L. Xu, L. Cui, T. Weise, X. Li, Z. Wu, and F. Nie, "Semi-supervised multi-layer convolution kernel learning in credit evaluation," *Pattern Recognition*, vol. 120, 2021.

[12] Z. Liu and Y. Zhang, "Credit evaluation with a data mining approach based on gradient boosting decision tree," *Journal of Physics: Conference Series*, vol. 1848, no. 1, p. 8, Article ID 012034, 2021.

[13] L.-A. Dong, X. Ye, and G. Yang, "Two-stage rule extraction method based on tree ensemble model for interpretable loan evaluation," *Information Sciences*, vol. 573, pp. 46–64, 2021.

[14] X. Huang, X. Liu, and Y. Ren, "Enterprise credit risk evaluation based on neural network algorithm," *Cognitive Systems Research*, vol. 52, pp. 317–324, 2018.

[15] M. S. Uddin, G. Chi, M. A. Al Janabi, and T. Habib, "Leveraging random forest in micro-enterprises credit risk modelling for accuracy and interpretability," *International Journal of Finance & Economics*, 2020.

[16] F. Wang, L. Ding, H. Yu, and Y. Zhao, "Big data analytics on enterprise credit risk evaluation of E-business platform," *Information Systems and E-Business Management*, vol. 18, pp. 1–40, 2019.

[17] C. Luo, D. Wu, and D. Wu, "A deep learning approach for credit scoring using credit default swaps," *Engineering Applications of Artificial Intelligence*, vol. 65, pp. 465–470, 2017.

[18] H. Zhong, C. Miao, Z. Shen, and Y. Feng, "Comparing the learning effectiveness of BP,ELM,I-ELM,and SVM for corporate credit ratings," *Neurocomputing*, vol. 128, no. 27, pp. 285–295, 2014.

[19] X. Hu, J. Hu, L. Chen, and Y. Li, "Credit risk assessment model for small, medium and micro enterprises based on RS-PSO-SVM integration," in *Proceedings of the 2021 IEEE 6th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*, pp. 342–345, Chengdu, China, 2021.

[20] J. Sun, J. Lang, H. Fujita, and H. Li, "Imbalanced enterprise credit evaluation with DTE-SBD: decision tree ensemble based on SMOTE and bagging with differentiated sampling rates," *Information Sciences*, vol. 425, pp. 76–91, 2018.

[21] G. Wang and J. Ma, "Study of corporate credit risk prediction based on integrating boosting and random subspace," *Expert Systems with Applications*, vol. 38, no. 11, pp. 13871–13878, 2011.

[22] N. Arora and P. D. Kaur, "A Bolasso based consistent feature selection enabled random forest classification algorithm: an application to credit risk assessment," *Applied Soft Computing Journal*, vol. 86, pp. 1–29, 2019.

[23] Z. Tian, J. Xiao, H. Feng, and Y. Wei, "Credit risk assessment based on gradient boosting decision tree," *Procedia Computer Science*, vol. 174, pp. 150–160, 2020.

[24] Y. LeCun, B. Boser, J. S. Denker et al., "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 9741278, 13 pages
https://doi.org/10.1155/2022/9741278

Research Article

An Improved Whale Optimization Algorithm Based on Aggregation Potential Energy for QoS-Driven Web Service Composition

Xuyang Teng, Yuanhao Luo, Tao Zheng, and Xuguang Zhang

School of Communication Engineering, Hangzhou Dianzi University, Hangzhou 310018, Zhejiang, China

Correspondence should be addressed to Yuanhao Luo; lyh10001@hdu.edu.cn

Received 5 November 2021; Accepted 4 January 2022; Published 7 February 2022

Academic Editor: Yingjie Wang

With more complex user needs, the web service composition (WSC) has become a key research area in the current circumstance. The swarm intelligence algorithms are proved to solve this problem well. However, no researchers have applied the whale optimization algorithm (WOA) to the WSC problem. In this work, we propose a logarithmic energy whale optimization algorithm (LEWOA) based on aggregation potential energy and logarithmic convergence factor to solve this problem. Firstly, the improved algorithm uses a chaotic strategy to enhance the initial swarm diversity. After that, a logarithmic convergence factor is applied to obtain the nonlinear search step. Furthermore, aggregation potential energy as the spatial evaluation is employed in the swarm intelligence algorithms for the first time. Finally, the aggregation potential energy is used to dynamically adjust the nonlinear weight, which improves the search efficiency and prevents the algorithm from falling into local optimization. The experimental results of the benchmark functions show that the LEWOA has better optimization ability and convergence speed than other swarm intelligence algorithms. In the second experiment of the WSC optimization, the effectiveness and superiority of the LEWOA are verified.

1. Introduction

The internet of things (IoT) has become the hot spot of information technology reform. And the research on IoT mainly focuses on privacy protection [1, 2], edge computing and data processing [3], web service composition optimization [4], and so on. In this work, we mainly study the advanced strategies of WOA and the application of the improved whale optimization algorithm in web service composition optimization problems.

Presently under the complex requirements of user needs and scene scheduling, it is far from enough only with a single web service module to provide the solution. Therefore, combining web services to solve the problem has become an inevitable choice for different task requirements that specific workflows can represent. The workflow is a combination of various web services, and each web service has quantity choices of subservices. Obviously, WSC is an NP-hard problem. In reference [5], Strunk explained the related issues and applications of service composition. Quality of service (QoS) is currently used as the criterion better to evaluate the pros and cons of web services. QoS aims to evaluate web services quantitatively. Although a large number of web services have similar functions, the QoS with different attributes of each web service is different. So the quantity of WSC tends to explode exponentially in the face of specific problems. Furthermore, the time consumption cannot be ignored if the exhaustive algorithm is selected in a large order of magnitude. The swarm intelligence algorithms can be regarded as an outstanding solution to trade off efficiency and effectiveness. In reference [6], Ouarda proposed that the swarm intelligence algorithm effectively solves such large-scale and NP-hard problems.

Here are some typical research and application of various swarm intelligence algorithms in face of WSC problem. In reference [7, 8], the authors proposed IDPSO and

IDIPSO, respectively, aiming to improve the speed and performance of finding the optimal solution of WSC problem. In reference [9], the author proposes an improved ant colony algorithm; EFACO applies a pheromone-driven scheme composed of QoS multiple weights to improve the efficiency of ant colony search. In reference [10], the genetic algorithm (GA) is used to solve the WSC problem based on elitism and an elite-based learning mechanism.

It should be noted that the traditional swarm intelligence algorithm is oriented to a continuous field, but the WSC is a discrete problem. This article will separately integer coding the candidate service set of the workflow and the multidimensional coordinates of the swarm intelligence algorithm, making them correspond to each other. Then, the traditional continuous algorithm has been transformed into a discrete optimization algorithm. Because it only converts the definition of position coordinates without changing the algorithm's optimization principles, the discrete swarm intelligence algorithm can still ensure optimization effectiveness. Nevertheless, the algorithms are more likely to fall into a local optimum in a discrete environment. Aiming to prevent such circumstances and ensure continuous optimization, we must reduce the possibility of search agents gathering. For example, swarm mutation strategy or splitting swarm are to keep the diversity of search agents in the optimization process.

However, not all swarm intelligence algorithms can be applied to WSC. The complexity and the optimization model of an algorithm are the key considerations. WOA is a new swarm intelligence algorithm proposed by Mirjalili and Lewis in reference [11] that is derived from one of the special hunting behaviours of humpback whales called the bubble-net hunting technique [12]. In reference [13], Gharehchopogh made a comprehensive survey of WOA. At present, ant colony system (AG) and particle swarm algorithm (PSO) are widely applied. The optimization principles of WOA are close to PSO in essence. WOA shows its advantages in simpler structure, fewer adjustment parameters, and more excellent global searchability. These essential factors determine WOA can better apply to optimize this problem with minor changes, which ensures the optimization properties of the algorithm will not be significantly affected.

There are still some problems with the WOA algorithm. When facing complex multimodal problems, it shows slow convergence speed and low convergence accuracy and easily falls into the local optimal demerits. Currently, three main ways are able to enhance these defects. First, combining with traditional mathematical principles: in reference [14], Chu incorporates WOA with simulated annealing algorithm to improve the searchability of the algorithm. In references [15, 16], chaotic strategy has been used to initialize the swarm to enhance the diversity of the initial state. Second, combining with the metaheuristic algorithm: in reference [17], Jadhav integrates GWO into WOA and proposes the WGC algorithm. In reference [18], Trivedi leads into the PSO model and presents PSO-WOA to improve the local searchability of WOA. Third, combining with ANN: in reference [19], WOA is combined with ANN to improve the accuracy of the image segmentation algorithm. In reference [20], ANN is used to find the optimal weight to hasten the convergence speed of WOA and enhance its capacity of jumping out of the local optima, while in reference [21], it increases the recognition accuracy of the algorithm with SVM using WOA to optimize the parameters.

Although there are a large number of improved WOA algorithms, most of them have ignored the influence of the distance relationship between search agents on searchability. In these areas of improving the convergence speed, helping the algorithm jump out of the local optimum, and enhancing the search accuracy, WOA still needs further research. In this work, we first try to improve WOA's optimization ability through three new optimization strategies, focusing on improving the original algorithm based on the aggregation potential of the search agent. In the simulation experiment, this article uses the public test function set to verify the effectiveness of the optimization strategies. Finally, the improved algorithm is transformed into a discrete algorithm DLEWOA through several methods to optimize the WSC problem, and the superiority of the improved algorithm is verified through the QWS public data set.

1.1. Standard WOA Algorithm. In WOA, the search agent will optimize through three strategies: encircling prey, bubble-net attacking, and searching for prey. Moreover, WOA will use the coefficient vector A (in Section 1.2) and the random probability P (in Section 1.3) to control the strategy of the next generation. In encircling strategy, the search agent will select the current optimal individual as the target direction; in search for prey, a random search agent will be selected as the target direction; while in bubble-net attacking, the search agents approach the target along shrinking encircling and spiral updating methods simultaneously.

1.2. Encircling Prey. Before encircling prey, the search agent will first select the current optimal candidate solution as the target value. This value is assumed to be the optimal value or close to the optimal value that will be updated with evolutionary iterations. The remaining search agents will approach the target search agent. Equation (1) allows any search agent to update its position for encircling prey. Equation (2) represents the distance between the remaining search agents and the target. The equations are as follows:

$$X(j+1) = X^*(j) - A \cdot D, \tag{1}$$

$$D = \left| C \cdot X^*(j) - X(j) \right|, \tag{2}$$

where j indicates the current iteration, A and C are coefficient vectors, and X^* is the position vector of the best solution obtained so far and should be updated in each iteration if there is a better solution. The equations of A and C are as follows:

$$A = 2a \cdot r_a - a, \tag{3}$$

$$C = 2 \cdot r_c, \tag{4}$$

where r_a and r_c are two random vectors distributed uniformly within [0,1]. a is the convergence factor that is defined as follows:

$$a = 2\left(1 - \frac{j}{T_{\max}}\right). \tag{5}$$

The convergence factor a controls the value of A. Encircling prey and hunting method are chosen when $|A| < 1$, while the searching method is selected when $|A| \geq 1$.

1.3. Bubble-Net Attacking Method. At this phase, the search agent has two simultaneous hunting behaviours: Shrinking encircling and spiraling updating.

Shrinking encircling. Search agents will randomly approach the current best agent

Spiraling updating. Search agents will spirally approach the current best agent

The mathematical model is as follows:

$$D' = \left|X^*(j) - X(j)\right|, \tag{6}$$

$$X(j+1) = D' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(j), \tag{7}$$

where b is a logarithmic helix shape constant and l is a random number in [−1, 1]. Equation (7) represents the distance of the j-th whale to the current best agent. Equation (8) represents the search agent updating position along a spiral-shaped path. Due to a decreased linearly, the position updating range of the search agent will be decreased so as to improve the search accuracy. WOA sets a random number P in [0, 1] to model this simultaneous behaviour. The equation is as follows:

$$X(j+1) = \begin{cases} X^*(j) - A \cdot D, & P < 0.5, \\ X^*(j) + D' \cdot e^{bl} \cdot \cos(2\pi l), & P \geq 0.5. \end{cases} \tag{8}$$

1.4. Search for Prey. This phase emphasizes guaranteeing the WOA algorithm to perform a global search. In this phase, $|A| \geq 1$. A random whale will be chosen from the current group. X_{rand} is the position vector of the random whale. Then, the remaining search agents move towards it, reflecting that the whale will follow the whale group as a whole, which can help the WOA algorithm get rid of the local optimum to a certain extent and perform a global search. The mathematical model is as follows:

$$X(j+1) = X_{rand}(j) - A \cdot D, \tag{9}$$

$$D = \left|C \cdot X_{rand}(j) - X(j)\right|. \tag{10}$$

2. Improved LEWOA Algorithm

The original WOA performs well in the low-dimensional unimodal optimization field. However, the nonlinear optimization process of the original WOA does not match with linear decreasing search step size, which will weaken the ability of global optimization in the exploration stage and reduce the convergence accuracy in the exploitation stage. For WOA, although it has the probability to escape through the search for prey strategy after falling into the local optimum, the algorithm itself does not have a specific method of avoiding the local optimum. This paper presents an improved whale optimization algorithm based on logarithmic convergence factor and aggregation potential energy, aiming to optimize the defects from three aspects.

2.1. Chaotic Map. The searchability of the swarm intelligence algorithm is greatly affected by the initial diversity of swarm conditions. In the WOA algorithm, the initial population state is generated randomly, which is unable to guarantee the swarm diversity in the search space. In this sense, a chaotic strategy can solve the problem well for its ergodicity, high randomness, and regularity. Reference [22] shows the effectiveness of combining a chaotic strategy with WOA. In general, the more homogeneous the chaotic sequence is, the more diverse the swarm states will be; hence, we choose the cubic map with better uniformity performance to initialize the swarm. The cubic map expression is as follows:

$$\begin{cases} y(n+1) = 4y(n)^3 - 3y(n), \\ -1 \leq y(n) \leq 1, n = 0, 1, 2, \ldots, \end{cases} \tag{11}$$

where y_0 cannot be 0 as the initial value of the iteration; otherwise, the chaotic map cannot be established. The steps through the cubic map to initialize the swarm are as follows:

(1) First, a d-dimensional vector coordinate $y_{1d} = (y_{11}, y_{12}, y_{13}, \ldots, y_{1d})$ is randomly generated, substituting it as the first search agent position into equation (13) and iterating it to obtain $n_1 + n_2$ d-dimensional vectors.

(2) Due to the limitations of the chaotic model, the coordinate values obtained through iteration are all between [−1, 1]. And it is necessary to map the chaotic sequence into the search space through the map function. The function is as follows:

$$x_{i\,d} = lb + (1 + y_{id})\frac{(ub - lb)}{2}, \quad i = 1, 2, 3, \ldots, dim, \tag{12}$$

where d represents the space dimension of the solution, ub is the upper limit of the d-th dimension of the limited space, while lb is the lower limit one. y_{id} is the premapping coordinate of the ith search agent in the d dimension obtained according to equation (13), and $x_i d$ is the postmapping agent.

(3) Calculate the fitness value of each coordinate and choose the points with the minimum fitness as the initial position of the search agent.

2.2. The Aggregation Potential Energy. In the process of searching from dispersion to aggregation and finally converging to the optimal point, the state of motion between search agents will affect each other. Hence, their position relationship cannot be simply summarized by the motor pattern of a single search agent. In the field of crowd panic detection [23], the state of the crowd motion can be well illustrated by the increase or decrease of the crowd potential energy [24]. Since there are some similarities in the moving between the swarm and the crowd, this paper introduces aggregation potential energy to describe the aggregation level of the swarm. Aggregation potential energy can well represent the characteristics of individual distribution during the optimization process, so as to better understand the position transformation relationship of search agents in the search space. According to the base definition of potential energy, the larger the aggregation potential energy is, the more scattered the search agents' distribution is. In this state, the algorithm is still in the exploration phase; otherwise, the algorithm is in the exploitation phase or falls into the local optima.

Due to the high sophistication of calculating the distance between each coordinate, this paper defines the aggregation potential energy of swarm as the Euclidean distance between individuals, setting the mean values of coordinates of all search agents in each generation as the population centre. In this way, we replace distance between individuals with Euclidean distance between individuals, and the final population aggregation potential energy is obtained by equation (15). The expression is as follows:

$$E(j) = \phi \frac{\sum_{i=1}^{n} \left| c_i - \sum_{i=1}^{n} c_i / n \right|}{n}, \tag{13}$$

where ϕ is the search range correction factor and is generally 1 here, c_i is the position of the i-th search agent, and n is the total number of search agents.

By comparing the convergence curve with the change of the aggregation potential energy during experiments, it can be shown that the search agents will aggregate to the optimal point with the increase of the iteration. At the same time, the aggregation potential energy will generally decrease and approach 0. For the problem of judging whether the group is conducting the local search or global search, we cannot confirm to a fixed point. In the optimization process, we can only estimate whether the group's motion strategy tends to global or local search, which is the same for aggregation potential energy. The boundary of aggregation potential energy between global and local is an empirical value obtained through a large number of experiments, generally limited to about $10^{-2} \sim 10^{-4}$. And in the face of different search environments, the optimal boundary value of aggregation potential energy needs to be obtained by testing and adjusting.

With the optimization of the aggregation potential energy, the global searchability of the algorithm can be improved. When the algorithm falls into the local optimum, the search agents converge, and the aggregation potential energy decreases. At this time, the chaotic map disturbance being added to the swarm, the algorithm can get rid of the current state by regenerating half search agents. The steps are as follows:

(1) Evaluating whether the algorithm falls into local optimum. Different aggregation potential energy thresholds should be set for different problem models to improve the estimation accuracy. If the aggregation potential energy reaches the set threshold value but the current optimal value does not reach the theoretical optimal value, the algorithm will be judged to fall into the local optimum.

(2) Jumping out of the local optimization through cubic chaotic map. A d-dimensional vector quantity is randomly generated using the cubic chaotic model, iterating it into equation (13) for times. After which particle points are randomly selected and mapped to the search area through equation (14). Then the original half of the search agents are substituted with the worst fitness to enter the iteration again.

The algorithm can be well restarted under the preferable uniformity of the cubic chaotic model by rescattering search agents and replacing them with poor fitness. In the new search process, the search agents that previously fell into the local optimum will reoptimize driven by the new search agents, thus improving the optimization ability of WOA.

2.3. Nonlinear Inertia Weight and Logarithmic Convergence Factor. In WOA, the step sizes of each generation are controlled by the coefficient A. The convergence factor **a** controls the change of A. However, as a linear diminishing factor, **a** leads to the mismatch between the linear diminish of the search step size and the nonlinear convergence of the algorithm. Therefore, this paper converts **a** in logarithmic convergence form, which improves the optimization ability of the algorithm while guaranteeing that the convergence factor **a** decreases. The optimization process can generally be divided into two stages: global optimization and local optimization. The former generally lasts a short time, and the latter lasts a long time. Therefore, the linearly reduced convergence factor cannot match the convergence characteristics of this algorithm. The convergence factor obtained by logarithmic type can quickly reduce the convergence step in the early stage and change the search step in the later stage, so it is appropriate for **a** to be advanced in this new type. The new expression of a is as follows:

$$a = 2 - 2 \cdot \ln\left((e-1) \frac{j}{T_{\max}} + 1 \right). \tag{14}$$

Swarm intelligence algorithm requires a large step size in the exploration phase. While in the exploitation phase, it requires a small one. In the overall search process, it is

adverse for algorithm optimization if the linear variation of step size in the search process does not conform to the actual nonlinear search process. Reference [25] corroborates the effectiveness of the nonlinear adaptive strategy to improve the optimization ability of swarmed-based algorithms. Hence, this paper introduces nonlinear adaptive inertia weight to improve the step size, particularly, combining the aggregation potential energy based on the mathematical model proposed in reference [25]. In the light of the aggregation potential energy, the search state of each search agent can be judged, controlling the algorithm to distribute different inertia weight strategies, aiming to enhance the rate of convergence and the accuracy of convergence. The improved nonlinear inertia weight equation is as follows:

$$w(t)_i = \begin{cases} \gamma \cdot \left(w_1 - \dfrac{(w_2 - w_1)}{T_{\max}} \cdot \dfrac{f(j)_i - f(j)_{\min}}{E(j) \cdot (f(j)_{\max} - f(j)_{\min})} \right), \\ f(j)_i < f(j)_{avg}, E \geq 1 \\ \gamma \cdot \left(w_2 + \dfrac{(w_2 - w_1)}{T_{\max}} \cdot \dfrac{f(j)_i - f(j)_{avg}}{E(j) \cdot (f(j)_{\max} - f(j)_{\min})} \right), \\ f(j)_i \geq f(j)_{avg}, E \geq 1 \\ \gamma \cdot \left(w_1 - \dfrac{(w_2 - w_1)}{T_{\max}} \cdot \dfrac{f(j)_i \cdot E(j)}{f(j)_{\max} - f(j)_{\min}} \right), \\ E < 1 \end{cases} \tag{15}$$

where $w(j)_i$ represents the inertia weight value of the i-th search agent of the j-th generation; w_1 and w_2 are the initial minimum and maximum inertia weight values, respectively; and $T_{\max}$ is the maximum number of iterations. $E(j)$ is the j-th value of the aggregation potential energy. $f(j)_i$ is the i-th fitness of the search agents of the j-th generation, and $f(i)_{avg}$ is the average fitness value of all the search agents of the j-th generation. As for $f(j)_{\max}$ and $f(j)_{\min}$, they are the maximum fitness value and the minimum one of the swarm of the j-th generation. γ is the search range correction factor, which is inversely proportional to ϕ, and their product is 1.

Equation (17) is the inertia weight equation controlled by the aggregation potential energy E. Based on the experiment analysis, we find that when $E \geq 1$, the aggregation potential energy can be considered large with the relatively scattered distribution of search agents. The algorithm is basically in the exploration phase. In this case, if the fitness value of an individual is less than the average fitness value, the search agent is close to the current optimal point, and it should be assigned a smaller inertial weight to approach the optimal point; on the contrary, search agents with poor fitness should be assigned larger inertia weights to enlarge the search step size that is able to improve the global search capability. When $E < 1$, the aggregation potential energy is small, and it is appropriate to search locally with the concentrated distribution of search agents. Each search agent can be distributed with a smaller inertia weight under the protection of jumping-from-local optimal strategy to optimize with high precision around the optimal point, thus ensuring convergence accuracy. The equation of the update of the new position of the search agent is as follows:

$$X(j+1) = \begin{cases} w(t) \cdot X^*(j) - A \cdot D, \\ |A| < 1, P < 0.5 \\ w(t) \cdot X_{rand}(j) - A \cdot D, \\ |A| \geq 1, P < 0.5 \\ w(t) \cdot X^*(j) + D' \cdot e^{bl} \cdot \cos(2\pi l). \\ |A| < 1, P \geq 0.5 \end{cases} \tag{16}$$

2.4. The Flow of the LEWOA. The flowchart of LEWOA is shown in Figure 1, and the detailed steps are described as follows:

Step 1: initialize parameters n, n_1, n_2, d, and $T_{\max}$, set iteration initial value j as 1, and identify optimization targets and search areas.

Step 2: use equation (13) to generate $n_1 + n_2$ d-dimensional vectors, map them into the search area by equation (14), calculate their fitness values, and select the n points with the worst fitness values as the initialization search agents.

Step 3: calculate the fitness value of each search agent to update the current optimal fitness search agent as X^*.

Step 4: update parameters $\boldsymbol{a}$, l, P, $\boldsymbol{A}$, $\boldsymbol{C}$, E, and w.

Step 5: estimate whether the algorithm is trapped in a local optimum by aggregation potential energy E and the current optimal fitness value. If local optimal is entered, regenerate $n/2$ points through step 2, replacing the $n/2$ search agents with the worst current fitness values, and if not, skip this step.

Step 6: update the positions of search agents through equation (18).

Step 7: determine whether the current generation has reached the maximum number of iterations. If not, $j = j + 1$ and turn to step 3. If reached, exit the loop and output the X^*.

3. Web Service Composition Problem Modeling

3.1. QoS-Driven Web Service Composition. WSC problem generally evaluates service capability through QoS. This work divides QoS attributes into positive attributes and negative attributes, including response time, availability, success ability, reliability, service price, throughput rate, credibility, and so on. The establishment of this model selects the first four kinds of attributes for research. In addition, in order to standardize the attribute value of each QoS, the normalization function of QoS is established as follows:

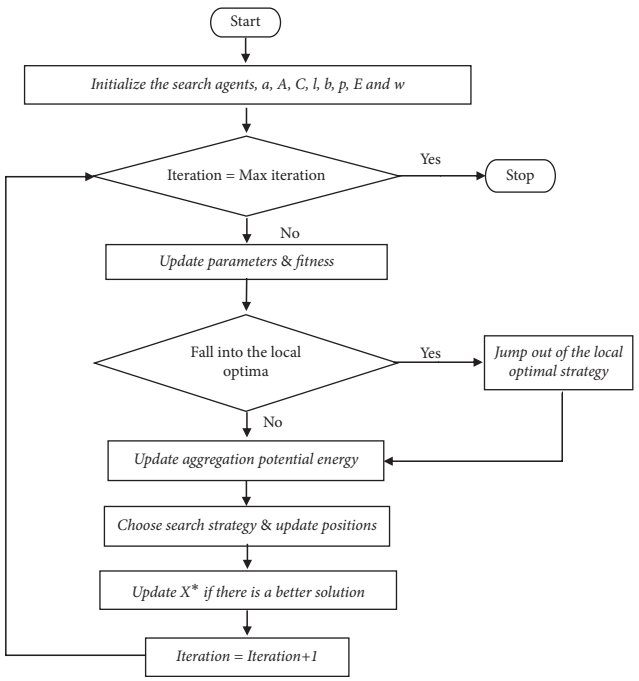


Figure 1: The flowchart shows LEWOA algorithm search strategies.

$$QoS_{i,j}'(S) = \begin{cases} \dfrac{QoS_{i,j}(S) - QoS_{j\_\min}}{QoS_{j\_\max} - QoS_{j\_\min}}, & \text{Positive} - \text{factor}, \\ \dfrac{QoS_{j\_\max} - QoS_{i,j}(S)}{QoS_{j\_\max} - QoS_{j\_\min}}, & \text{Negtive} - \text{factor}, \end{cases} \tag{17}$$

where i is the service number, j is the QoS attribute of the j-th service, $QoS_{i,j}(S)$ is the value of the i-th service and the j-th attribute, and $QoS_{j\_\max}$ and $QoS_{j\_\min}$ are the maximum and minimum values of the j-th QoS attribute, respectively. $QoS'_{i,j}(S)$ is the normalized value.

WSC is often carried out by the following several common workflow patterns: series, concurrent, select, and circulation, as shown in Table 1. Further to say, all of these four patterns can be transformed into series types. Hence, in this work, we choose a series WSC workflow as an instance to complete the experiment, as shown in Figure 2.

3.2. Problem Modeling. The basic idea of applying the LEWOA algorithm to the WSC problem is as Figure 3 shows: in the WSC problem, m specific candidate services are selected from each abstract service class to form a service composition. Therefore, the coordinate dimension of the search agents can be set to m to map the corresponding abstract service class. The coordinate value range of a dimension is mapped to n candidate services available for selection in the corresponding abstract service class. Figure 3 shows the specific mapping process. It should be noted that in practical problems, the number of candidate services in each service class might be different. Therefore, a specific

Table 1: Definition of the aggregate function.

QoS	Series	Concurrent	Select	Circulation
Response time	$\sum_i^n T_i$	$\max T_i$	$\sum_i^n p_i * T_i$	$m * \sum_i^n T_i$
Availability	$\prod_i^n A_i$	$\min A_i$	$\prod_i^n p_i * A_i$	$\prod_i^n A_i$
Success ability	$\prod_i^n S_i$	$\min S_i$	$\prod_i^n p_i * S_i$	$\prod_i^n S_i$
Reliability	$\prod_i^n R_i$	$\prod_i^n R_i$	$\prod_i^n p_i * R_i$	$\prod_i^n R_i$

Note: Q_i is the QoS of the i-th candidate service; n is the number of tasks; and m is the number of cycles.

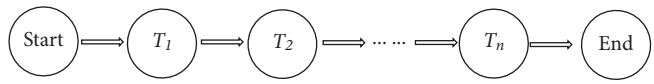


Figure 2: The model of the series structure. Every T_n means one-part specific task of the whole WSC workflow.

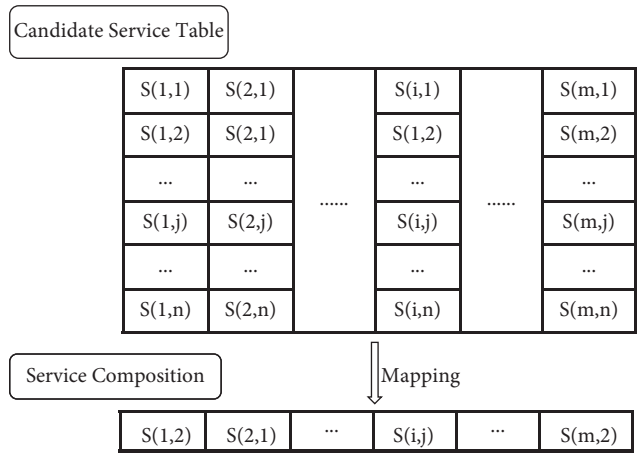


Figure 3: Candidate services table with m columns and n rows, and each grid represents a specific service. The service composition is obtained through the combination of mapping.

search agent represents a specific combination of web services, and its QoS can be calculated through the fitness function. Based on the value of QoS, we can use LEWOA to optimize WSC, and then the appropriate service composition can be obtained.

According to the above selected parameters and workflow patterns, the fitness function can be converted as follows:

$$f(x) = F(\min T(s),\ \min A(s),\ \min S(s),\ \min R(s)), \tag{18}$$

where $T(s)$ is the response time, $A(s)$ is availability, $S(s)$ is success ability, and $R(s)$ is reliability.

3.3. Feasibility Analysis of Improved Strategy. After the QoS-driven WSC problem is converted into a single objective optimization problem, its essence has been transformed into efficiently selecting the optimal composition from a large

number of permutations. For most swarm intelligence algorithms, the search logic can be regarded as taking randomly from the coding candidate set and feeding back according to the obtained value to optimize the next selected generation, which is a logical and efficient random extraction. According to this extraction principle, we can find two ways to improve the optimization: on the one hand, having as many extraction times as possible in limited generations and, on the other hand, ensuring that the extracted value can better optimize the extraction strategy of the next generation.

Based on the first optimization principle, considering the time and complexity of the WSC problem, it is hard for the swarm intelligence algorithm to guarantee to search the optimal value of the web service composition. Therefore, only when the algorithm keeps searching within limited generations will it not cause search waste. First, we need to avoid the algorithm falling into the local optimum. Second, the algorithm needs to ensure the diversity of the search population, so as to obtain the most information in each generation of extraction. For the improved algorithm, first, the cubic chaotic mapping principle is adopted to ensure the diversity of initial search agents; second, the aggregation situation is estimated by the aggregation potential energy of the search agent. As long as the algorithm enters the local search stage, the split mutation strategy is adopted to regenerate a new part of the population to ensure the swarm diversity of the next generation.

The second optimization principle needs a swarm intelligence algorithm to efficiently feedback the search information obtained by the previous generation and choose the suitable search strategy. For example, in applying the ant colony algorithm to web service composition, pheromones are efficiently used to transmit the search information of the previous generation. However, in the original WOA algorithm, no such parameter can better feedback information to the next generation. In the face of the solution of most continuity problems, WOA's search logic will not have a great defect. However, it is difficult to have good optimization performance when encountering complex high-dimensional problem models and discrete problems. The new parameter of aggregation potential energy in LEWOA proposed in this paper can be better used as a medium to transmit the search information between each generation. In the aspect of search strategy, the search strategy of the next generation is determined according to the aggregation potential energy of each generation. In an aspect of step size, the improved algorithm changes the linear step size and convergence factor to nonlinear variation, which is to match the actual nonlinear search process.

3.4. Encoding Rules for Mapping. Since the coordinate values of each dimension of the search agent in the LEWOA are continuous, the serial number of the specific candidate service is discrete. Therefore, the fuzzy function f_d should be used to convert the coordinate values of the search agent into corresponding integer code to form the discrete logarithmic energy whale optimization algorithm (DLEWOA). The fuzzy function f_d is as follows:

$$f\ d\left(x_{i,d}^{t}\right) = \begin{cases} z,\ (z-0.5) < x_{i,d}^{t} < (z+0.5) \\ iff\ (Y=0, z, z+1), x_{i,d}^{t} = (z+0.5) \\ m_d, x_{i,d}^{t} \in [0, 0.5) \bigcup (m_d - 0.5, m_d] \\ iff\ (Y=0, m_d, 1), x_{i,d}^{t} = 0.5 \end{cases}, \quad (19)$$

where $x_{i,d}^{t}$ is the coordinate value of the t generation of the search agent i in the d dimension, m_d is the number of candidate services of the abstract service class corresponding to this dimension, and z is the integer on [1, m_d]. The random variable Y is the result of a Bernoulli test with a probability of 0.5. The value of the function *iff*(P, u, v) depends on whether the proposition P is true. If true, it is u; else, it is v.

4. Simulation Experiment and Analysis

4.1. Parameters Setting and Benchmark Functions. To evaluate the optimization ability of the LEWOA, several simulation experiments are conducted based on eight benchmark functions (shown in Table 2), whose experiment results are compared to other swarm intelligence algorithms. The experiment parameters are set as follows: the number of the search agents is 30, the average experimental simulation is 30 times, and the maximum number of iterations is 1,500. As for MWOA, the inertia weight is set according to reference [25], whose minimum value w_1 equals 0.01 and the maximum one w_2 equals 0.4. For the reason that the original fitness parameters are weighted by the aggregation potential energy of the population, after a multitude of experiments, the parameter of the inertia weight is $w_1 = 0.01$, and the maximum weight parameter is $w_2 = 0.08$. The simulation experiment in this section is conducted in the conditions of Intel Core, CPU i7-7700HQ, 2.80 GHz, 8 GB, and MATLAB 2016a. Table 3 manifests the specific algorithmic parameters.

Table 2 is the model of eight benchmark functions. F1, F2, F3, and F4 belong to unimodal function mainly utilized to test the convergence rate and accuracy of the algorithm; F5 and F6 are multimodal functions mainly wielded to test the ability of global search of the algorithm; and F7 and F8 belong to the multimodal function of mixed dimensions, which are applied to test the impact of search dimension on the searchability of the algorithm.

4.2. Experimental Results and Analysis. In this section, a comprehensive analysis of the LEWOA will be presented. The experiment records the fitness values of the best search agent and calculates the mean value and the standard deviation to evaluate the accuracy and stability of algorithm optimization results. It analyses the number of convergence and depicts the variation curves of the fitness values of the best search agent and convergence generations to evaluate the algorithm's convergence rate. The results are shown in Table 4, in which the black-labeled represents the optimal data.

Table 2: Eight classical benchmark test functions.

Function	Dimension	Range	Optimal value
$F_1(x) = \sum_{i=1}^{n} x_i^2$	30	[−100,100]	0
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	[−10,10]	0
$F_3(x) = \sum_{i=1}^{n} (\sum_{j=1}^{i} x_j)^2$	30	[−100,100]	0
$F_4(x) = \max_i \{\lvert x_i \rvert, 1 \le i \le n\}$	30	[−100,100]	0
$F_5(x) = \sum_{i=1}^{n} i x_i^4 + \text{random}[0,1)$	30	[−1.28,1.28]	0
$F_6(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	30	[−5.12,5.12]	0
$F_7(x) = (\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} (x_i - a_{ij})^6})^{-1}$	2	[−65,65]	1
$F_8(x) = \sum_{i=1}^{2} [a_i - \frac{x_1 (b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}]^2$	4	[−5,5]	3.00×10^{-4}

Table 3: Algorithm parameter setting.

Algorithm	Parameter	Value
MFO [26]	Convergence factor [a]	Linearly decreases from −1 to −2
ABC [27]	Step size [a]	1
	Fraction [P_a]	0.25
PSO [28]	Acceleration [C_1, C_2]	2
	Inertia weight [w]	0.8
	$V_{\max}$	0.05
GSA [29]	Acceleration [A]	20
	Power	1
	Gravitational [G_0]	100
WOA [11]	Convergence factor [a]	Linearly decreases from 2 to 0
MWOA [25]	Convergence factor [a]	Nonlinearly decreases from 2 to 0
LEWOA	Convergence factor [a]	Nonlinearly decreases from 2 to 0

Table 4: Performance comparisons of three algorithms on eight benchmark functions.

Function	Evaluation	WOA	MWOA	LEWOA
F1	Ave	$3.60e-277$	0	**0**
	Std	0	0	**0**
	Generations	1,500	796	**620**
F2	Ave	$9.36e-156$	0	**0**
	Std	$4.45e-155$	0	**0**
	Generations	1,500	902	**740**
F3	Ave	11022.94	0	**0**
	Std	8145.24	0	**0**
	Generations	1,500	807	**604**
F4	Ave	36.16	0	**0**
	Std	29.77	0	**0**
	Generations	1,500	887	**739**
F5	Ave	$7.04e-4$	$\mathbf{1.94e-5}$	$3.66e-5$
	Std	$8.22e-4$	$6.22e-5$	$\mathbf{4.04e-5}$
	Generations	1,500	1,500	1,500
F6	Ave	0	0	**0**
	Std	0	0	**0**
	Generations	422	37	**19**
F7	Ave	1.72	1.69	**1.65**
	Std	2.95	0.933	**0.819**
	Generations	1,500	1,500	1,500
F8	Ave	$5.52e-4$	$3.59e-4$	$\mathbf{3.53e-4}$
	Std	$2.28e-4$	$\mathbf{4.99e-5}$	$5.56e-5$
	Generations	1,500	1,500	1,500

According to the test results from F1 to F6 in Table 4, LEWOA has surpassed WOA from the perspectives of each comparison. This result shows that LEWOA is obviously superior to WOA in the accuracy, stability, and rate of optimization based on the six benchmark functions. From F1 to F4 and in F6, LEWOA and MWOA all reach the target values, and the former convergence rate is better since the convergence number of iterations of LEWOA is evidently less than those of MWOA, which is more apparent in the unimodal function test environment. In terms of multimodal functions from F5 to F8, except for F8 in which the standard deviation of LEWOA is slightly inferior to those of MWOA, there are no disparities on account of the order of magnitude. For other benchmark functions, the accuracy and stability of LEWOA's convergence are better than those of MWOA, and the convergence rate is also preponderant. In the case of F8, considering that the increase of the search dimensions has the corresponding influence on the effect of optimization, it can be concluded that the test results in the mean value and standard deviation of the optimal fitness of LEWOA are much better than those of WOA, and it also has better accuracy and stability of convergence for MWOA.

To more intuitively reflect the algorithms' convergence in the benchmark function, a convergence curve is drawn due to one experiment that is the closest to the 30 times of the average test results, comparing the characteristics of convergence of WOA, MWOA, and LEWOA. Figure 4 relatively corresponds to functions F1 to F8, in which the bold font is the best result.

From the convergence curve of Figure 4, LEWOA has a qualitative improvement over WOA in search efficiency and local search accuracy. Especially in F3 and F4, it can be seen that compared with the improved algorithms, WOA proves to fall into the local optimum easily when processing complex problems, while the improved algorithms using the nonlinear strategy still have better convergence. For F1, F2, and F6, the WOA appears very weak in the later local search phase due to its linear allocation step size. On the contrary, the convergence rate of the improved algorithms does not decrease at the later stage because the later algorithms can still allocate an appropriate search step. Compared with the MWOA algorithm, LEWOA is better in convergence generations while maintaining astringency because, under the control of aggregation potential energy, LEWOA can more efficiently allocate optimal search strategies for search agents at each generation.

In theory, the optimization ability of the swarm intelligence algorithm increases exponentially with the increase of the swarm population. Therefore, in order to demonstrate the optimization ability of LEWOA, Table 5 discusses the influence of the ability of optimization of LEWOA on the population size, in which the black-labeled represents the optimal data. The swarm number in Table 5 takes into account the nonlinear growth optimization ability of the swarm intelligence algorithm. From unimodal test functions $f_1 \sim f_4$, we can find that the change of population has little impact on the excellent optimization ability of LEWOA. In general, small populations have more efficient optimization speed in such a simple search environment. In the multimodal function of f_4–f_8, although the large population has relatively good optimization results, careful observation of the data shows that the small population's optimal value and standard deviation have only a decline of one order of magnitude at most. The LEWOA algorithm for the small population, even in a complex environment still maintains good convergence accuracy and stability. Compared with PSO, GSA, ABC, and other algorithms that need a large number of populations to maintain their astringency, the LEWOA algorithm still occupies a dominant position in the field of small population search.

Table 6 compares data from LEWOA and the current swarm intelligence algorithms, selecting MFO, PSO, ABC, GSA, and WOA for performance comparison. To ensure the objectivity and accuracy of the experimental data, each algorithm runs 30 times independently, and the test functions are f_1~f_8. The average value of optimal solutions of six algorithms and the standard deviation of 30 times of independent operation are shown in Table 5, in which the bold font is the best result.

Concluded from the test data of 8 benchmark functions f_1~f_8 in Table 6, the optimal data of LEWOA are not acquired merely in f_7, which embodies the superiority of optimization of the improved algorithm. For the high-dimensional unimodal function f_1~f_4, LEWOA is able to converge to the optimal value every time under the 1,500 generations, which other comparison algorithms cannot achieve. The excellent optimization results not only show that LEWOA has higher precision local convergence ability in unimodal problems but also has better algorithm stability. The improvement of the local convergence ability of LEWOA is due to the improved inertia weight and nonlinear convergence factor, which enables the algorithm to allocate appropriate search steps to deal with the complex search environment. In higher dimensional multimodal functions f_5~f_6, LEWOA excels better in the accuracy of optimization and stability than other current algorithms other than WOA, which indicates WOA itself possesses a good global optimization ability of global optimization. According to the results of f_5, LEWOA is better in the accuracy and stability of convergence than those of WOA. It proves the progress of the ability of global optimization of the improved algorithm in higher dimensional multimodal problems and the effectiveness of the combination of aggregation potential energy with nonlinear inertia weight strategy to improve the ability of optimization. For mixed low dimensional multimodal functions f_7 and f_8, the optimal solution and standard deviation of LEWOA are only slightly smaller than that of the ABC algorithm in f_7. However, from f_7 to f_8, with the increase of function dimension, it exactly turns out LEWOA performs better than the ABC algorithm both in convergence accuracy and stability. This change shows that with the help of aggregation potential energy, LEWOA can better jump out of local optimal and conduct global optimization, which proves its astringency and dominance in complex problems.

4.3. Experimental Simulation of Web Service. In this section, the LEWOA is applied to the QoS-driven WSC problem and compared the optimization performance in the QWS data

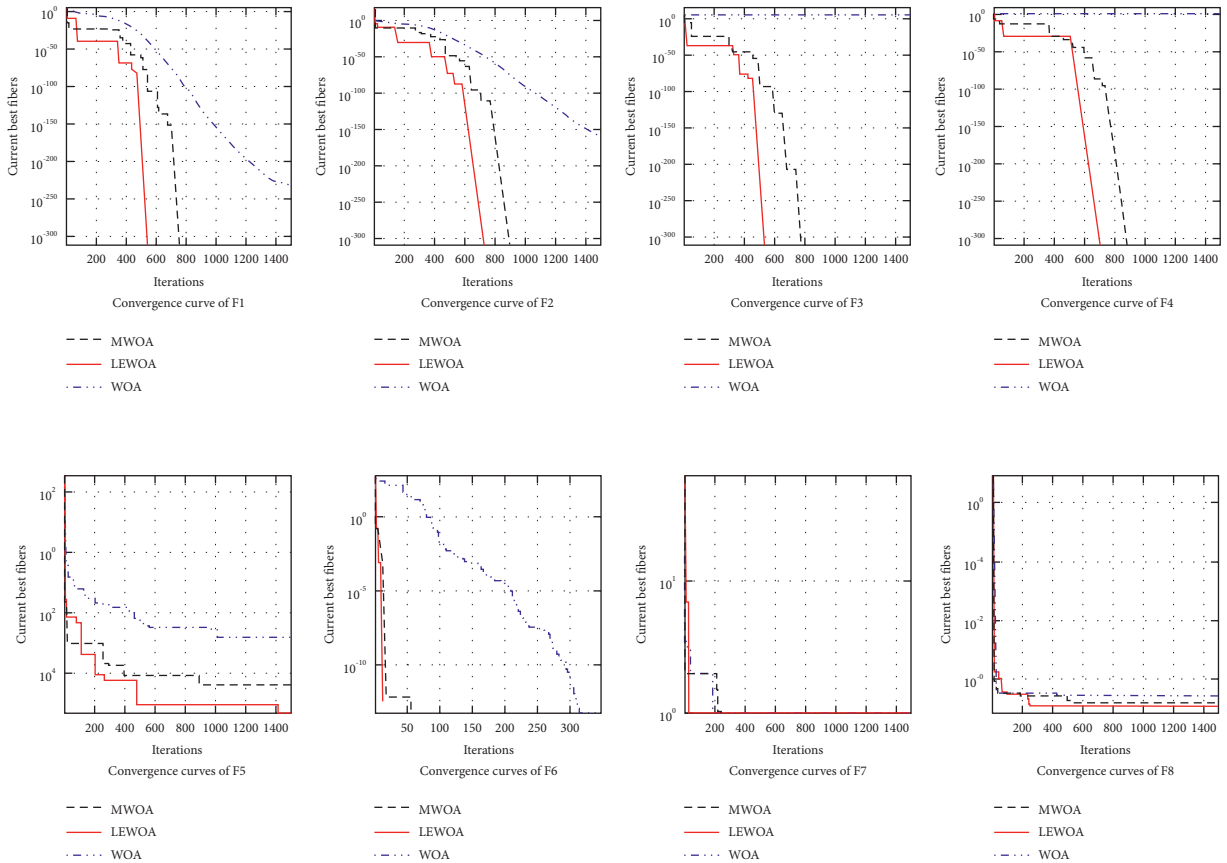


Figure 4: Convergence curves of 8 benchmark functions. The plot of current best fitness with respect to iteration recorded for 1,500 iterations. Display 3 different algorithms convergence curves.

Table 5: Performance comparisons of LEWOA on eight benchmark functions different population size.

Function	Evaluation	The number of search agents			
		10	20	25	30
F1	Ave	0	0	0	0
	Std	0	0	0	0
	Generations	356	**345**	348	352
F2	Ave	0	0	0	0
	Std	0	0	0	0
	Generations	465	463	**456**	459
F3	Ave	0	0	0	0
	Std	0	0	0	0
	Generations	**340**	359	355	348
F4	Ave	0	0	0	0
	Std	0	0	0	0
	Generations	467	**451**	459	461
F5	Ave	$1.14\,e-4$	$3.98\,e-5$	$3.75\,e-5$	$\mathbf{3.59\,e-5}$
	Std	$1.16\,e-4$	$4.82\,e-5$	$3.57\,e-5$	$\mathbf{3.19\,e-5}$
	Generations	1,500	1,500	1,500	1,500
F6	Ave	0	0	0	0
	Std	0	0	0	0
	Generations	18	16	16	**14**
F7	Ave	3.51	2.57	1.35	**1.34**
	Std	3.25	2.93	0.74	**0.67**
	Generations	1,500	1,500	1,500	1,500
F8	Ave	$4.06\,e-4$	$3.56\,e-4$	$3.58\,e-4$	$\mathbf{3.46\,e-4}$
	Std	$1.14\,e-4$	$4.18\,e-5$	$4.93\,e-5$	$\mathbf{3.17\,e-5}$
	Generations	1,500	1,500	1,500	1,500

Table 6: Performance comparisons of six algorithms on eight benchmark functions.

Function	Evaluation	LEWOA	WOA	MFO	PSO	ABC	GSA
F1	Ave	**0**	3.1739 $e-230$	3.3126 $e-47$	0.0505	0.0010	7.593 $e-17$
	Std	**0**	**0**	1.0324 $e-46$	0.0204	9.6424 $e-4$	3.0781 $e-17$
F2	Ave	**0**	2.4810 $e-156$	1.6667	0.5709	7.0518	4.4585 $e-8$
	Std	**0**	1.3409 $e-155$	4.6113	0.7609	18.9981	9.0015 $e-9$
F3	Ave	**0**	1.928 $e+4$	500.000	9.0176	5.9959 $e+4$	258.5095
	Std	**0**	6.9793 $e+3$	1.52556 $e+3$	3.4258	8.8142 $e+3$	127.6924
F4	Ave	**0**	28.7110	0.8791	2.1892	59.2963	0.0102
	Std	**0**	31.5702	1.7280	0.8402	4.8688	0.0504
F5	Ave	**3.1616 $e-15$**	9.7329 $e-4$	0.0063	0.0133	0.4046	0.0453
	Std	**3.2410 $e-5$**	0.0010	0.0053	0.0054	0.1437	0.0138
F6	Ave	**0**	**0**	24.1561	27.4233	229.7770	26.1342
	Std	**0**	**0**	16.0788	10.5341	15.4841	15.9193
F7	Ave	1.5272	1.7813	2.7264	2.7730	**0.9980**	2.8253
	Std	0.8536	2.4926	3.5507	2.4099	**2.7390 $e-5$**	1.9226
F8	Ave	**3.5318 $e-4$**	5.2018 $e-4$	0.0022	4.2767 $e-4$	0.0012	0.0024
	Std	**5.5550 $e-5$**	2.7276 $e-4$	0.0050	3.586 $e-4$	1.5012 $e-4$	0.0014

Table 7: Performance comparisons of three algorithms on QWS.

Algorithm	Ave	Worse	Best	Std
DLEWOA	**1.4664**	**1.5121**	**1.4034**	**0.0293**
DWOA	1.6223	1.7296	1.5447	0.0492
DMWOA	1.6311	1.7080	1.5602	0.0471
DPSO	1.6530	1.7085	1.4295	0.0633
DMFO	1.7607	1.8020	1.6120	0.0579

Table 8: Difference level analysis relative to DLEWOA.

Algorithm	P-value	α	Confidence interval
DLEWOA	—	0.05	[0, 0]
DWOA	2.4457 $e-17$	0.05	[0.1353, 0.1560]
DMWOA	5.5559 $e-20$	0.05	[0.1657, 0.1836]
DPSO	1.5275 $e-14$	0.05	[0.1687, 0.2054]
DMFO	1.6085 $e-19$	0.05	[0.2784, 0.3103]

set with the DWOA algorithm and DPSO. The experiment sets 10 service classes, each service class contains 250 kinds of services and constructs a web service composition model in series. Four QoS evaluation criteria in QWS are selected, which are response time, availability, success ability, and reliability. They are brought into equation (17) for normalization. In addition to availability, the other three are treated as positive factors, and the weights of the four are set as 0.2, 0.3, 0.2, and 0.3.

The parameter settings of the following comparison algorithms mentioned in Table 7 are the same as those in Table 3. In order to adapt to the discrete search process of WSC problem, the coordinates of each algorithm are integer processed by fuzzy function equation (19).

Since, in the WSC problem of this work, the search scope is expanded to two orders of magnitudes, in order to ensure the search efficiency, ϕ and γ will be changed to 0.01 and 100, respectively. Besides, the optimization ability of the WSC problem is guaranteed by the global search as mentioned in Section 3.3. Therefore, in order to ensure the diversity of the population and enhance the global searchability of the DLEWOA, the aggregation potential energy threshold is set to 1, and 20 independent experiments are conducted under the same conditions. The other experimental parameters are set the same as before. The experimental results are as follows, among which bold font is the best:

To further illustrate DLEWOA, we use the P-value obtained by t-test to test whether the result values of each algorithm belong to the same distribution to prove the uniqueness of the LEWOA algorithm. P-value is a probability of observed samples and more extreme cases on the premise that the original hypothesis is true. We initially assumed that the values obtained by other algorithms belong to the same distribution as those obtained by LEWOA. Therefore, the smaller the P-value obtained, the more rejected the original hypothesis. Generally speaking, when the value of P-value is less than 0.001, it can be considered that there is a significant difference. In Table 8, where α is the

significance level and set to 0.05 and then the confidence level is 95%. In this article, we use the t-test function from MATLAB to obtain P value and confidence interval.

It can be seen from Table 7 that compared with the other algorithms, the average optimization result and the optimal value of the DLEWOA are much better, and it also has the lowest standard deviation, which shows that the DLEWOA algorithm in the WSC problem guarantees both validity and stability. Obviously, based on the data of Table 8, no matter which algorithm DLEWOA is compared with, the order of magnitude of the p-value obtained is small enough to verify its uniqueness.

At the same time, this paper tries to conduct a comparative experiment on experimental data through a pseudorandom traversal algorithm in the same experimental environment. When the optimization result reaches about 1.50, the pseudorandom algorithm needs to iterate around 9,000 generations, which takes several times as the swarm intelligence algorithm. Although the pseudorandom traversal algorithm can get better optimization results with enough iterations, the swarm intelligence algorithm is obviously a better choice when time cost is considered.

5. Conclusions

This paper mainly advances WOA into LEWOA combining three strategies aimed at its defects in processing some multimodal functions and studies the application of DLEWOA in the QoS-driven WSC problem. The DLEWOA is proposed by using integer coding with the fuzzy function, which solves the problems of mismatch between continuity algorithm and discrete problem model. In the first analysis of LEWOA with eight test functions, the improved algorithm demonstrates its strengths in the convergence rate, optimization ability, and convergence accuracy. In the meantime, this paper tests the impact of swarm's quantity on algorithms, showing that the improved algorithm can still ensure a higher convergence rate and search accuracy in the small population. In the second experiment of QoS-driven WSC, this paper tests the DLEWOA through the QWS data set, and the experiment proves the superiority of the improved algorithm in the comprehensive performance of the WSC optimization problem. As a result, the above experiments validate the effectiveness and superiority of the improved algorithm: LEWOA.

Data Availability

The optimization functions used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this study.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant nos. 61906055, 61771418, 61872105, and 62072136 and the National Key R&D Program of China under Grant no. 2020YFB1710200.

References

[1] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–599, 2018.

[2] T. Liu, Y. Wang, Y. Li, X. Tong, L. Qi, and N. Jiang, "Privacy protection based on stream cipher for spatiotemporal data in IoT," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 7928–7940, 2020.

[3] A. Alnoman, S. K. Sharma, W. Ejaz, and A. Anpalagan, "Emerging edge computing technologies for distributed IoT systems," *IEEE Network*, vol. 33, no. 6, pp. 140–147, 2019.

[4] Y. Song and Y. Z. Gong, "Web service composition on IoT reliability test based on cross entropy," *Computational Intelligence*, vol. 36, no. 4, pp. 1650–1662, 2020.

[5] A. Strunk, "QoS-aware service composition: a survey," in *Proceedings of the Eighth IEEE European Conference on Web Services IEEE*, pp. 67–74, Ayia Napa, Cyprus, December 2010.

[6] O. Zedadra, A. Guerrieri, N. Jouandeau, G. Spezzano, H. Seridi, and G. Fortino, "Swarm intelligence-based algorithms within IoT-based systems: a review," *Journal of Parallel and Distributed Computing*, vol. 122, pp. 173–187, 2018.

[7] L. Huang, X. Zhang, Y. Huang, G. Wang, and R. Wang, "A QoS optimization for intelligent and dynamic web service composition based on improved PSO Algorithm," in *Proceedings of the 2011 Second International Conference on Networking and Distributed Computing*, pp. 214–217, Beijing, China, September 2011.

[8] X. Zhao, B. Song, P. Huang, Z. Wen, J. Weng, and Y. Fan, "An improved discrete immune optimization algorithm based on PSO for QoS-driven web service composition," *Applied Soft Computing*, vol. 12, no. 8, pp. 2208–2216, 2012.

[9] F. Dahan, K. E. Hindi, A. Ghoneim, and H. Alsalman, "An enhanced ant colony optimization based algorithm to solve QoS-aware web service composition," *IEEE Access*, vol. 9, pp. 34098–34111, 2021.

[10] M. Chen, Q. Wang, W. Sun, X. Song, and N. Chu, "GA for QoS satisfaction degree optimal web service composition selection model," in *Proceedings of the 2019 6th International Conference on Behavioral, Economic and Socio-Cultural Computing (BESC)*, Beijing, China, October 2019.

[11] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.

[12] W. A. Watkins and W. E. Schevill, "Aerial observation of feeding behavior in four baleen whales: Eubalaena glacialis, balaenoptera borealis, megaptera novaeangliae, and balaenoptera physalus," *Journal of Mammalogy*, vol. 60, no. 1, pp. 155–163, 1978.

[13] F. S. Gharehchopogh and H. Gholizadeh, "A comprehensive survey: whale optimization Algorithm and its applications," *Swarm and Evolutionary Computation*, vol. 48, pp. 1–24, 2019.

[14] D. Chu, H. Chen, and X. Wang, "Whale optimization algorithm based on adaptive weighting and simulated annealing," *Acta Electronica Sinica*, vol. 5, no. 5, pp. 992–999, 2019.

[15] K. Gaganpreet and A. Sankalap, "Chaotic whale optimization algorithm," *Journal of Computational Design and Engineering*, vol. 5, pp. 275–284, 2018.

[16] D. Oliva, M. Abd El Aziz, and A. Ella Hassanien, "Parameter estimation of photovoltaic cells using an improved chaotic

whale optimization algorithm," *Applied Energy*, vol. 200, pp. 141–154, 2017.

[17] A. N. Jadhav and N. Gomathi, "WGC: hybridization of exponential grey wolf optimizer with whale optimization for data clustering," *Alexandria Engineering Journal*, vol. 57, no. 3, pp. 1569–1584, 2018.

[18] I. N. Trivedi, P. Jangir, A. Kumar, N. Jangir, and R. Totlani, "A novel hybrid PSO-WOA algorithm for global numerical functions optimization," *Advances in Computer and Computational Sciences*, vol. 554, pp. 53–60, 2018.

[19] Virupakshappa and B. Amarapur, "Computer-aided diagnosis applied to MRI images of brain tumor using cognition based modified level set and optimized ANN classifier," *Multimedia Tools and Applications*, vol. 79, no. 5-6, pp. 3571–3599, 2018.

[20] I. Aljarah, H. Faris, and S. Mirjalili, "Optimizing connection weights in neural networks using the whale optimization algorithm," *Soft Computing*, vol. 22, no. 1, pp. 1–15, 2018.

[21] A. M. Al-Zoubi, H. Faris, J. f. Alqatawna, and M. A. Hassonah, "Evolving Support Vector Machines using Whale Optimization Algorithm for spam profiles detection on online social networks in different lingual contexts," *Knowledge-Based Systems*, vol. 153, pp. 91–104, 2018.

[22] Z. Xu, Y. Yu, H. Yachi, J. Ji, Y. Todo, and S. Gao, "A novel memetic whale optimization algorithm for optimization," *Advances in Swarm Intelligence. Lecture Notes in Computer Science*, vol. 10941, pp. 384–396, 2018.

[23] X. Zhang, Q. Yu, and H. Yu, "Physics inspired methods for crowd video surveillance and analysis: a survey," *IEEE Access*, vol. 6, pp. 66816–66830, 2018.

[24] G. Xiong, J. Cheng, X. Wu, Y.-L. Chen, Y. Ou, and Y. Xu, "An energy model approach to people counting for abnormal crowd behavior detection," *Neurocomputing*, vol. 83, pp. 121–135, 2012.

[25] Y. Zhang and F. Chen, "A modified whale optimization algorithm," *Computer Engineering*, vol. 44, no. 3, pp. 208–213, 2018.

[26] S. Mirjalili, "Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm," *Knowledge-Based Systems*, vol. 89, pp. 228–249, 2015.

[27] D. Karaboga and C. Ozturk, "A novel clustering approach: artificial Bee Colony (ABC) algorithm," *Applied Soft Computing*, vol. 11, no. 1, pp. 652–657, 2011.

[28] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the ICNN'95—International Conference on Neural Networks*, pp. 1942–1948, Perth, WA, Australia, 1995.

[29] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 5457044, 12 pages
https://doi.org/10.1155/2022/5457044

Research Article

A Hierarchical Network with User Memory Matrix for Long Sequence Recommendation

Jiawei Dong, Fuzhen Sun, Tianhui Wu, Xiangshuai Wu, Wenlong Zhang, and Shaoqing Wang

School of Computer Science and Technology, Shandong University of Technology, Shandong, Zibo 255000, China

Correspondence should be addressed to Fuzhen Sun; sunfuzhen@sdut.edu.cn

Received 26 September 2021; Accepted 30 December 2021; Published 31 January 2022

Academic Editor: Yingjie Wang

In many recommendation scenarios, the interactions between users and items are divided into a series of sessions according to the time interval. The traditional Recurrent Neural Network has some shortcomings, such as limited memory ability, inflexible access to memory data, and obvious deficiency in feature capture for long sequences. To deal with the mentioned issues, we propose a hierarchical network with user memory matrix, named HNUM^2, which utilizes the memory network to store users' long-term and short-term interests. The memory network is more flexible to access memory data, which can solve the problem of insufficient capture of long sequence features. The proposed model is a hierarchical recommendation algorithm, which consists of two layers. The first layer is the session-level GRU model, which obtains the sequence characteristics of the current session to predict the next item. The second layer is the user-level memory network model which exploits the attention mechanism and incorporates the write module and read module. The experimental results on two public available datasets show that HNUM^2 has achieved significant performance improvement comparing to the state-of-the-art methods.

1. Introduction

With the development of the big data era, recommender systems are still an effective means to solve information overload [1]. Sequential Recommender Systems (SRSs) have received more and more attention in recent years. Through the interaction between users and items, SRSs understand and generate user behavior sequences while capturing changes in users' interests [2]. Session-based recommender systems (SBRs) are a branch of sequential recommender systems, which received considerable attention from industry and academia [3]. Deep learning technology has set off an upsurge in academia and industry. More and more scholars have applied deep learning technology to recommender systems [4]. Deep learning models have powerful learning ability and can avoid the problem of traditional recommendation models, such as the manual design model features [5]. Yap et al. [6] introduced a recommendation framework based on personalized sequential pattern mining, which used a new score metric to effectively learn user-specific sequences important for accurate personalized recommendations. In recent years, similarity-based methods have been applied to session-based recommendations, with good results on sparse datasets. Hidasi et al. [7] first applied Recurrent Neural Network to recommender systems, which designed a parallel session recommendation model GRU4REC. Experimental results showed that Recurrent Neural Network has a good performance in session-based recommendation algorithms. Quadrana et al. [8] proposed a hierarchical recommendation model. The model designed two levels of RNN: the user-level RNN model and the session-level RNN model.

RNN has a relatively good performance in the sequential tasks. It can store limited information and more content as the memory units. However, it loses more information [9]. In 2014, Weston et al. [10] introduced a new learning model, memory network. In the same year, the DeepMind team of Google proposed neural turing machines [11]. Both of them use external memory for memorization. The neural turing machine was designed with attention-based read and write

operations that allow for more flexible reading of memories. In 2015, Sukhbaatar et al. proposed end-to-end memory network [12], where external storage space of the network is a memory matrix, which is introduced to better capture long sequence features.

Memory network was initially used in Q&A systems. Recently, memory network has been widely used in the recommender systems, which has attracted people's attention. Chen et al. [13] stored and updated user's history by using external storage matrix in memory network and enhanced the expressiveness of the model. Huang et al. [14] obtained two benefits from the hybrid module by using a mixture of RNN and key-value memory networks (KV-MNS). A combination of sequential preference representation and attribute-level preference representation is used as the final representation of user preferences. Due to the addition of knowledge-based information, the model is highly interpretable [15]. To take full advantage of textual information and visual information, Ma et al. [16] proposed new cross-attention memory network to perform multimodal tweet reference recommendation, which combined users' interests with external memory and uses cross-attention mechanism to extract textual information and visual information [17].

Based on the above problems, the contributions of this paper are essentially threefold:

(1) According to previous work, sessions are assumed to be independent of each other, and historical session information is ignored. To solve the above problem, we propose a hierarchical network with user memory matrix (HNUM^2), which considers the interaction between sessions and historical session information to read the user's historical sessions and provides initial input to the GRU unit within the session.

(2) We proposed a hierarchical recommendation model. The first layer is a session-level GRU model for predicting the next item. The second layer is the user-level memory network model, which refers to the changes in users' long-term interests. The model consists of two modules: the read module and the write module.

(3) The experimental results show that the proposed model has better performance improvement than the current algorithm when the number of user sessions is large.

The rest of the paper is organized as follows: In Section 2, we briefly review the existing research on session-based recommender systems and Recurrent Neural Network. In Section 3, we first present the whole structure of the model, then introduce the two levels of the model and the loss function, and finally give the algorithm flow of the model. Section 4 describes and analyzes these assessments, and a large number of experiments on two real datasets of different volumes demonstrate the recommender performance of the proposed model compared to other models. In Section 5, we summarize our work and propose several future research directions.

2. The Related Work

We first review current models of session-based recommender systems and then introduce Recurrent Neural Network (RNN) and GRU. Finally, we review the latest research on memory network.

2.1. Session-Based Recommendation Algorithm. Session sequences refer to a set of item sequences used by a user in an interactive transaction or collected over a period of time [18]. Traditional recommendation algorithms only model user's long-term preferences and static preferences and ignore short-term and dynamic transaction patterns of users, which can lead to missing the transfer of user preferences over time. In this case, a user's intention at a past time can easily be replaced by a new user's historical behavior, resulting in poor and unreliable recommendations. In order to solve the above problems, it is necessary to consider the affair structure to capture richer information in the recommendation [19]. Therefore, session-based recommender systems are proposed.

Session-based recommendation problem can be expressed as sequence prediction problems; we define a session $\{x_1, x_2, \ldots, x_{s-1}, x_s\}\{x_1, x_2, \ldots, x_{s-1}, x_s\}$, where $x_i\,(1 \le i \le S)$ denotes the index of the user's interactive items in the total number of N items. Define the output as the sort list $y = [y_1, y_2, \ldots, y_n] \in R^N$ of all possible items in the session, where y_i corresponds to the score of item i. The usual practice is to sort according to the size of y_i, taking the top-K items.

Aghdam et al. [20] introduced a hierarchical hidden Markov model to capture changes in user preferences, used the feedback sequence of users to items, modeled users as a hierarchical hidden Markov process, and used users' current content attributes as hidden variables in this model. Gu et al. [21] proposed using Markov chains to track user purchase behavior chains, using purchase intervals to improve the temporal diversity of e-commerce recommendations. The algorithm has a significant improvement in accuracy, conversion rate, and time diversity. He et al. [22] proposed Mixture Variable Memory Markov (MVMM) model, which is a new method of sequential query prediction. This method attempts to capture the user's search intent based on the user's past query sequences. Markov model only considers relatively short historical information, and its representation ability is minimal [23].

2.2. Recurrent Neural Network. Recurrent Neural Network (RNN) is a kind of neural network specifically designed for sequential data. By receiving its own information, RNN achieves a certain "memory function" and retains a certain amount of memory for the processed information [24]. Given an input sequence $\{x_1, x_2, \ldots, x_t, \ldots, x_T\}$ of length T, x_t represents the input vector of the sequence data at the moment t. The index t is not necessarily the elapsed time in the real world, and sometimes it only represents the position in the sequence data. The active value h_t of the hidden layer with feedback edge is updated by the following formula:

$$h_1 = f(h_{t-1}, x_t), \tag{1}$$

where $h_0 = 0$. $f(\cdot)$ is a nonlinear function. Figure 1 shows an example of Recurrent Neural Network.

Assuming that the input to the RNN at moment t is x_t, the hidden layer state h_t is not only related to the input x_t at the current moment, but also related to the hidden layer state h_{t-1} at the previous time.

$$z_t = \mathbf{U}h_{t-1} + \mathbf{W}x_t + b, \tag{2}$$

$$h_t = f(z_t). \tag{3}$$

where Z_t is the net input of the hidden layer, $f(\cdot)$ is the nonlinear activation function, usually logistic function or Tanh function, $\mathbf{U}$ is the state-state weight matrix, $\mathbf{W}$ is the state-input weight matrix, and b is the bias term. Figure 2 shows the Recurrent Neural Network expanded by time.

h_{t-1} is a memory feature, which extracts the input features of the previous $t-1$ moments. Sometimes h_{t-1} is called the old state, and h_t is the new state. Therefore, the RNN model is particularly suitable for sequence problems. Structurally, the RNN can be regarded as a neural network model with loops, and it can be expanded into a standard neural network model, but this neural network is not separated. In this way, RNN performs the same calculation process each time, but the inputs are different each time, which seriously restricts the ability of RNN to capture features of long sequences.

2.3. Gated Recurrent Unit. Gated Recurrent Unit (GRU) is a kind of RNN with gated control units. Because the structure of the GRU unit is simpler and easier to train, the efficiency of training can be improved by using GRU. The GRU unit not only saves computing costs but also does not cause performance degradations. At present, there is no relevant research to point out that the performance of the GRU unit is worse than other recurrent networks. The input and output structure of GRU are the same as those of RNN.

The GRU combines the forget gate and the input gate into one: the update unit. In addition, GRU does not require additional memory units and introduces a linear dependency directly between the current state h_t and the historical state h_{t-1}. In the GRU network, the current candidate state is $\widetilde{h}_t$.

$$\widetilde{h}_t = \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h), \tag{4}$$

where $r_t \in [0,1]$ is a reset gate, which is used to control whether the computation of the candidate state $\widetilde{h}_t$ depends on the state h_{t-1} of the previous time.

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r). \tag{5}$$

When $r_t = 0$, the candidate state $\widetilde{h}_t = \tanh(W_c x_t + b)$ is related to the current input x_t, but not related to the history state. When $r_t = 1$, the candidate state $\widetilde{h}_t = \tanh(W_h x_t + U_h h_{t-1} + b_h)$ is related to the current input x_t and the historical state h_{t-1}, which is consistent with the simple recurrent network. The hidden state h_t of the GRU network is updated in the following way:

$$h_t = z_t \cdot h_{t-1} + (1 - z_t) \cdot \widetilde{h}_t. \tag{6}$$

$z \in [0,1]$ is the update gate, which controls how much information is retained by the current state from the historical state and how much new information it receives from the candidate state.

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z). \tag{7}$$

When $z_t = 0$, there is a nonlinear function between the current state h_t and the historical state h_{t-1}. If both $z_t = 0$ and $r = 1$ exist, the GRU network degenerates to the simple recurrent network. If both $z_t = 0$ and $r = 0$ exist, the current state h_t is only related to the current input x_t and not to the history state h_{t-1}. When $z_t = 1$, the current state h_t is equal to the previous state h_{t-1} and is independent of the current input x_t.

2.4. Memory Network. Generally, memory network can be regarded as composed of five components. The first component is a memory module $m = \{m_1, m_2, m_3, \ldots, m_n\}$ used to store memories, and this module is usually implemented with m_i as the matrix of indexes. The other four-module components are Input module, Generalization module, Output module, and Response module. These four modules are usually referred to simply as I, G, O, and R.

Memory network is a general machine learning framework so that memory network can target different problems. Due to the use of long-term memory components for learning performs better than RNN in long-term memory, so it is called memory network.

The Input module, Generalization module, Output module, and Response module can use any existing algorithm in the field of machine learning, such as SVM and random forest [25]. The working process of each of the four modules is introduced, respectively.

Module I: The function of module I is to do a simple preprocessing of the external input. Usually, the external input is transformed into a vector that is easier to handle in machine learning. For example, word2vec technology converts words into dense vectors.

Module G: The implementation of module G is very flexible. For example, the easiest way is to add the output of module I directly into the memory space. Literature [26] uses a first-in-first-out method to add new memories into the memory space when memory network is applied in the recommender systems.

$$m_{H(x)} = I(x), \tag{8}$$

where $H(\cdot)$ is the function of the selected slot, and $I(x)$ is the output of module I.

Module O: The most important task of module O is responsible for reading memory and generating outputs. Both module O and module G can be implemented in the simplest way, such as reading the memory in order.

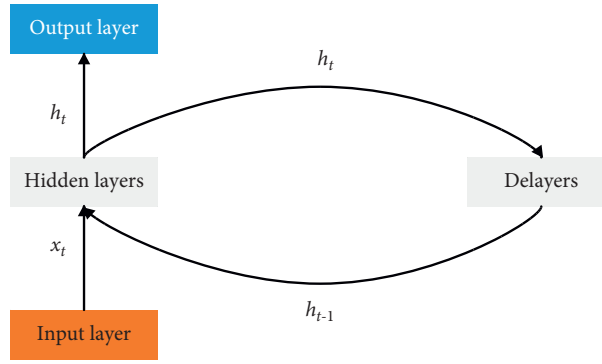


Figure 1: Simplified Recurrent Neural Network (RNN) structure diagram. The input of the Recurrent Neural Network subject consists of the x_t of the input layer and the hidden state of the previous moment h_{t-1}.

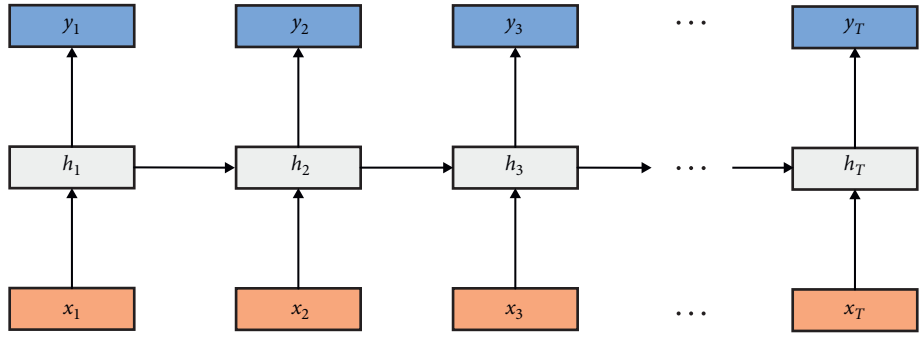


Figure 2: Recurrent Neural Network expanded by time. The RNN uses the results generated by the previous time step of the hidden layer, as part of the current time step, and influences the output of the current time step.

Module R: Module R converts the output of module O to the externally requested format.

2.5. Neural Turing Machine. Memory network is a branch of deep learning. The Facebook team's paper published in 2014 proposed memory network and introduced its application in Q&A systems [10]. In 2014, the Google DeepMind team used a similar idea to propose Neural Turing Machines (NTM) [11].

The NTM proposed by the DeepMind team refers to the idea of LSTM and generates an erase vector e_t and an add vector a_t for memory network to control the update of memory matrix.

The core of the model is module O and module R. Assuming that the input question in the Q&A systems is x, the task of module O is to select the TOP-N related memory from all the memories according to the input question vector. The specific selection method is first to select the most relevant memory.

$$o_1 = O_1(x, m) = \arg\max_{i=1,\ldots,N} s_O(x, m_i). \tag{9}$$

Next, select the memory o_2 that is most relevant to both of them based on the selected o_1 and input x together.

$$o_2 = O_2(x, m) = \arg\max_{i=1,\ldots,N} s_O\left(\left[x, m_{o_1}\right], m_i\right). \tag{10}$$

For equation (10) above, if linear vectors represent both x and o_1, they can be divided into the following way of addition:

$$s_O(x, m_i) + s_O\left(m_{o_1}, m_i\right). \tag{11}$$

Finally, module R needs to generate a text response r. The simplest response is to return m_{ok}, which is the output of the previously uttered sentences retrieved, and use the scoring function to calculate the relevance of all the candidate words to the input of module R, with the final word with the highest score being the correct answer.

$$r = \text{argmax}_{w \in W} s_R\left(\left[x, m_{o_1}, m_{o_2}\right], w\right), \tag{12}$$

where W is the set of all words in the dictionary and $s_R(\cdot)$ is the function that scores the matches.

3. The Proposed Model

3.1. Problem Formulation. Firstly, we introduce the overall structure of the model and then describe each module, respectively.

The hierarchical network with user memory matrix (HNUM^2) is a hierarchical network. The overall structure of the model is shown in Figure 3. The model consists of two layers. The first layer is a session-level GRU model, which is used to describe the sequence characteristics of the current session and store the user's short-term interests to predict

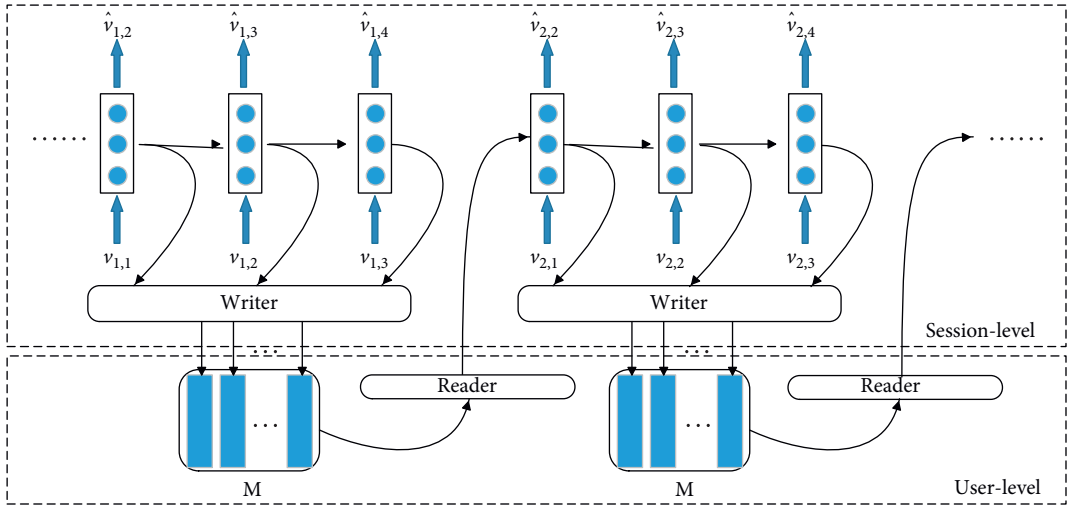


FIGURE 3: The overall structure of the model HNUM^2, which consists of two layers. The first layer is a session-level GRU model, and the second layer is a user-level memory network model.

the next item. The second layer is a user-level memory network model, which stores the entire user's historical information and describes the user's long-term interests. At the beginning of a user's session, the read module reads the memory vector in the memory matrix $\mathbf{M}$ corresponding to the current user and reads the memory as the user's preference vector to initialize the hidden layer of the GRU unit. At the same time, the user's hidden state, short-term interest, and the current stage of the click product are input to the session-level GRU unit. At the end of each time step, the output predicts the item clicks by this user in the next phase and the hidden state of the GRU in the next phase, which is stored into the memory matrix $\mathbf{M}$ by the write module. The same process is performed again when a user's session ends and the next session begins.

In Table 1, we introduce some of the notations used in this paper.

3.2. The Formal Description of HNUM2. Define $U = \{u_1, u_2, \ldots, u_N\}$ as the set of all users, $V = \{v_1, v_2, \ldots\}$ is a set of items, and $S^u = \{s_1^u, s_2^u, \ldots\}$ is the set of the session of user u. $V^s = \{v_1^s, v_2^s, \ldots\}$ is a sequence of interactive items generated in a user's sessions s, in which v_i is one of the interactive items in the whole model, and our goal is to predict the user's next interactive item $\hat{v}_{i+1}^s$. $\mathbf{M}^u = \{m_1^u, m_2^u, \ldots, m_K^u\} \in R^{D\times K}$ is the memory matrix of user u, and $m_k^u \in R^D$ is the k th memory vector of $\mathbf{M}^u$, which is used to store the long-term interests of the user. The size of $\mathbf{M}^u$ depends on the number of memory vectors K and the length D of the vectors in the memory matrix. Among them, K and D are the hyperparameters of the model.

3.3. Memory Reading Module. The read module is mainly responsible for reading the long-term interests of the user in the memory matrix, which is used to guide the training of the session phase. Specifically, set p^u to be the preference embedding of user u, and the interaction item v_i of the current session is used as input; p^u is obtained by reading the memory from $\mathbf{M}^u$. p^u can be expressed as

$$p^u = \text{READ}\left(\mathbf{M}^u, v_i\right). \tag{13}$$

v_i is the embedding vector of the i th interaction item in the current session. Intuitively, the previous i memory vectors will have different effects on the current interest, so the attention mechanism is introduced to assign weight values to different memory vectors.

The specific process of READ$(\cdot)$ operation is shown in equations (13)–(15).

$$w_{i,k} = v_i \cdot m_k^u, \tag{14}$$

$$z_{ik} = \frac{\exp\left(\beta w_{ik}\right)}{\sum_j \exp\left(\beta w_{ij}\right)}, \tag{15}$$

where β is an intensity parameter, which can enlarge or reduce the degree of focus. When $\beta = 1$ is a standard softmax, z_{ik} is used as the attention weight to derive the preference vector p^u for user u.

Therefore, the user's historical behavior can be accessed according to the impact of the user's historical behavior on the current item.

$$p^u = \sum_{k=1}^{K} z_{ik} \cdot m_k^u. \tag{16}$$

3.4. Memory Writing Module. The write module is responsible for updating the GRU hidden state into the memory matrix after a time step. Neural turing machine refers to the idea of the update gate of the LSTM:

(1) The input gate is used to determine the information to be added.

(2) The forget gate is used to determine the information to be discarded.

Table 1: Notations.

Symbol	Size	Description
U	$1 \times N$	The set of all users
S^u	$1 \times N$	The set of sessions for user u
V^s	$1 \times N$	The sequence of items that generate interactions in a session s
K		The number of memory vectors of memory matrix
D		The length of memory vector in matrix
$\mathbf{M}^u$	$R^{D\times K}$	The memory matrix of user u
m_k^u	R^D	The k-th memory vector of user u
p^u	K	Preference vector of user u
$\hat{r}_{s,i}$	R	The score of positive samples
$\hat{r}_{s,j}$	R	The score of negative samples

(3) The update gate is used to add or delete the information.

Specifically, the neural turing machine generates an erase vector and an add vector, in which the values of each element range from 0 to 1, indicating the information to be added or removed.

Since the whole process is matrix read and write operations are differentiable, the whole model parameters can be trained by gradient descent. For the erase vector erase_i:

$$\text{erase}_i = \sigma\left(E^T h_i + b_e\right). \tag{17}$$

$\sigma(\cdot)$ is the sigmoid function, E and b are the erase parameters, and h_i is the current hidden state of the user.

Update feature preference memory by attention weight and erase vector.

$$m_k^u \leftarrow m_k^u \cdot \left(1 - z_{ik} \cdot \text{erase}_i\right). \tag{18}$$

z_{ik} is the attention weight of the write phase.

After erasing, update the feature preference memory using the add vector add_i:

$$\text{add}_i = \tanh\left(A^T h_i + b_a\right), \tag{19}$$

$$m_k^u \leftarrow m_k^u + z_{ik}.\text{add}_i, \tag{20}$$

where A and b_a are the parameters in the add operation.

This erase-add updates strategy allows forgetting and reinforcing the learning process for the user preference embedding vector. The model can automatically learn to erase parameters and add parameters to determine which signals need to be weakened or enhanced.

3.5. Loss Function. Classical Bayesian Personalized Ranking (BPR) is a matrix factorization method using pairwise ranking loss [27]. BPR compares the scores of positive samples and negative samples [28]. In the iterative loss calculation process, the scores of the positive items are compared with the scores of the next item in the same batch, and their average value is used as the loss. The loss at a certain point in a session is defined as

$$L_s = -\frac{1}{N_s}\sum_{j=1}^{N_s} \ln\left(\sigma\left(\hat{r}_{s,i} - \hat{r}_{s,j}\right)\right), \tag{21}$$

where N_s is the number of samples, $\hat{r}_{s,i}$ is the score of the positive sample, and $\hat{r}_{s,j}$ is the score of the negative sample.

Both $\hat{r}_{s,i}$ and $\hat{r}_{s,j}$ are the output of GRU through the LeakyReLU activation function, and σ is the sigmoid function.

3.6. Hierarchical Network with User Memory Matrix

(1) We group sessions by the user set $U = \{u_1, u_2, \ldots, u_N\}$, and the sessions of each user u are arranged in chronological order. The sequence of user-item interactions in the session is arranged chronologically.

(2) In the training of the same user, the different sessions are horizontally stitched together to form a triplet ⟨UserId, SessionId, ItemId⟩ and sent into the session-level GRU.

(3) The read module reads the memory matrix $\mathbf{M}$ according to the GRU hidden state h_s of the user's current session s. The memory m_i^u read by the read module is used as the user's preference vector p^u to initialize the hidden layer unit of the GRU.

(4) The memory Write module writes the final state of the session to the memory network when a time step of the GRU ends and updates the memory matrix for training at the user-level.

The pseudocode of the HNUM2 execution process is shown in Algorithm 1.

4. Experiments

4.1. Datasets. (1). MovieLens-25M. MovieLens-25M (hereafter referred to as MovieLens) is a dataset provided by the MovieLens website developed by the GroupLens group at the University of Minnesota in the United States. MovieLens-25M is a publicly available dataset and is widely used in movie recommendations [29]. The version of the dataset used in this paper contains about 25 million rating records on the MovieLens website. To fit the algorithm proposed in this paper, the rating data for each user is sorted by time, and then the data is divided by days. We remove sessions with length less than 5 and we remove users with less than 6. For each user, 80% of sessions are used as training dataset and 20% as testing dataset.

input: triple < UserId, SessionId, ItemId >,
output: the prediction score $\widehat{V}^s = \{\widehat{v}_1^s, \widehat{v}_2^s, \ldots, \widehat{v}_m^s\}$.
(1) group the session by users into $U = \{u_1, u_2, \ldots, u_N\}$.
(2) initialize memory-matrix: **M**
(3) **for** i in epoch:
(4) **for** j in user u_i:
(5) //Session-level
(6) read **M** by reader into m_k^u as preference vector p^u
(7) **if** new session
(8) use p^u to initialize GRU hidden state h_s
(9) z_{ik} as the weight of user interest attention
(10) $m_k^u \leftarrow m_k^u \cdot (1 - z_{ik} \cdot \text{erase}_i)$, erase_i by equation (18)
(11) //User-level
(12) when the end of a time step
(13) write state to **M** by writer
(14) $m_k^u \leftarrow m_k^u + z_{ik}.\text{add}_i$, add_i by equation (20)
(15) computer the loss according equation (21)
(16) end for
(17) end for

Algorithm 1: HNUM^2.

(2). Adressa. Adressa [30] is a news dataset published in the RecTech item, which contains the contextual information about the user and details such as the headline and content of the news [31]. For registered users in the dataset, their historical behavior records can be obtained based on their IDs. The experiment in this paper needs to obtain user's long-term historical behavior information, so the registered users in the dataset can be selected as the experimental data. The dataset provides information such as the type of user's equipment and location [32]. There are start symbols and stop symbols of the session in the dataset, and the session can be divided accordingly. There are two versions of the dataset, one is a large dataset with 20 million reading behaviors with 10 weeks of traffic on the Adresseavisen news portal, and the other is a small dataset with 2 million reading behaviors with only one week of traffic. In this paper, we use a large dataset containing 20 million reading behaviors and filter out users with at least 5 sessions and at least 6 session lengths. We use 80% of these users as the training dataset and 20% as the testing dataset.

4.2. Evaluation Standard. **Recall@K**: Since the recommender systems can only recommend several items simultaneously, the actual items that users may choose should be in the first few items in the list. Therefore, the first evaluation metric of this paper is Recall@K, which indicates the proportion of required items among the top-K items in all test cases. In some scenarios, Recall does not consider the actual ranking of the items, while the absolute order is not important [33]. The traditional calculation formula of Recall is as follows:

$$\textbf{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{22}$$

where TP represents the number of positive samples predicted as positive samples, FN represents the number of positive samples predicted as negative samples, and Recall measures that multiple positive samples are divided into positive samples. In the personalized ranking task of the recommender systems, the calculation of Recall is defined as follows:

$$\textbf{Recall} = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |T(u)|}, \tag{23}$$

where $R(u)$ refer to the list of N items recommended for user u and $T(u)$ refer to the set of items preferred by user u in the testing dataset.

The work in this paper used the method of calculating Recall used in [7], which regarded session-based recommendation as a task of the item-by-item recommendation. There is only one target item in the current stage of the session. The final Recall score is the average of all users.

MRR@K: The second evaluation metric used in the experiment is Mean Reciprocal Rank (MRR), which is the average of the reciprocal rank of the required items. If the rank is greater than K, the reciprocal rank is set to 0. MRR considers the ranking of the items, which is very important in focusing on recommendations. The calculation formula is as follows:

$$\textbf{MRR} = \frac{1}{Q} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i}, \tag{24}$$

where $|Q|$ indicates the number of items of interest to the users, and rank_i indicates the ranking of items that the users are interested in, in the recommendation list. When the rank of the real value is greater than the set cut-off value, the inverse of the rank is set to 0. MRR better reflects the quality of the recommendation in the ranking problem, because people tend to pay more attention to the first few items in the recommendation list [34]. When the rank of the real value is very low, even if the real value is in the recommendation list, it cannot be considered a high-quality recommendation result.

4.3. Experimental Design. Firstly, we introduce the software and hardware platform used in the experiment. In this paper, we use the Tensorflow framework to build the model, and experiments are carried out on the hardware platform Tesla P100. During the training process, RMSProp is used as an optimizer to optimize the model, and the batch_size is set to 128. For the experimental environment, the better balance between performance and efficiency can be achieved when the batch_size is 128. The parameters of the model are initialized by the normal distribution, which has a mean of 0 and a standard deviation of 0.01. The initial learning rate is 0.001, and the attenuation coefficient of the learning rate is 0.96. To avoid overfitting, the parameter keep_prob of dropout is 0.8 and the number of GRU units is 100. It is found that, due to the complex structure of the network, the saturation of the activation function often occurs when using Tanh as the activation function, resulting in falsely high experimental results. Therefore, LeakyReLU is used as the activation function after the output layer of the GRU unit [35]. The formula of LeakyReLU function is as follows:

$$y_i = \begin{cases} x_i & \text{if } (x_i \geq 0) \\ a_i x_i & \text{if } (x_i < 0) \end{cases}, \tag{25}$$

where $a_i \in (0, 1)$. The LeakyReLU function does not produce saturation and avoids neuron death [36]. In the traditional memory matrix, the number of memory vectors is usually set within 2–15, and its length is set to 100. All hyperparameters are the optimal choices obtained after adjustment based on experimental results.

4.4. Analysis of Experimental Results

4.4.1. The Effectiveness of the Algorithm. To explore the recommendation performance of the HNUM^2 model, we compared the proposed model with the HGRU model and the GRU4REC model for experiments. The HGRU model and GRU4REC model are described below.

The GRU4REC model [7] is a classic session recommendation model based on deep learning, which uses GRU to capture the user's interests in the session and then generates a recommendation list according to the user's interests. This model is a common baseline algorithm model in the field of session recommendation.

The HGRU model [8] is a hierarchical session recommendation model in which both layers of the model use GRU units to capture user's interests. Throughout the session, the model evolves potential hidden states on RNN endpoints across sessions and uses hidden states on GRU to represent user's historical interests.

To compare the parameter settings of the experiments, the HNUM^2 model performs best when the number of memory vectors is 20 in our experiments, and we set the number of memory vectors to 20. The GRU4REC model uses 100 GRU units and the batch_size is set to 128. For the HGRU model, the number of GRU units in the session-level and the number of GRU units in the user-level are both set to 100.

As can be seen in Tables 2 and 3, the HGRU and HNUM^2 models have generally better recommendation results than the GRU4REC model in the session-based recommendation algorithm. The GRU4REC model does not consider the user's historical behavior information and captures the user's interests in the current session, whereas both the HGRU model and the HNUM^2 model utilize the user's historical behavior, so it has better recommendation performance. The HGRU model preforms weaker than HNUM^2 on Recall for the same dataset. Because the HGRU model compresses user's interests into the hidden states of GRU units when portraying user's long-term interests, this approach is not conducive to the dynamic of historical states. The proposed model using memory network avoids this situation. The experimental results show that the performance of each algorithm on the MovieLens dataset is worse than that on the Adressa dataset. MovieLens is not a dataset for session-based recommendations, and the dataset does not show that the chronological sequence of ratings is related to the viewing order. Therefore, the performance of the session-based recommendation algorithm on the MovieLens dataset is not ideal.

Compared with the baseline algorithms GRU4REC and HGRU, the HNUM^2 algorithm has better performance on Recall and MRR, which validates the effectiveness of the proposed algorithm.

4.4.2. Exploration of Long-Term Memory Ability. In order to explore the memory ability of the model to remember users' long-term interests, experiments were designed to compare the different performances of the model when the number of sessions was 10 and the number of sessions was 5. The two datasets were divided into a dataset with 5 sessions and a dataset with 10 sessions. The experiment compares the performance improvement ratio of the HGRU model and the HNUM^2 model when the number of sessions increases.

The main comparison is the memory ability of multiple sessions before a user, while the GRU4REC model only considers sessions and not users, so we do not compare GRU4REC.

Comparing the data in Tables 4 and 5 with the data in Table 2, it can be seen that both HGRU and HNUM^2 show improvements in Recall and MRR when the number of sessions is selected as 10, but the degree of improvement is different. Figures 3 and 4 compare the percentage performance improvement of the two models with 10 sessions versus 5 sessions.

As shown in Figures 5 and 6, it can be seen that the HNUM^2 model can obtain the improvement of recommendation effects when facing longer number of sessions. The reason is that the memory network can store long sequence of information, and more sessions can bring more user information, which can be stored in the memory network.

Table 2: Results of Recall@K and MRR@K on the Adressa dataset with 5 sessions.

	Recall@5	Recall@10	Recall@20	MRR@5	MRR@10	MRR@20
GRU4REC	0.1023	0.1854	0.3074	0.0620	0.0761	0.0846
HGRU	0.1607	0.2897	0.4529	0.0906	0.1104	0.1224
HNUM^2	**0.1638**	**0.2935**	**0.4550**	**0.0931**	**0.1129**	**0.1249**

Table 3: Results of Recall@K and MRR@K on the MovieLens-25M dataset.

	Recall@5	Recall@10	Recall@20	MRR@5	MRR@10	MRR@20
GRU4REC	0.0213	0.0450	0.0831	0.0090	0.0146	0.0173
HGRU	0.0247	0.0571	0.0958	0.0112	0.0191	0.0298
HNUM^2	**0.0286**	**0.0623**	**0.1034**	**0.0134**	**0.0227**	**0.0326**

Table 4: Results of Recall@K on the Adressa dataset with 10 sessions.

	Recall@5	Recall@10	Recall@15	Recall@20
HGRU	0.1682	0.3027	0.3976	0.4693
HNUM^2	**0.1775**	**0.3146**	**0.4096**	**0.4831**

Table 5: Results of MRR@K on the Adressa dataset with 10 sessions.

	MRR @5	MRR@10	MRR@15	MRR@20
HGRU	0.0954	0.1159	0.1240	0.1271
HNUM^2	**0.1018**	**0.1225**	**0.1307**	**0.1353**

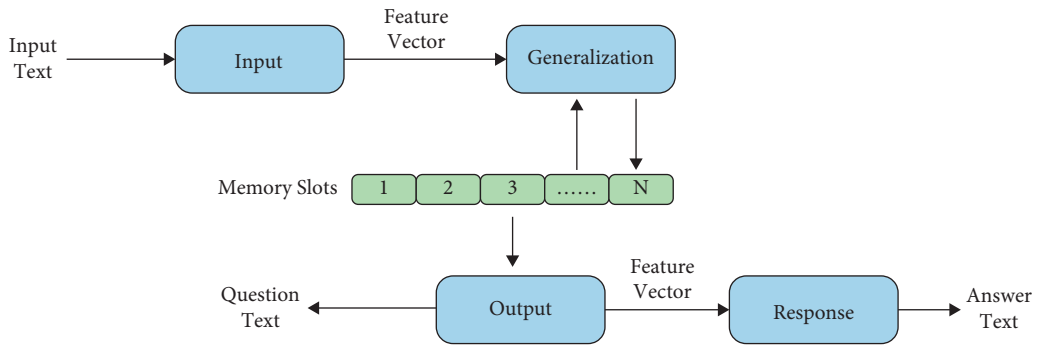


Figure 4: The general structure of the memory network, which consists of 5 components: Memory module, Input module, Generalization module, Output module, and Response module.

4.4.3. Effect of Parameter K on the Model. To explore the effect of different number of vectors K in the memory matrix on the model performance, a comparison experiment was designed. The experimental dataset was selected as the Adressa dataset that better fits the session recommendation model. The value of K for fixed TOP-K is constant at 20, while a dataset with 10 sessions is used. Different memory vector numbers K were selected to calculate the Recall and MRR values of the model. As shown in Figures 7 and 8, the values of Recall and MRR vary continuously with the value of K. The value of K determines the number of memory vectors, which in turn affects the memory ability of the model. As the number of memory vectors increases, the model can capture the user's long-term interests.

As shown in Figures 7 and 8, the value of K has a certain influence on the model effect, which shows an increasing trend followed by a decreasing trend. This is because more memory vectors can store more user information, and the user's interests that can be described become more accurate. As shown in Figure 7, the Recall of the model zigzags up for K values from 2 to 7, reaching an optimal value of 0.4813 at $K = 10$. As the value of K increases, there is no corresponding improvement in the recommendation performance of the model. However, when the number of

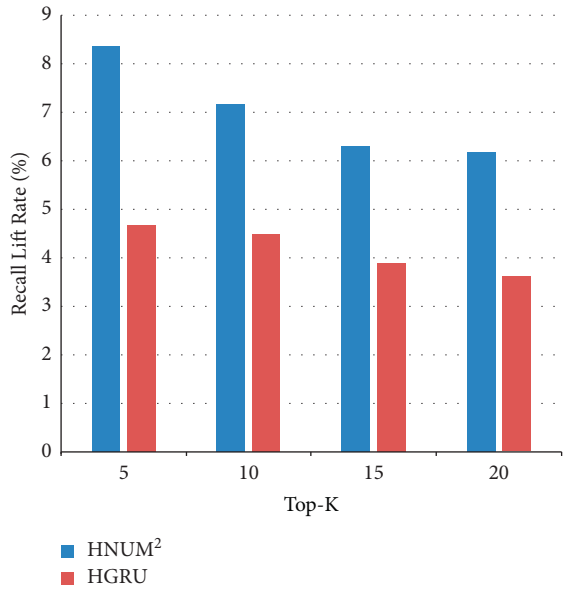


Figure 5: Comparison of long sessions on Recall improvement.

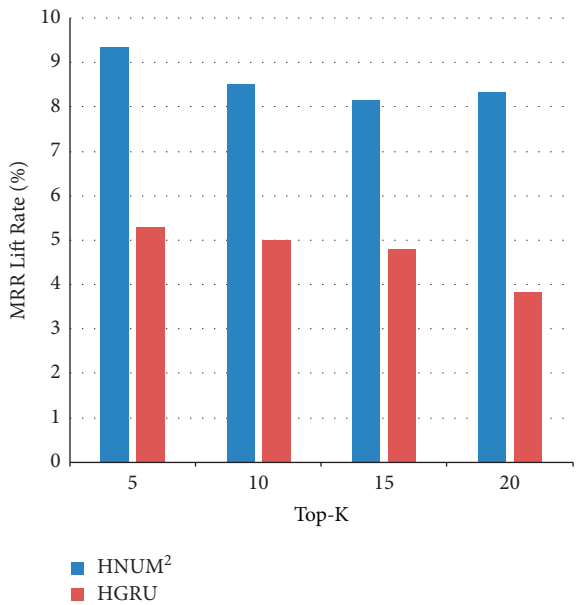


Figure 6: Comparison of long sessions on MRR improvement.

memory vectors increases to more than 10, the recommendation performance of the model begins to decline slightly. The reason is that the memory matrix generates more noise when the number of memory vectors is large. Figure 8 shows that the MRR achieves the optimal value of 0.1353 when the value is 10. Therefore, we can conclude that appropriately increasing the number of memory vectors can improve the memory ability of the memory matrix but also brings problems such as noise. Appropriate control of the number of parameters and the priority of the more important influencing factors is an important way to improve the recommendation performance.

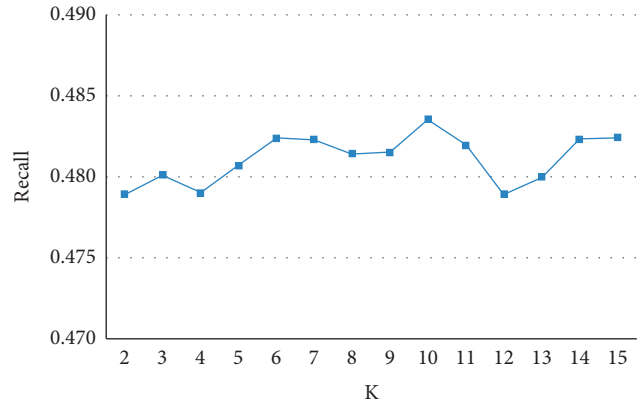


Figure 7: The influence of K on Recall.

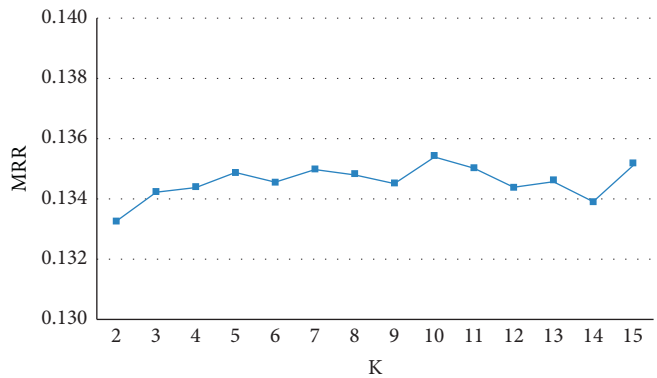


Figure 8: The influence of K on MRR.

5. Conclusion

In order to solve the problem of the insufficient memory ability of traditional recurrent neural networks, we proposed a hierarchical network with a user memory matrix (HNUM^2). In the proposed model, we use a memory network, which can capture the user's long-term interests and combine user's long-term and short-term interests to generate recommendations, which in turn improves the overall recommendation effectiveness of the algorithm. The experimental results show that the proposed model has better performance in session recommendation and better recommendation for problems with long sequences.

With the continuous improvement of information technology, the form of data has changed greatly, from the traditional scoring data to multisource heterogeneous information including images, text, and labels. The following work can further explore the fusion of multisource heterogeneous information. The current graph neural network as a new method for long sequence recommendation has opened up a new direction for sequence recommendation, and future work can apply memory networks to graph neural networks to improve the long sequence memory capability of the model.

Data Availability

All data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by Shandong Provincial Natural Science Foundation, China (ZR2020MF147).

References

[1] P. He, H. Wu, C. Zeng, and Y. Ma, "Truser: an Approach to service recommendation based on trusted users," *Chinese Journal of Computers*, vol. 42, no. 4, pp. 851–863, 2019.

[2] S. Wang, L. Hu, Y. Wang, L. Cao, Q. Sheng, and M. Orgun, "Sequential recommender systems: challenges, progress and prospects," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)*, pp. 6332–6338, Macao, China, July 2019.

[3] S. Wang, Y. Wang, Q. Sheng, M. Orgun, L. Cao, and D. Lian, "A survey on session-based recommender systems," *ACM Computing Surveys*, vol. 9, no. 4, 2021.

[4] L. Huang, B. Jiang, S. Lv, Y. Liu, and D. Li, "Survey on deep learning based recommender systems," *Chinese Journal of Computers*, vol. 41, no. 7, pp. 1619–1647, 2018.

[5] G. Lu and W. Zhang, "Survey of deep learning applied in recommendation system," *Software Engineering*, vol. 23, no. 2, pp. 5–8, 2020.

[6] G.-E. Yap, X.-L. Li, and P. S. Yu, "Effective next-items recommendation via personalized sequential pattern mining," *Database Systems for Advanced Applications*, vol. 7239, pp. 48–64, 2012.

[7] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," in *Proceedings of the International Conference on Learning Representations (ICLR)*, Vancouver, Canada, April 2016.

[8] M. Quadrana, A. Karatzoglou, and B. Hidasi, "Personalizing session-based recommendations with hierarchical recurrent neural networks," in *Proceedings of the Conference on Recommender Systems*, pp. 130–137, Como, Italy, August 2017.

[9] J. Liu, Y. Wang, and X. Luo, "Research and development on deep memory network," *Chinese Journal of Computers*, vol. 43, no. 2, pp. 1–52, 2020.

[10] J. Weston, S. Chopra, and A. Bordes, "Memory networks," in *Proceedings of the International Conference on Learning Representations (ICLR)*, Vancouver, Canada, November 2015.

[11] A. Graves, G. Wayne, and I. Danihelka, "Neural turing machines," arXiv preprint arXiv:1410.5401, 2014.

[12] S. Sukhbaatar, A. Szlam, and J. Weston, "End-to-end memory networks," *Advances in Neural Information Processing Systems*, vol. 28, pp. 2440–2448, 2015.

[13] X. Chen, H. Xu, and Y. Zhang, "Sequential recommendation with user memory networks," in *Proceedings of the International Conference on Web Search and Data Mininghttp*, pp. 108–116, Melbourne, Australia, February 2018.

[14] J. Huang, W. Zhao, and H. Dou, "Improving sequential recommendation with knowledge-enhanced memory networks," in *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 505–514, Ann Arobor, MI, USA, June 2018.

[15] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[16] R. Ma, Q. Zhang, and J. Wang, "Mention recommendation for multimodal microblog with cross-attention memory network," in *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 195–204, Ann Arbor, MI, USA, June 2018.

[17] Z. Sun, Y. Wang, Z. Cai, T. Liu, X. Tong, and N. Jiang, "A two-stage privacy protection mechanism based on blockchain in mobile crowdsourcing," *International Journal of Intelligent Systems*, vol. 36, no. 5, pp. 2058–2080, 2021.

[18] X. Xia, H. Yin, J. Yu, Q. Wang, L. Cui, and X. Zhang, "Self-supervised hypergraph convolutional networks for session-based recommendation," *Association for the Advancement of Artificial Intelligence*, https://arxiv.org/abs/2012.06852, 2020.

[19] X. Yue, Y. Liu, and C. Yu, "Session-based multi-rate RNN recommendation model," *Journal of Shanxi University*, vol. 42, pp. 332–339, 2019.

[20] M. Aghdam, N. Hariri, and B. Mobasher, "Adapting recommendations to contextual changes using hierarchical hidden Markov models," in *Proceedings of the 9th ACM Conference on Recommender Systems*, pp. 241–244, Vienna, Austria, September 2015.

[21] W. Gu, S. Dong, and Z. Zeng, "Increasing recommended effectiveness with Markov chains and purchase intervals," *Neural Computing & Applications*, vol. 25, no. 5, pp. 1153–1162, 2014.

[22] Q. He, D. Jiang, and Z. Liao, "Web query recommendation via sequential query prediction," in *Proceedings of the 2009 IEEE 25th International Conference on Data Engineering*, pp. 1443–1454, Shanghai, China, April 2009.

[23] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart Cyber-Physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[24] M. Gao and B. Xu, "Recommendation algorithm based on recurrent neural network," *Computer Engineering*, vol. 45, no. 8, pp. 198–202+209, 2019.

[25] Y. Zhan, X. Luo, and Y. Wang, "Supervised hierarchical deep hashing for cross-modal retrieval," in *Proceedings of the ACM International Conference on Multimedia*, pp. 3386–3394, Seattle, WA, USA, August 2020.

[26] Y. Song and J. Lee, "Augmenting recurrent neural networks with high-order user-contextual preference for session-based recommendation," 2018, https://arxiv.org/abs/1805.02983.

[27] Y. Wang, Z. Cai, Z.-H. Zhan, B. Zhao, X. Tong, and L. Qi, "Walrasian equilibrium-based multiobjective optimization for task allocation in mobile crowdsourcing," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 4, pp. 1033–1046, 2020.

[28] Y. Zhan, Y. Wang, Y. Sun, X. Wu, X. Luo, and X. Xu, "Discrete online cross-modal hashing," *Pattern Recognition*, vol. 122, Article ID 108262, 2021.

[29] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *Proceedings of the 39th IEEE International Conference on Distributed Computing Systems (ICDCS)*, pp. 144–153, Dallas, TX, USA, July 2019.

[30] J. Gulla, L. Zhang, and P. Liu, "The Adressa dataset for news recommendation," in *Proceedings of the International*

Conference on Web Intelligence, pp. 1042–1048, Leipzig, Germany, August 2017.

[31] Y. Wang, Z. Chen, X. Luo, and X. Xu, "High-dimensional sparse cross-modal hashing with fine-grained similarity embedding," in *Proceedings of the Web Conference 2021*, pp. 2900–2909, New York, NY, USA, April 2021.

[32] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[33] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks: a survey towards private and secure applications," *ACM Computing Surveys*, vol. 37, no. 4, 2020.

[34] Y. Wang, Y. Gao, Y. Li, and X. Tong, "A worker-selection incentive mechanism for optimizing platform-centric mobile crowdsourcing systems," *Computer Networks*, vol. 171, no. C, 2020.

[35] T. Liu, Y. Wang, Y. Li, X. Tong, L. Qi, and N. Jiang, "Privacy protection based on stream cipher for spatiotemporal data in IoT," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 7928–7940, 2020.

[36] Z. Lu, Y. Wang, X. Tong, P. Wang, C. Mu, and Y. Li, "Data-driven many-objective crowd user selection for mobile crowdsourcing in industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 99, 2021.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 2479314, 8 pages
https://doi.org/10.1155/2022/2479314

Research Article

Hybrid Collaborative Filtering Algorithm Based on Sparse Rating Matrix and User Preference

Hengtao Wang, Hongman Wang, Fangchun Yang, and Jinglin Li

Engineering Research Center of Information Network, Ministry of Education,
School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing, China

Correspondence should be addressed to Hongman Wang; wanghm@bupt.edu.cn

Received 12 October 2021; Accepted 12 January 2022; Published 31 January 2022

Academic Editor: Yingjie Wang

This study presents a hybrid collaborative filtering recommendation algorithm for sparse data (HCFDS) to increase the recommendation impact by addressing the problem of data sparsity in standard collaborative filtering methods. To begin, the similarity calculation divergence is evident in a data sparse environment due to the difference in user scoring standards and the rise in weight of the same score in the overall score. The user similarity algorithm IU-CS and item similarity algorithm II-CS are suggested in this work by incorporating the score difference threshold and the same score penalty factor, in order to address the deviation of similarity computation caused by the excessive dilation. Second, this work offers a filling optimization technique for score prediction to address the issue of missing score matrix data. The II-CS algorithm presented in this work is used to forecast the missing items in the scoring matrix first, and then, the user's preference score in the item category dimension is utilized to correct the score prediction value and fill the matrix. Finally, the IU-CS method presented in this work is used in this study to provide recommendations on the filled score matrix. Experiments indicate that, when compared to the preoptimization method and other algorithms, the optimized algorithm successfully solves the problem of data sparsity and the recommendation accuracy is considerably increased.

1. Introduction

The fast advancement of Internet technology has resulted in a quick increase in data on the network, reducing the efficiency of information gathering significantly. The recommendation system [1] is a good way to tackle the difficulties mentioned above. It can suggest interesting material for users based on their previous information attributes and activities.

The algorithm at the heart of a recommendation system involves collaborative filtering, content, association rules, and hybrid recommendation [1]. Among these, the collaborative filtering algorithm [2] is the most widespread and is well-known, consisting mostly of two collaborative filtering algorithms based on user [3] and item [4]. The most significant distinction between the two algorithms is their "similarity" metric, which supports the recommendations' proposal. In both algorithms, "similarity" takes under consideration users' and items' similarities. However, the user-based collaborative filtering algorithm is to recommend according to the similarity between users, while the project-based collaborative filtering algorithm is to recommend according to the similarity between projects.

The calculation of similarity is the heart of the collaborative filtering algorithm [3], and data sparsity and similarity calculation techniques have an impact on the correctness of the output. The problem of data sparsity arises from the vast number of users and items in the recommendation system, and users are unable to rate all things, resulting in a substantial amount of data missing in the user-item scoring matrix, which has a direct impact on recommendation accuracy. When the score matrix data is sparse, the difference in user scoring standards and the weight of the same score in the overall score rises, and the calculation deviation of user similarity rises. Due to a lack of data on user ratings, each user's rating has a significant impact on the computation of item similarity, resulting in a divergence of results.

Domestic and international academics have been conducting extensive studies in order to address the data sparsity and similarity calculation accuracy issues.

To solve the problem of data sparsity, C. Li and Ma. [5] created the score matrix comprising the user's average weighted score and the item's average weighted score. Deng et al. [6] used a user-based collaborative filtering recommendation algorithm to make predictions, filling in the score gaps with the predicted values as the intermediate results, and then made recommendations based on item similarity. Yue Xi et al. [2] proposed first using the cosine similarity to calculate the similarity between items and then predicting the score to fill the data. The above method uses cosine similarity to predict the score when the data is sparse, and the accuracy of the predicted value is low.

The problem of data sparsity causes the cosine similarity computation to deviate. Ruonan Ji et al. [7] improved the modified cosine similarity of users by combining implicit feedback and time characteristics of users, and Wang and Zheng [3] weighted similarity to improve cosine similarity by considering the items of users' common score and the proportion in the sum of users' total score. X. Gao et al. [8] utilize cosine similarity to determine user similarity based on the modified score data and compensate for the score deviation caused by user status.

As far as the degree of similarity is concerned, Qu et al. [1] analyzes the degree of similarity of every characteristic of an item individually and then utilizes the neural network of BP (back-propagation) to obtain the final degree of similarity. Huo et al. [9] used the attenuation feature of the Logistic Equation for the purpose of adding the user's time component of interest to the calculation of similarity. Zhao et al. [10] and others use similarity analysis and weight calculation to identify the element's similarity to extract video features based on usage behavior, item name, and other factors.

The aforementioned research has used various techniques to enhance similarity calculation methods, the majority of which include criteria other than scores, and while the accuracy has increased to some extent, the data sparsity problem has not been addressed.

To address the problem of data sparsity, this paper offers a hybrid collaborative filtering recommendation method for sparse data (HCFDS). Firstly, in order to solve the problem of obvious deviation of similarity calculation due to the difference of user scoring standards and the increase of the weight of the same score in the total score in the data sparse environment, this algorithm introduces the scoring difference threshold and the same scoring penalty factor on the basis of the work by J. S. Breese et al. [11] and user similarity algorithm IU-CS and item similarity algorithm II-CS are proposed. Secondly, to address the problem of missing score matrix data, this study proposes a filling optimization approach for score prediction based on the work by Deng et al. [6]. In this method, the missing items in the scoring matrix are preliminarily predicted and scored by using the II-CS algorithm proposed in this paper, and then the predicted scores are corrected by using the user's preference scores in the item category dimension to improve the rationality of filling scores. Finally, the HCFDS algorithm uses IU-CS provided in this paper to make score predictions on the filled matrix. This technique overcomes the problem of data sparsity while simultaneously enhancing the accuracy of recommendations.

2. Methods

This paper offers a hybrid collaborative filtering recommendation algorithm for sparse data (HCFDS) to tackle the problem of data sparsity. To begin with, in the event of sparse data, the difference in user rating criteria and the weight of the same rating in the similarity contribution rise, resulting in a higher level of significant deviation in user similarity computation. Therefore, in this study, firstly, the user ratings are normalized, and then the user rating difference threshold and the same rating penalty factor are introduced, and a user similarity optimization algorithm IU-CS is proposed. After correcting the significant influence of users' difference in high scores on a single item and the influence of higher proportion of the same score in the total score on users' similarity, the algorithm introduces the threshold value of item score difference and the penalty factor of item score. This II-CS algorithm aims at the problems that (1) the single user's score difference between items is too large with fewer users score data and that (2) the weight of the same score in similarity contribution has a great influence on the calculation results of item similarity with few users. An item similarity optimization algorithm II-CS is proposed, which corrects the high score difference of individual users between items, reduces the contribution weight of the same score similarity of items, and optimizes the calculation of item similarity. Secondly, aiming at the problem of missing data in the scoring matrix, a scoring prediction filling optimization method is proposed, which uses the above-mentioned II-CS algorithm to preliminarily predict the score and then uses the item category preference score to correct the filling score. Finally, the IU-CS algorithm proposed above is used for recommendation on the filled matrix. The following is a description of the algorithm in this paper.

2.1. User Similarity Optimization Algorithm IU-CS Based on User Score Difference Threshold and Same Score Penalty Factor. Cosine similarity is a common computing method in collaborative filtering algorithm. The degree of similarity between users is shown in the following formula:

$$\operatorname{sim}(x, y) = \frac{|P(x) \cap P(y)|}{\sqrt{|P(x)||P(y)|}}. \tag{1}$$

Here, $P(x)$ represents the item set of user x's behavior.

Formula (1) measures the similarity between users according to the cooccurrence of users in items but does not consider the influence of popular items, that is, users' behaviors on popular items cannot reflect their personal preferences, whereas behaviors on some unpopular items can better explain their personal interests. Therefore, J. S. Breese et al. [11] punish popular items to reduce the

contribution of popular items to user similarity, as shown in the following formula:

$$\mathrm{sim}(x, y) = \frac{\sum_{t \in V} (1/\log(1 + |Q(t)|))}{\sqrt{|P(x)||P(y)|}}. \quad (2)$$

Here, $Q(t)$ is the user set that produces behavior on item t, and V represents user x and user y common item set.

Formula (2) only considers the user's behavior on the item and does not consider the significant influence of the user's rating difference and the same rating weight on the similarity calculation under the condition of sparse data. In order to improve the accuracy of the results, the following improvements are made based on formula (2).

2.1.1. Normalization of User Ratings. Since different users have different rating standards, the same rating may represent different meanings, so it is necessary to normalize the rating according to the maximum value of users' rating, as shown in the following formula:

$$s'_{ut} = \frac{s_{ut}}{\max(R_u)}. \quad (3)$$

Here, s_{ut} represents the score of user u on item t, and s'_{ut} is the normalized result. $\max(R_u)$ represents the maximum score of user u.

2.1.2. Introducing User Score Difference Threshold to Correct the Significant Influence of High Rating Difference on User Similarity under Sparse Data. In the case of sparse data, the number of items evaluated by users is relatively small, and the difference of individual items between users will obviously influence the result of similarity calculation.

In order to correct the influence of user score difference on similarity under sparse data, a threshold of score difference α is introduced. When the absolute value of different users' score differences for the same item is below the threshold, it is considered that the score has a positive effect on similarity; otherwise, it is a bad influence, as shown in the following formula:

$$\mathrm{sim}(x, y) = \frac{\sum_{t \in V} \left(\left(\alpha - abs\left(s'_{xt} - s'_{yt}\right)\right)/\log(1 + |Q(t)|)\right)}{\sqrt{|P(x)||P(y)|}}. \quad (4)$$

In formula (4), s'_{xt} and s'_{yt} represent user x and user y's normalized ratings of item t.

2.1.3. Punishing the Similarity Contribution Weight of the Same Score with Less Common Items among Users. In formula (4), when users have the same score for the same item, there is the greatest influence on similarity. However, different users produce the same score for the same item with low frequency. Therefore, under normal circumstances, the same score accounts for a relatively low proportion in the positive contribution of similarity, but few items shared by users will lead to a higher proportion of the same score in the positive contribution of similarity, which will lead to the deviation of calculation results, especially in the case of sparse data. In order to reduce the influence of its high proportion on similarity, it is necessary to punish the same score with less common items and reduce its positive contribution to similarity.

On the basis of equation (4), the penalty factor λ_u of user's same score is introduced to punish the same score of users who have less common items and reduce their positive contribution to the similarity.

$$\mathrm{sim}(x, y) = \frac{\sum_{t \in V} \left(\left(\alpha - abs\left(s'_{xt} - s'_{yt}\right)\right)/\log(1 + |Q(t)|)\right) \bullet \varepsilon}{\sqrt{|P(x)||P(y)|}},$$

$$\varepsilon = \begin{cases} \dfrac{\min(\mathrm{Len}(V), \lambda_u)}{\lambda_u}, & s'_{xt} = s'_{yt}, \\ 1, & s'_{xt} \neq s'_{yt} \end{cases} \quad (5)$$

Here, $\mathrm{Len}(V)$ is the common item set length of user x and user y.

2.2. An Optimization Algorithm II-CS for Item Similarity Degree Based on Item Difference Threshold and Penalty Factor of the Same Score. The similarity of items is improved with the idea of improving similarity of users. J. S. Breese et al. [11] penalize active users for improvements based on cosine similarity, as shown in the following equation:

$$\mathrm{sim}(a, b) = \frac{\sum_{x \in U} (1/\log(1 + |P(x)|))}{\sqrt{|Q(a)||Q(b)|}}. \quad (6)$$

Here, U stands for a set of users who score items a and b, $P(x)$ stands for a list of items for user x, and $|Q(a)|$ and $|Q(b)|$ stand for sets of users who score items a and b, respectively.

On the basis of formula (6), the factors such as the score difference between items and the number of common users of items are introduced for improvement.

2.2.1. Introducing the Threshold of Item Score Difference to Correct the Significant Influence of High Score Difference on Item Similarity under Sparse Data. In the case of sparse data, the user's item score data will be less, and the individual user's item score has a significant impact on the calculation of similarity, resulting in a deviation of results. In order to reduce the influence of individual user's item score on the calculation of similarity, the difference threshold β of item score is introduced. When the absolute value of user's item score difference is below the threshold, the similarity will increase; otherwise, the similarity will decrease.

The formula for calculating the similarity after the score is introduced is as follows:

$$\mathrm{sim}(a, b) = \frac{\sum_{x \in U} \left(\left(\beta - abs\left(s_{xa} - s_{xb}\right)\right)/\log(1 + |P(x)|)\right)}{\sqrt{|Q(a)||Q(b)|}}. \quad (7)$$

In formula (7), s_{xa} and s_{xb} represent user x's ratings of items a and b, respectively.

2.2.2. The Similarity Contribution Weight of the Same Score with Few Common Users Is Penalized. Because the ratings of different items by the same user may be influenced by internal factors such as personal preference, when there are few common users of items, it is difficult for users to exclude the influence of internal factors from the same ratings of different items. However, the positive contribution of the same ratings to the similarity of items at this time is relatively high, which leads to the deviation of calculation results. In the case of sparse data, the number of common users will be even less, which is particularly significant. Based on this situation, it is necessary to reduce its proportion to correct the influence of users' internal factors on item similarity. The penalty factor λ_i of item's same score is introduced to punish the item's same score with few common users on the basis of formula (7) and reduce its positive contribution to similarity. The final improved formula is as follows:

$$\text{sim}(a,b) = \frac{\sum_{x\in U}\left(\left(\beta - abs\left(s_{xa} - s_{xb}\right)\right)/\log\left(1+|P(x)|\right)\right)\bullet\varepsilon}{\sqrt{|Q(a)||Q(b)|}},$$

$$\varepsilon = \begin{cases} \dfrac{\min\left(\text{Len}(U),\lambda_i\right)}{\lambda_i}, & s_{xa} = s_{xb}, \\ 1, & s_{xa} \neq s_{xb}. \end{cases} \tag{8}$$

Here, $\text{Len}(U)$ is the length of the common user list for items a and b.

2.3. Optimization of Sparse Score Matrix Vacancy Prediction and Filling. For the missing values in the scoring matrix, this paper proposes an optimization method for the filling of scoring prediction; the item category preference score was used to revise the filling score.

2.3.1. Score Matrix Filling Based on II-CS. The score matrix is shown in Table 1. U_i represents user i, I_i represents item i, and the values in the matrix represent the user's score on the item. *?* indicates that the user has not scored an item.

Because there are a lot of missing values in the matrix, it is impossible to calculate the user similarity when the data is sparse. For example, if you calculate the similarity between U_1 and U_2, you cannot use formula (5) to calculate the user similarity.

In order to solve the problem of data sparsity, firstly, the II-CS algorithm is used to predict the users' ungraded items based on the existing data in the scoring matrix, and then the results are filled into the matrix. The predicted score of user x for item a is calculated by the following formula:

$$F(x,a) = \overline{s_a} + \frac{\sum_{b\in n}\text{sim}(a,b)\left(s_{xb} - \overline{s_a}\right)}{\sum_{b\in n}|\text{sim}(a,b)|}. \tag{9}$$

Here, n is a similar set of items for a, s_{xb} is user x's rating of item b, and $\overline{s_a}$ is item a's average rating.

2.3.2. Integration with User Item Category Preference Score. Because the preliminary prediction score by formula (9) only depends on the user's scoring data, and in fact the user's scoring behavior will also be affected by their category preference, in order to make the filled score more reasonable, the prediction score is revised by calculating the user's category preference score for the item.

Table 1: User-item scoring matrix.

	I_1	I_2	I_3	I_4
U_1	1	?	3	2
U_2	?	4	?	?
U_3	?	2	1	?
U_4	2	?	5	?

Each item may correspond to multiple categories, and the items-category matrix is shown in Table 2.

In Table 2, C_{ij} represents whether item i belongs to category j. If item i belongs to category j, the value is 1; otherwise, it is 0.

Combining the user-item scoring matrix and the item-category matrix, the user's score of the item category is calculated. User x ratings for item categories i are defined as follows:

$$f(x,i) = \frac{\sum_{C_{ai}=1, a\in P(x)} s_{xa}}{\text{Len}(x,i)}. \tag{10}$$

Here, $\text{Len}(x,i)$ is the number of items in category i in the list of items for user x, and a is the item for which user x has a score.

The degree of preference of users for different categories is different, and the degree of preference of users x for item categories i is calculated as shown in the following formula:

$$\varphi(x,i) = \frac{\text{Len}(x,i)}{\text{Len}(x)}. \tag{11}$$

Here, $\text{Len}(x)$ is the number of items that the user has scored.

An item may belong to more than one category. The score of the item on the category preference dimension is influenced by the category preference degree and the category score. Comprehensive formulas (10) and (11) are used to score the category preference of user x for item a, as shown in the following formula:

$$G(x,a) = \frac{\sum_{i\in C(a)} f(x,i)\varphi(x,i)}{\sum_{i\in C(a)}\varphi(x,i)}. \tag{12}$$

In formula (12), $c(a)$ represents the category set to which item a belongs. The initial item-based collaborative filtering prediction score is revised using the user's rating on the item's category preference, and the final filled score matrix is as follows:

$$R(x,a) = kF(x,a) + (1-k)G(x,a). \tag{13}$$

In formula (13), k is the score correction coefficient.

2.4. Improved Algorithm Description. To solve the problem of data sparsity, this paper proposes a hybrid collaborative filtering algorithm for sparse data (HCFDS). Firstly, the missing items in the scoring matrix are preliminarily

Table 2: Items-category matrix.

	C_1	C_2	...	C_i
I_1	1	0	...	1
I_2	1	1	...	0
I_3	0	1	...	1
I_4	1	1	...	1

predicted and scored by using the II-CS algorithm proposed in this paper. Secondly, the user's preference score in the item category dimension is used to correct the score prediction value. Finally, the IU-CS algorithm proposed in this paper is used to make recommendations on the filled score matrix.

The pseudocode of HCFDS is as follows:

(i) HCFDS
(ii) **INPUT**: trainset, testset, cateitem, cateuser
(iii) **OUTPUT**: L
(iv) **BEGIN**:
(1) FOR item $a \in$ trainset
(2) FOR item $b \in$ trainset
(3) IF a ! = b
(4) *sim(a,b)* = formula (8)
(5) END FOR
(6) END FOR
(7) FOR user $x \in$ trainset
(8) FOR item $a \in$ itemset (x) − ownset (x)
(9) IF cateitem(a) ∩cateuser(x) ! = ϕ
(10) $G(x, a)$ = formula (12)
(11) *R(x, a)* = formula (13)
(12) trainset.*add* (user, item, *R(x, a)*)
(13) END FOR
(14) END FOR
(15) FOR user $x \in$ trainset
(16) FOR user $y \in$ trainset
(17) IF x ! = y
(18) *sim (x, y)* = formula (5)
(19) END FOR
(20) END FOR
(21) FOR user $u \in$ testset
(22) FOR item $a \in$ itemset (u) − ownset (u)
(23) $P(u, a)$ = formula (14)
(24) END FOR
(25) END FOR
(26) L = sort (u, P, N)
(i) **RETURN** L
(ii) **END**

In the pseudocode, trainset stands for training set, testset stands for test set, L stands for a user generated list of recommendations of length N, itemset stands for item set, ownset stands for user-owned item set, cateitem stands for category set, and category user represents a collection of user-owned item categories.

The final algorithmic steps in this article are as follows:

(i) Step 1: on the initial dataset, HCFDS uses the item II-CS optimization algorithm to calculate the item similarity by formula (8).
(ii) Step 2: using the similarity of items in step 1, the user's unrated items in the user-item scoring matrix are scored and predicted. Select the nearest neighbor of the item and get the preliminary prediction score $F(x, a)$ by formula (9).
(iii) Step 3: HCFDS calculates the user's score $G(x, a)$ on category preference by formula (12), revises the initial predicted score in step 2, gets the final filling score $R(x, a)$, fills in the missing items in the scoring matrix, and solves the problem of data sparsity.
(iv) Step 4: on the basis of filling the matrix in step 3, using IU-CS optimization algorithm, the user similarity is calculated by formula (5).
(v) Step 5: HCFDS uses the user similarity in step 4 to predict the score. It selects the user's nearest neighbor and calculates the prediction score of the user's unrated items by the following formula:

$$P(x, a) = \overline{s_x} + \frac{\sum_{y \in m} \text{sim}(x, y)\left(y_a - \overline{s_y}\right)}{\sum_{y \in m} |\text{sim}(x, y)|}. \tag{14}$$

(vi) In formula (14), m is a similar set of users x, y_a is the score of user y on item a, and $\overline{s_x}$ and $\overline{s_y}$ are the average scores of users x and y.
(vii) Step 6: according to the score prediction results ranking, HCFDS selects the highest score of the first N items for recommendation.

3. Experiments

Aiming at the above-mentioned algorithms proposed in this paper, firstly, the comparison experiments of user-based and item-based collaborative filtering algorithm before and after improvement are carried out to verify the effectiveness of the improved similarity calculation. Then, the comparison experiments are carried out between the final optimized algorithm HCFDS and the improved user-based collaborative filtering algorithm to verify the effectiveness of the HCFDS algorithm in solving the data sparsity problem. Finally, the experimental comparison is made between the HCFDS algorithm and other algorithms to verify the improved performance of the algorithm.

3.1. Dataset. The dataset used in this experiment is MovieLens dataset provided by GroupLens project team of University of Minnesota. The scale of data used in the experiment is 100 K, including 100,000 pieces of historical scoring data of 1682 movies by 943 users. $u1$ base is selected as the training set, and $u1$ test is selected as the test set.

3.2. Evaluation Metrics. In this paper, the Mean Absolute Error (MAE) is used as an evaluation index, which reflects the error between the predicted value and the true value of the item score. The smaller the value of MAE, the higher the accuracy of the improved algorithm. If the predicted score is $\{r_1, r_2, r_3, \ldots, r_N\}$ and the actual score is $\{s_1, s_2, s_3, \ldots, s_N\}$, the calculation formula is as follows:

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |r_i - s_i|}{N}. \tag{15}$$

In formula (15), N is the number of prediction scores, r_i is the prediction score of item i, and s_i is the actual score of item i.

3.3. Experimental Results and Analysis

3.3.1. Comparison before and after Improvement of User-Based Collaborative Filtering Algorithm. The enhanced similarity calculation formula (5) is used to improve the user-based collaborative filtering (UBCF) algorithm. The user rating difference threshold and the same rating penalty factor choose the value that corresponds to the minimal MAE and different neighbors K for trials, respectively. As demonstrated in Figure 1, the outcomes before and after the algorithm improvement are compared.

It can be seen from Figure 1 that MAE decreases with the increase of the number of neighbors K before and after the improvement of the algorithm, indicating that the more the number of neighbors is, the more accurate the prediction result is. It can be clearly seen from the figure that the improved algorithm has smaller MAE under different K values, and when K is 5, it increases by 3.6%. With the increase of K value, MAE tends to be stable and increases by 1.5% when K is 50. The result of scoring prediction is more accurate, which shows that the improvement can improve the algorithm's performance.

3.3.2. Comparison before and after Improvement of Item-Based Collaborative Filtering Algorithm. The item-based collaborative filtering technique is improved using the improved item similarity calculation formula (8). The best solution is the equivalent value when MAE is the least, and different neighbors K are chosen for the experiment based on the difference threshold of item score and the penalty factor of the same score. Figure 2 depicts a comparison of the outcomes before and after the algorithm modification.

It can be seen from Figure 2 that, under different neighbor numbers K, the MAE of the improved algorithm is significantly lower than that before the improvement, and when K is smaller, it is increased by 4.7% when K is 5. With the increase of K value, MAE tends to be stable and increases by 4.5% when K is 50. It is shown that the improved algorithm is effective and can improve the accuracy of prediction.

3.3.3. Comparison between HCFDS Algorithm and Improved UBCF Algorithm. To solve the problem of data sparsity, this paper uses the improved item-based collaborative filtering

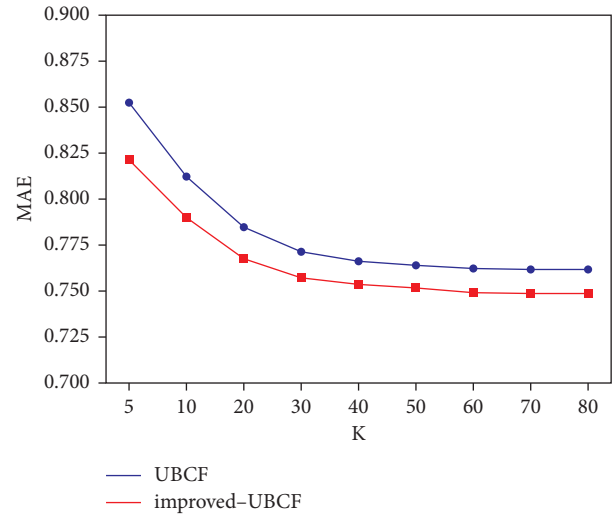


Figure 1: MAE before and after improvement of UBCF algorithm under different K values.

prediction score to fill the missing items in the score matrix, so that the data distribution becomes dense. The sparsity [2] is introduced to measure the sparsity of data, as shown in the following formula:

$$d = 1 - \frac{T}{|U| \cdot |I|}. \tag{16}$$

In formula (16), T represents the size of the initial dataset, and $|U|$ and $|I|$ represent the numbers of users and items, respectively.

In this paper, the sparsity of the dataset is 0.949 before filling, and it is 0.720 after filling, which is 5.5 times lower than that before filling, so the sparsity problem is solved effectively.

In order to verify the influence of solving the problem of data sparsity on the algorithm results, the improved UBCF algorithm is used to generate recommendations on the filled datasets. A comparison between the final optimization algorithm HCFDS and the improved UBCF algorithm is shown in Figure 3.

The improved UBCF algorithm in the figure is based on the unfilled original dataset and has the problem of data sparsity. The final optimization algorithm HCFDS is recommended based on the filled dataset, which solves the problem of data sparseness.

The overall accuracy of the HCFDS algorithm was improved significantly before the comparison. When K is smaller, that is, is 5, it increases by 7.9%. With the increase of K value, MAE tended to be stable. When K was 50, MAE increased by 1.6%. Therefore, the HCFDS algorithm proposed in this paper is effective and can improve the accuracy of recommendation.

3.3.4. Comparison of Different Optimized Recommendation Algorithms. To verify that the accuracy of HCFDS is better than that of the single optimization similarity algorithm or that of the matrix filling algorithm, the optimization

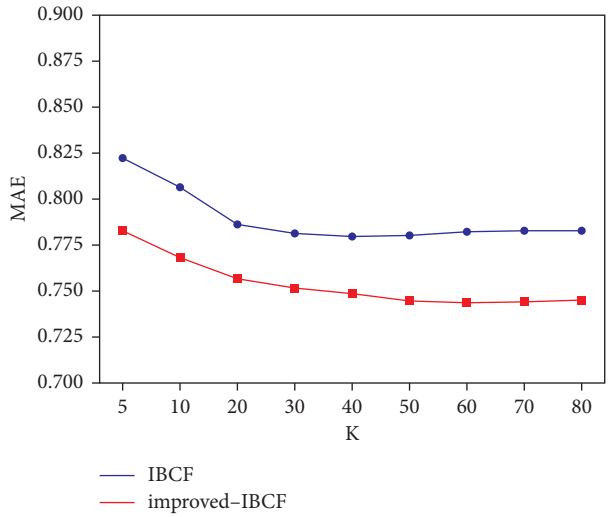


Figure 2: MAE before and after improvement of IBCF algorithm under different K values.

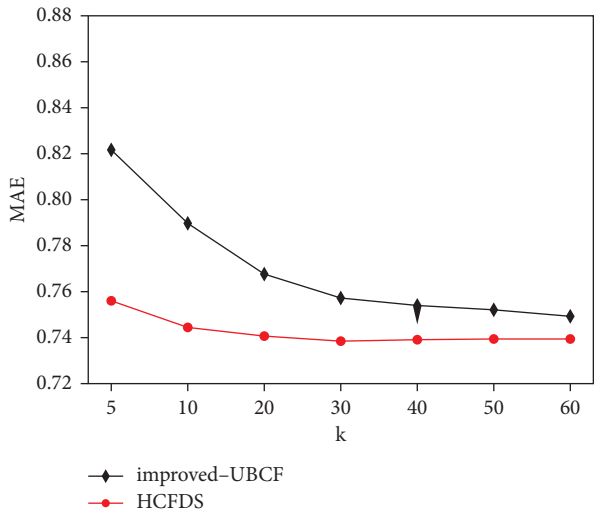


Figure 3: Comparison of HCFDS with modified UBCF under different K values.

algorithm HCFDS in this paper is compared with the optimization algorithm of Deng et al. [6], the optimization algorithm of support weight of Wang and Zheng. [3], and the optimization algorithm proposed by X. Gao et al. [8] based on cosine similarity, and the results are shown in Figure 4.

It can be seen from Figure 4 that the fluctuation of each optimization algorithm tends to be stable with the increase of K value, and the gap decreases. However, regardless of the value of K, the accuracy of the HCFDS algorithm suggested in this study is always superior to those of other algorithms. When K value is small, such as $K = 5$, it is 5.1% higher than Wang Wei, 5.5% higher than Deng et al., and 1.8% higher than X. Gao et al. With the increase of K value, MAE tends to be stable. when $K = 30$, it is 2.7% higher than Wang and Zheng, 3.9% higher than Deng et al., and 2.2% higher than X. Gao et al. Therefore, the HCFDS algorithm proposed in this paper has higher accuracy.

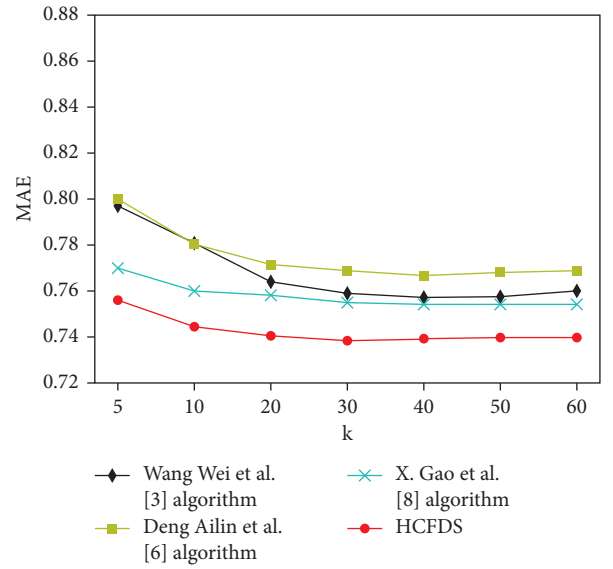


Figure 4: Comparison of different optimal recommendation algorithms under different K value.

4. Conclusion

In this paper, an improved hybrid collaborative filtering algorithm (HCFDS) is proposed to solve the problem of data sparsity in traditional collaborative filtering algorithms. To begin, in a sparse data environment, the user rating is normalized to unify the rating standards owing to differences in user rating standards and the rise in the weight of the same user rating in the overall rating. Following that, the IU-CS user similarity optimization method is presented, which incorporates the user rating difference threshold and the same rating penalty factor. In view of the problems that the difference of individual users' scores between items is too large when users' scoring data on items are few and the weight of the same scores of items increases in similarity contribution when users are few, this algorithm introduces item scoring difference threshold and penalty factor of the same scores of items and an item similarity optimization algorithm II-CS is proposed to correct the significant influence of the difference of individual users' or items' scores and the high proportion of the same scores in the total scores on similarity calculation under the condition of sparse data. Secondly, aiming at the problem of missing data in the scoring matrix, this study proposes a scoring prediction filling optimization method. In this method, aiming at the missing items in the scoring matrix, the II-CS algorithm proposed in this paper is used to preliminarily predict the score, and then the item category preference score is used to correct the predicted score, so that the filling value is reasonable. Finally, the algorithm uses the IU-CS algorithm proposed in this paper to make recommendations on the filled matrix. The aforementioned methods suggested in this study are evaluated on the public dataset MovieLens against the unoptimized algorithm and other algorithms. When K is 5, MAE is 7.9% greater than the modified UBCF algorithm, 5.1% higher than Wang and Zheng, 5.5% higher than Deng

et al., and 1.8% higher than X. Gao et al. MAE becomes more stable as the K value increases. When $K = 30$, it outperforms the modified UBCF algorithm by 2.4%, Wang and Zheng by 2.7%, Deng et al. by 3.9%, and X. Gao et al. by 2.2%. As a result, the HCFDS method presented in this work may considerably increase score prediction accuracy and successfully address the problem.

Data Availability

Previously reported MovieLens dataset data were used to support this study and are available at https://movielens.umn.edu. These prior studies (and datasets) are cited at relevant places within the text as references [9].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the https://doi.org/10.13039/501100001809National Natural Science Foundation of China (61876023).

References

[1] Z. Qu, J. Yao, X. Wang, and S. Yin, "Attribute weighting and samples sampling for collaborative filtering," in *Proceedings of the IEEE International Conference on Big Data & Smart Computing.*, January 2018.

[2] Y. Xi, T. Dan, S. Hongping, and A. Yiwen, "Improvement of collaborative filtering recommendation algorithm based on data sparsity," *Engineering Science and technology*, vol. 52, no. 1, 2020.

[3] W. Wang and J. Zheng, "Improvement of collaborative filtering algorithm based on user similarity," *Journal of East China Normal University*, no. 3, pp. 60–66, 2016.

[4] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

[5] C. Li and L. Ma, "Item-based collaborative filtering algorithm based on group weighted rating," in *Proceedings of the 2020 13th International Symposium on Computational Intelligence and Design (ISCID)*, pp. 114–117, Hangzhou, China, December 2020.

[6] A. Deng, Y. Zhu, and B. Shi, "Collaborative filtering recommendation algorithm based on item scoring prediction," *Journal of Software Engineering*, vol. 14, no. 09, pp. 1621–1628, 2003.

[7] R. Ji, Y. Tian, and M. Ma, "Collaborative filtering recommendation algorithm based on user characteristics," in *Proceedings of the 2020 5th International Conference on Control, Robotics and Cybernetics (CRC)*, pp. 56–60, Wuhan, China, October 2020.

[8] X. Gao, Z. Zhu, X. Hao, and H. Yu, "An effective collaborative filtering algorithm based on adjusted user-item rating matrix," in *Proceedings of the 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, pp. 693–696, Beijing, Chna, March 2017.

[9] H. Huo, P. Zhu, and H. Zhang, "Time context unifying collaborative filtering," in *Proceedings of the 2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, pp. 1274–1278, Chengdu, China, October 2016.

[10] N. Zhao, P. Wenchao, and C. Xu, "A video recommendation algorithm for multidimensional feature analysis filtering," *Computer Science*, vol. 47, no. 04, pp. 103–107, 2020.

[11] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Madison WI, USA, July 1998.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7715753, 10 pages
https://doi.org/10.1155/2022/7715753

Research Article

Image Anomaly Detection Based on Adaptive Iteration and Feature Extraction in Edge-Cloud IoT

Weiwei Zhang,[1] **Xinhua Tang**,[2] **and Jiwei Zhang**[3]

[1]*School of Science, Shandong Jianzhu University, Jinan 250101, China*
[2]*School of Cyberspace Security, Shandong University of Political Science and Law, Jinan 250014, China*
[3]*School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing 100876, China*

Correspondence should be addressed to Xinhua Tang; 000522@sdupsl.edu.cn

Received 4 November 2021; Revised 13 December 2021; Accepted 30 December 2021; Published 27 January 2022

Academic Editor: Yingjie Wang

The Internet of Things (IoT) has penetrated into various application fields. If the multimedia information obtained by the IoT device is tampered with, the subsequent information processing will be affected, resulting in an incorrect service and even security threat. Therefore, it is very necessary to study multimedia forensics technology for IoT security. In the edge-cloud IoT environment, an image anomaly detection technology for security service is proposed in this paper. First, preprocessing is performed before image anomaly detection. Then, we extracted sparse features from the image to roughly localize the region of anomaly detection. Feature extraction based on the polar cosine transform (PCT) is then performed only on the candidate region of anomaly detection. To further improve the detection accuracy, we use iterative updating. This method makes use of the feature that the edge node is closer to the multimedia source in physical location and migrates the complex computing task of image anomaly detection from the cloud computing center to the edge node. Provide a security service for abnormal data and deploy it to the edge-cloud server to reduce the pressure on the cloud. Overall, preprocessing improves the ability of feature extraction in smooth or small region of anomaly detections, and the iterative strategy enhances the security service. Experimental results demonstrate that the proposed method outperforms state-of-the-art methods.

1. Introduction

In recent years, with the continuous integration of emerging technologies such as artificial intelligence, blockchain [1], big data [2], and the Internet of Things (IoT) [2–7] and the increasing number of intelligent devices [8], the image data to be processed by the IoT has increased exponentially. IoT technology has penetrated into many fields, and its development has attracted extensive attention. A large number of multimedia data are generated in IoT. If these multimedia data are tampered with, it will threaten the information security and the Internet [9]. Therefore, the research of multimedia forensics is of great significance. Image forensics is an important branch of multimedia forensics. Aiming at the problems of high delay and low processing efficiency of edge cloud, an image anomaly detection method based on edge computing is proposed. Deploy the image security service task to the edge device closest to the image data to be processed to share the computing pressure of the cloud server.

The methods of image anomaly detection [10] can be divided into active methods and passive methods. Active methods are aimed at embedding useful information in an image and then verifying the authenticity and integrity of the image by evaluating the embedded information. However, conventional digital cameras lack digital watermarking functions for security. Consequently, active methods cannot be used when embedded information is unavailable. Alternatively, passive methods, also known as blind forensics, do not require preprocessing of digital images. Thus, it is used to identify the authenticity of images without embedded information, being more applicable than active methods. To conceal tampering and make the image visually more realistic, postprocessing can be applied to the cloned

area with methods such as rotation, loss JPEG compression, scaling, and other distortions.

Two main types of passive forensic algorithms are used. One is based on block matching, also known as dense-field algorithm, and the other is based on key points, also known as sparse-field algorithm. Dense-field algorithms usually divide an image into circular or square overlapping blocks to extract a feature vector from each block. After lexicographic sorting, the similarity between the successive vectors is evaluated, and the region of anomaly detection is determined by thresholding. Generally, dense-field algorithms have high computational complexity and may lead to false matching of similar smooth areas in natural images. On the other hand, sparse-field algorithms extract selected points, called key points, to generate feature descriptors. Key points have distinctive characteristics and can reflect essential characteristics of an image to identify target objects. However, sparse-field algorithms cannot extract enough key points from smooth or small areas in images, limiting their performance. In addition, the sparsity of key points impedes the accurate localization of duplicated areas.

To handle the abovementioned problems and leverage both dense-field and sparse-field algorithms, we propose an algorithm integrating these algorithms. First, the region of anomaly detection is roughly localized using a sparse-field algorithm, and then, a dense-field algorithm is applied to accurately determine the region of anomaly detection. Furthermore, we propose an adaptive iterative strategy to improve the localization accuracy. The main contributions of this study are summarized as follows:

(1) In the edge-cloud IoT, an anomaly detection technology for security service is proposed to further construct the trust mechanism of network data. This method makes use of the feature that the edge node is closer to the multimedia source in physical location and migrates the complex computing task of image anomaly detection from the cloud computing center to the edge node.

(2) The advantages of dense-field and sparse-field algorithms are combined in the proposed method. The proposed algorithm first obtains the approximate location of anomaly detection by sparse-field algorithm and then obtains the accurate location of anomaly detection by dense-field algorithm.

(3) An adaptive iterative strategy is introduced to improve the accuracy of tampering localization. Even if few matching points are available, the region of anomaly detection can be accurately determined

The remainder of this paper is organized as follows. Section 2 presents related work. In Section 3, we detail the proposed algorithm. Section 4 reports experimental results. Finally, we draw conclusions in Section 5.

2. Related Work

Edge-cloud calculation in IoT means processing data at the edge of the network. Edge computing may solve the problems of response time requirements, battery life constraints, and bandwidth cost savings and provide data security services [11]. Ferrari et al. used full-cloud and edge-cloud architectures for industrial IoT anomaly detection [12]. The results show that edge domain can reduce data transmission and communication delay. Feature extraction and feature matching are the bases in image anomaly detection [13]. In a dense-field algorithm, detection involves block feature extraction and feature matching across blocks [14]. The discrete cosine transform (DCT) was first proposed by Fridrich et al. [15]. However, the corresponding algorithm has high computational complexity and low robustness. Subsequent improvements to feature extraction measures have been proposed, such as principal component analysis (PCA) [16], singular value decomposition (SVD) [17], discrete wavelet transform (DWT) [18], blur-invariant moment features [19], and local binary patterns (LBP) [20]. Bayram et al. [21] extracted scale-invariant features from each block using the Fourier-Mellin transform (FMT). However, this algorithm is only robust for small region rotations. On the other hand, the Zernike moments (ZM) proposed by Ryu et al. [22, 23] and the polar cosine transform (PCT) proposed by Li [24] allow to extract robust rotation-invariant features from small overlapping blocks. For matching, lexicographic sorting is widely used [25]. To accelerate matching, k-dimensional trees [19] and locality-sensitive hashing [24] have been adopted to detect similar patches. However, these algorithms have high computational complexity because all image blocks should be matched. Recently, a fast approximate nearest neighbor search algorithm called Patch Match (PM), which is based on nearest neighbor search, was introduced [26, 27]. Regarding performance, sparse-field algorithms are faster than dense-field algorithms because the former should process fewer points. The scale-invariant feature transform (SIFT) was proposed by Lowe [28] in 1999. Luo et al. [29] extracted rotation and scale invariant descriptors. Subsequently, an accelerated version called speeded up robust features (SURF) was proposed [30]. Other fast feature detection and description algorithms include oriented features from accelerated segment test (FAST) and rotated binary robust independent elementary features (BRIEF) [31], multisupport region order-based gradient histogram [32], and histogram of oriented gradients.

In recent years, blockchain [33] and deep learning have been used for information protection [34, 35]. Fusion strategies based on SIFT have achieved suitable detection results [36–39]. In particular, the histogram of oriented gradients has been applied to feature extraction and tampering detection using a support vector machine (SVM) [36]. Nonoverlapping superpixel segmentation has been used as a preprocessing step before applying feature extraction [37]. Features have been extracted and matched in two different color spaces for rough detection [38], and DCT features have been extracted for accurate localization. Furthermore, key points have been detected using a uniqueness metric and described using PCT [39], with iterative improvement enabling accurate localization. Despite its advantages, SIFT has various drawbacks. Specifically, it cannot detect

tampering of smooth or small areas in an image. In addition, the sparsity of feature points provided by SIFT impedes to accurately locate the region of anomaly detection. We propose three strategies to overcome the limitations of this method. First, the target image is represented in the Lab color space in smooth areas. Second, rescaling is applied in small areas. Third, the localization accuracy is improved by combining dense-field and sparse-field algorithms.

3. Proposed Algorithm

IoT technology [40] has penetrated into many fields [41], and its development has attracted extensive attention [42]. Edge cloud is a cloud computing platform built on edge infrastructure based on the core and edge computing capabilities of cloud computing technology to form an elastic cloud platform with comprehensive capabilities in computing, network, storage, and security at the edge. The edge-cloud IoT architecture is shown in Figure 1. We can see that the edge cloud, central cloud, and IoT terminal in Figure 1 form an end-to-end "cloud three-body collaboration" technical framework. By placing tasks such as computing and intelligent data analysis at the edge, cloud pressure can be reduced. The image data generated by massive terminal devices are transmitted to the cloud computing layer [43, 44] for centralized processing through the network, which has the problems of large amount of calculation and large image processing delay. An image anomaly detection method for security service, which is based on edge calculation, is proposed in this paper. Taking advantage of the fact that the edge nodes are closer to the multimedia source in physical location, the complex image analysis and processing computing tasks are migrated from the cloud computing center to the edge computing layer.

We propose an iterative algorithm based on dense-field and sparse-field algorithms in edge-cloud IoT. First, SIFT is applied to roughly locate the region of anomaly detection. Then, PCT feature extraction is performed only on the candidate region of anomaly detection, and PM is used for matching. As SIFT may partially identify a region of anomaly detection, an adaptive iterative strategy is introduced to further improve the localization accuracy. Finally, after morphological operations, the region of anomaly detection is accurately localized.

The flowchart of the proposed algorithm is shown in Figure 2. The algorithm comprises a rough localization stage (including preprocessing) and an accurate localization stage. The following subsections detail each process in the proposed algorithm.

3.1. Image Preprocessing. Firstly, the image is preprocessed. The image analysis process does not need to transmit the image to the cloud through the network for processing but directly analyzes and processes the image in the edge server close to the data source. SIFT is a feature extraction and matching algorithm that provides higher accuracy and robustness to scaling attacks than similar algorithms such as SURF, BRIEF, oriented FAST, and rotated BRIEF. SIFT can extract key points on a spatial scale without being affected by illumination, affine transformations, noise, and other image factors such as corner points, edge points, bright spots in dark areas, and dark spots in bright areas. Based on these key points, feature descriptors of each key point are generated. Owing to its superior performance, we use SIFT for feature extraction in the rough localization stage.

A common preprocessing step before applying SIFT is representing the target RGB (red–green–blue) image in grayscale. However, detection often fails when using grayscale images, especially in smooth areas. To prevent this problem, channels a and b of the Lab color space, the grayscale image, and contrast limited adaptive histogram equalization have been used for preprocessing before feature extraction [38]. Reducing the contrast threshold and rescaling the image have also been used as preprocessing methods [45]. Although such preprocessing methods can increase the number of matching points, they apply various techniques simultaneously, resulting in a large computational overhead. Figure 3 gives an example using SIFT for two preprocessing methods. Figures 3(a)–3(c) gives the tampered image, the tampered image and the ground truth, separately. Figures 3(d) and 3(e) show key points extracted from the grayscale and Lab space (channel a), separately. We can see that the key points in Lab space are denser than those in grayscale. In contrast, after representing the RGB image in Lab color space, tampering could be detected using channel a, as shown in Figure 3(f). The Lab color space allows to extract more key points than the grayscale representation for smooth areas. Nevertheless, the grayscale representation is more robust than the Lab color space against various postprocessing attacks.

In the proposed algorithm, three preprocessing methods are used: (1) RGB-to-grayscale transformation, (2) RGB-to-Lab transformation, and (3) image resizing. However, if these methods are used simultaneously, the computational overhead would notably increase. Therefore, only when one preprocessing method fails, the next method is used, effectively reducing the calculation burden. On the other hand, the proposed algorithm does not require many matching points for rough localization. Thus, if three or more matching points are identified, accurate localization can proceed iteratively. The main preprocessing steps are described as follows.

Step 1. The RGB image is converted into a grayscale image, and feature matching is performed.

Step 2. If the security detection fails, the image is represented in the Lab color space for detection.

Step 3. Otherwise, the image is expanded to repeat detection. If the security detection fails after applying the three preprocessing methods, the image is considered as authentic and safe.

3.2. Rough Localization Stage. Rough localization stage mainly includes three processes: (1) feature extraction, (2) generalized two nearest neighbor matching, and (3) mismatch elimination by using random samples with invariant

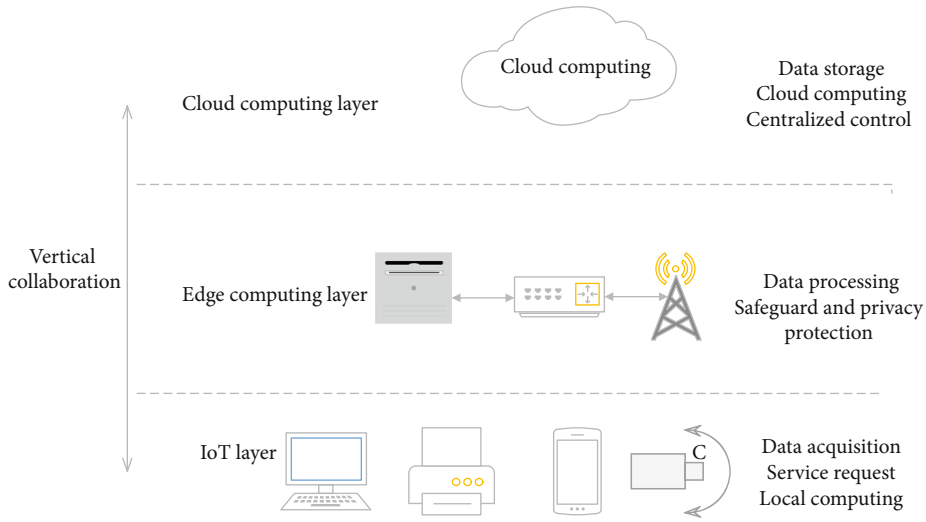


Figure 1: Edge-cloud IoT architecture.

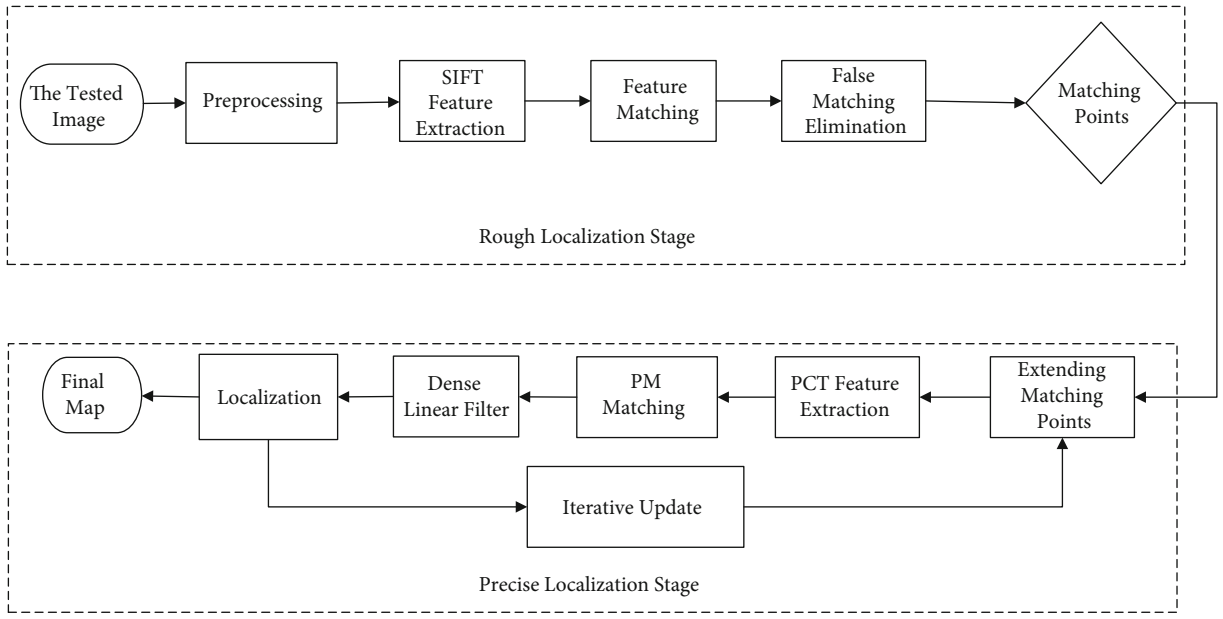


Figure 2: Flowchart of the proposed algorithm.

compatibility. We use the VLFeat open-source library [46] for feature extraction and description. After preprocessing, 128-dimensional SIFT features are extracted. We denote the key points as $x_i(i = 1,\cdots,n)$ and the feature descriptors as $f_i(i = 1,\cdots,n)$ for n feature points. Then, generalized two nearest neighbor matching is applied [47]. The Euclidean distance between a feature descriptor and the other descriptors is calculated. For example, we calculate the distance between f_1 and $f_2, f_3, \cdots, f_n$ and obtain distance vector $D = \{d_1, d_2,\cdots,d_{n-1}\}$ after sorting. If $d_k/d_{k+1} < T_{\text{thresh}}$ and $d_{k+1}/d_{k+2} \geq T_{\text{thresh}}$ for $k(1 \leq k \leq n-2)$, then feature point x_1 and the key points with distances of $\{d_1, d_2,\cdots,d_k\}$ from x_1 are considered to be matching. In this study, we set the threshold T_{thresh} to 0.05.

As many similar areas can appear in natural images, false matching should be prevented. To this end, we use agglomerative hierarchical clustering [47] to filter out classes with less than three points. Furthermore, we use robust random sample consensus to estimate homograph that allows to filter out the effects of unwanted outliers. When at least two classes are detected and at least three matched pairs between classes are available, we consider that the image is tampered.

The sparse-field algorithm can only provide an approximate location of the anomaly detection through the above-mentioned steps. For smooth or small region of anomaly detections, few matched points may be extracted, undermining the accuracy. As shown in Figure 3(f), after rough localization, only eight matching points are obtained, being

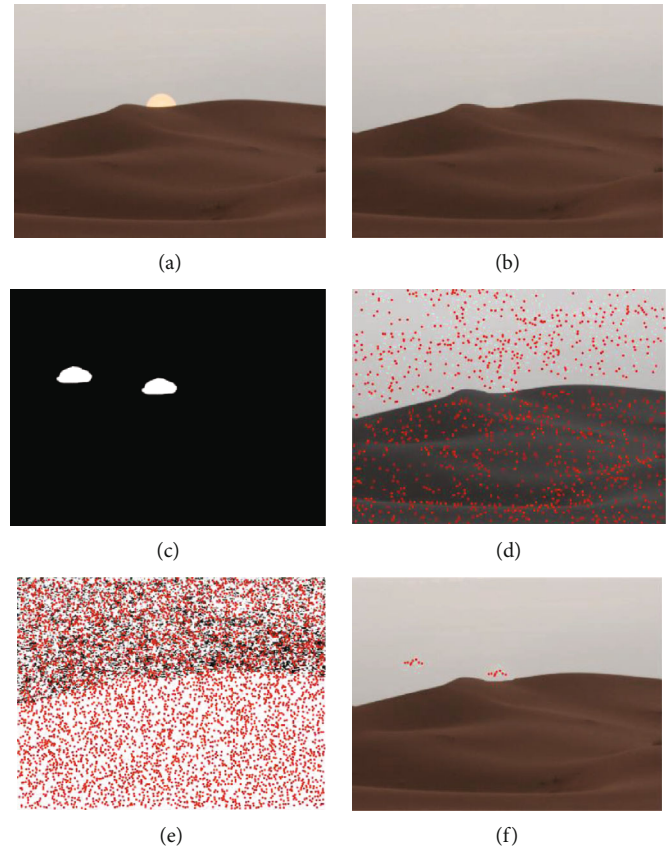


Figure 3: Key point extraction and matching for image represented in (a) the original image, (b) the tampered image, (c) the ground truth, (d) SIFT key point detection in grayscale, (e) SIFT key point detection in Lab color space (channel *a*), and (f) matched key points.

difficult to accurately determine the region of anomaly detection. Therefore, we use a dense-field algorithm and an iterative strategy for accurate localization in the following stage.

3.3. Accurate Localization Stage. To improve the localization accuracy, we use an iterative update strategy as described below.

Step 1. By centering at the matching points, the candidate tampering area (R) is expanded as follows:

$$R(x,y) = \begin{cases} 1 & \left\|(x,y) - x_j\right\| \le \dfrac{B}{2}, \quad \forall (x,y) \in I, \\ 0 & \text{otherwise}, \end{cases} \tag{1}$$

where $x_j (j = 1, \cdots, m)$ represents the matching points obtained during rough localization, I represents the target image, $B = 30 + [0.1\sqrt{M \times N}]$ is the expansion radius, and $M \times N$ is the size of the target image.

Step 2. Using R, block matching is used for accurate localization. Considering the powerful distinguishing performance of PCT, we use it to extract block features [24]. Specifically, 9-dimensional PCT block features are extracted from expanded matching area R. Let $f(r,\theta)$ denote the polar coordinates of the image. The PCT with order n and repetition l can be expressed as

$$M_{n,l} = \Omega_n \int_0^{2\pi} \int_0^1 [H_{n,l}(r,\theta)]^* f(r,\theta) r dr d\theta, \tag{2}$$

where $H_{n,l}(r,\theta) = \cos(\pi n r^2) e^{il\theta}$ is the kernel equation of PCT and

$$\Omega_n = \begin{cases} \dfrac{1}{\pi} & n = 0, \\ \dfrac{2}{\pi} & n \ne 0. \end{cases} \tag{3}$$

Then, the PCT feature vector can be calculated as

$$f = \left\{ \left| M_{n,l} \right| \middle| n + l \le 3, 0 \le n, l < 3 \right\}. \tag{4}$$

After PCT block feature extraction, PM [26] and dense linear filtering are applied for matching and filtering out mismatches, respectively. The PM algorithm proposed by Barnes et al. [26] is an approximate nearest neighbor search algorithm. The algorithm searches for similar image blocks globally in a single image through neighborhood search and random sampling. It mainly includes three steps:

Input:
I: the tested image;
T_{iter}: the maximum number of iteration;
T_{term}: algorithm termination threshold;
Obtaining the candidate tampering area $R(x, y)$ using formula (1);
while $i \le T_{\text{iter}}$ **do**
  $\text{map}^{(i)} \longleftarrow R(x, y)$ PCT feature extraction, PM matching and dense linear filtering;
  $\text{Cor\_map}^{(i)} \longleftarrow \text{map}^{(i)}$ corrosion;
  $\text{Exp\_map}^{(i)} \longleftarrow \text{Cor\_map}^{(i)}$ expansion;
  $\text{Dif\_map}^{(i)} \longleftarrow \text{Exp\_map}^{(i)} - \text{Cor\_map}^{(i)} > 0$;
  $\text{map\_new}^{(i)} \longleftarrow \text{Dif\_map}^{(i)}$ PCT feature extraction, PM matching and dense linear filtering;
  $\text{map}^{(i+1)} = \text{map}^{(i)} \cup \text{map\_new}^{(i)}$;
  $\text{map}^{(i+1)}$ morphology open operation;
  if $i \ge 2$ **then**
    $\nabla\text{map\_}F^{(i)} \longleftarrow \text{diff}(\text{map}^{(1)},\cdots,\text{map}^{(i)})$//calculate the first derivative;
    if $\nabla\text{map\_}F^{(i)} \le T_{\text{term}} | \, i \ge T_{\text{iter}}$ **then**
      break;
    end if
  end if
  $i = i + 1$;
end while
Output: the tamper localization $\text{map}^{(i)}$.

Algorithm 1: The proposed adaptive iterative algorithm.

random initialization, propagation, and random search. Filtering is mainly aimed at finding a dense approximate neighbor matching between image blocks through initialization, propagation, and random search. After this step, we obtain candidate region of anomaly detection $\text{map}^{(i)}$, where i is the number of iterations.

Step 3. To remove isolated small erroneous detections, corrosion is applied to $\text{map}^{(i)}$ with radius B, obtaining area $\text{Cor\_map}^{(i)}$ after corrosion.

Step 4. $\text{Cor\_map}^{(i)}$ is expanded with radius $(B + 10)$, obtaining area $\text{Exp\_map}^{(i)}$.

Step 5. Map $\text{dif\_map}^{(i)}$ is obtained as $(\text{Exp\_map}^{(i)} - \text{Cor\_ma p}^{(i)} > 0)$. The algorithm returns to Step 2 to obtain $\text{map\_ne w}^{(i)}$. Except for the first iteration, PCT feature matching is applied only to new area $\text{dif\_map}^{(i)}$ during any other iteration.

Step 6. The candidate region of anomaly detection is updated as $\text{map}^{(i+1)} = \text{map}^{(i)} \cup \text{map\_new}^{(i)}$ $(i \ge 1)$.

Step 7. The morphology open operation is applied to delete objects with area below T in $\text{map}^{(i+1)}$. In this study, we used eight neighborhoods and a minimum clone size T of 1200.

Step 8. The candidate region of anomaly detections obtained over iterations is denoted as $\text{map\_}F^i = \{\text{map}^{(1)},\cdots,\text{map}^{(i)}\}$ $(i \ge 2)$. Their first derivative is denoted as $\nabla\text{map\_}F^i = \text{diff}$ $(\text{map\_}F^i)$. If $\nabla\text{map\_}F^i(\text{end}) \le T_{term}$ (set to 500 in this study) or the number of iterations exceeds maximum limit T_{iter} (set to 5 in this study), the algorithm terminates. Otherwise, the algorithm returns to Step 3 to start a new iteration. The pseudocode is shown in Algorithm 1.

Table 1: F_1 score of various anomaly detection methods.

Study	Image level (%)	Pixel level (%)
Amerini et al. [48]	67	44
Li et al. [49]	86	85
Bravo-Solorio and Nandi [25]	94	85
Christlein et al. [50]	67	52
Tahaoglu et al. [38]	94	97
This study	96	97

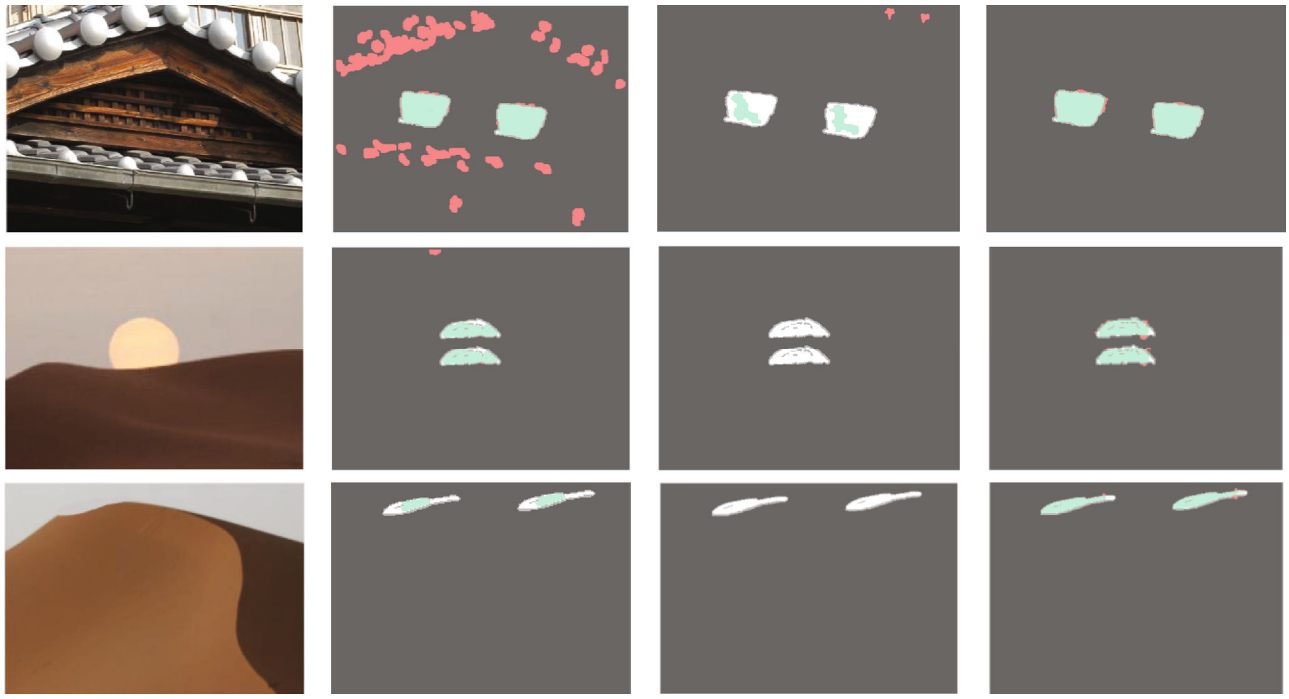
4. Experimental Results

We evaluated the performance of the proposed algorithm on the GRIP dataset [14]. This dataset contains 80 original images of 768×1024 pixels along with the corresponding copy-move forged images and ground truths. Most of the copies in this dataset are obtained from smooth areas. For the experiments, we used a computer equipped with a 2.60 GHz Intel(R) Core i7-9850H CPU running a MATLAB R2019a implementation.

4.1. Evaluation Criteria. We calculated the precision and recall at the image level and pixel level to evaluate the performance of the proposed anomaly detection algorithm:

(a) (b) (c)

Figure 4: Example of false matching elimination: (a) tampered image, (b) detection results from SIFT, and (c) false matching elimination.



(a) (b) (c) (d)

Figure 5: Anomaly detection results of different methods: (a) target image and detection results of (b) PM [27], (c) SIFT [39], and (d) the proposed algorithm.

$$\text{Precision} = \frac{T_p}{T_p + F_p}, \tag{5}$$

$$\text{Recall} = \frac{T_p}{T_p + F_N}, \tag{6}$$

where T_P is the number of tampered images in image level (or tampered pixels in pixel level) correctly detected, F_P is the number of original images in image level (or original pixels in pixel level) erroneously detected as tampered, and F_N is the number of tampered images in image level (or tampered pixels in pixel level) incorrectly detected as authentic.

The precision represents the accuracy of the predicted results, and the recall represents the accuracy of the total positive samples. Thus, higher precision and recall indicate a better algorithm. However, a low recall implies a high precision and vice versa. Thus, we used another comprehensive measure, the F_1 score, obtained as the harmonic mean of the precision and recall:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{7}$$

4.2. GRIP Dataset. Given an image, we need to determine the presence of tampering, in which case it becomes necessary to accurately localize the region of anomaly detection. We evaluated the proposed algorithm at the image level and pixel level separately. We combined 160 images, including 80 original images and 80 tampered images from the GRIP dataset. At the image level, we obtained precision of 93%, recall of 1, and F_1 score of 96%. At the pixel level, we obtained precision of 95%, recall of 99%, and F_1 score of 97%.

The F_1 score obtained from different methods are listed in Table 1. At the image level, the proposed algorithm provides the highest F_1 score. At the pixel level, the proposed

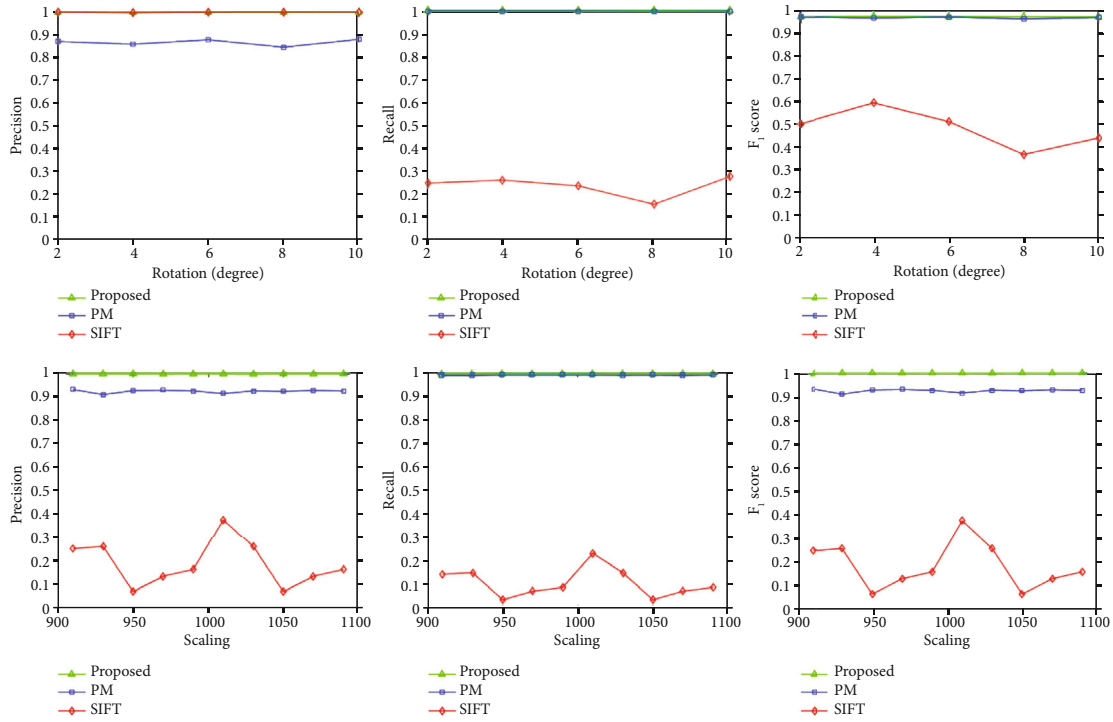


Figure 6: Detection result under rotation and scaling attacks.

algorithm has the same F_1 score as the method in Ref. [38] and higher F_1 score than the other methods. The proposed algorithm provides better detection because mismatched points obtained from rough localization are likely eliminated after accurate localization. Figure 4 shows an example of this situation. We tested the tampered image in Figure 4(a) at the image level. The detection results for SIFT matching are shown in Figure 4(b). The points enclosed by the red circle indicate SIFT mismatching, which is eliminated after PM matching, as shown in Figure 4(c).

Figure 5 shows examples of textured, mixed, and smooth region of anomaly detections. Figure 5(a) shows the forged images, and Figures 5(b)–5 (d) show the corresponding results for PM [27], SIFT [39], and the proposed algorithm, respectively. The red area in the detection result indicates false detection, while the white area indicates that tampering could not be detected, and the green area indicates correct detection. The remaining black areas represent areas that neither have been tampered with nor have been misdetected. The PM algorithm suitably detects tampering in smooth areas (second and third rows), but it provides false detection for the textured area (first row). SIFT fails to accurately localize the region of anomaly detection and is completely unable to detect tampering in the smooth area. In contrast, the proposed algorithm combining SIFT and PCT provides the best detection results.

4.3. FAU Dataset. We also used the public image dataset in Ref. [50] to test the performance of the proposed algorithm under rotation and scaling attacks in smoothed areas. In Figure 6, we tested 15 rotation or scaling images of smoothed area tampering. The first row shows rotation attacks from 2° to 10°, with step of 2°. The second row shows scaling attacks from 91% to 109%, with the step as 2%. We compare the proposed method with the state-of the art method: the SIFT-based method [39], indicated in red, and the PM-based method [27], indicated in blue. The results indicated in green are the detection result of the proposed method. We can see that the proposed scheme performed better than the other two methods in smoothed area tampering.

Table 2: Mean computation time of various anomaly detection methods.

Study	Mean computation time (s)
Tahaoglu et al. [38]	418
Zandi et al. [39]	437
This study	653

The computation time of the proposed algorithm and similar methods is listed in Table 2. By calculating 160 images in the GRIP dataset, the mean computation time of the proposed algorithm is slightly higher than that of the methods in Refs. [38, 39], but it remains within an acceptable range.

5. Conclusions

At present, in the image anomaly detection task of IoT, a large number of terminal devices transmit images to the cloud computing center through the network, resulting in large computing load and high image processing delay. In the edge-cloud IoT, a security service-oriented image anomaly detection technology is proposed in this paper. The RGB

image is represented in grayscale and channel *a* of the Lab color space, and it is resized for preprocessing. Then, SIFT feature extraction is applied. The preprocessing methods are not performed simultaneously, but each method is applied only if the preceding one cannot detect tampering, effectively reducing the computational overhead. SIFT feature matching then provides a rough localization of the anomaly detection, while PCT block feature extraction and PM feature matching provide accurate localization of the anomaly detection. An adaptive iterative update strategy is introduced to gradually improve the localization accuracy. The performance of the proposed algorithm was evaluated at the image and pixel levels. The experimental results show that migrating the image security service task to the edge computing device can reduce the pressure of the computing center, deal with the image data anomaly detection in time, and improve the image privacy and security. In the future, deep learning algorithms will be combined to improve the scope of application of image anomaly detection.

Data Availability

The datasets in the experiments include GRIP and FAU, which can be accessed in reference [14, 50], separately. [14]. D. Cozzolino, G. Poggi, and L. Verdoliva, "Copy-move forgery detection based on PatchMatch," in Proceedings of IEEE international conference of Image Process, pp. 5312-5316, Izmir, Turkey, 2014. [50]. V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou, "An evaluation of popular copy-move forgery detection approaches," IEEE Transactions on Information Forensics and Security, vol. 7, no. 6, pp. 1841-1854, 2012.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Doctoral Research Fund of Shandong Jianzhu University (No. X20022Z) and the Shandong Province Soft Science Research Project (Grant no. 2020RKB01671).

References

[1] Y. Xu, J. Ren, Y. Zhang, C. Zhang, B. Shen, and Y. Zhang, "Blockchain empowered arbitrable data auditing scheme for network storage as a service," *IEEE Transactions on Services Computing*, vol. 13, no. 2, pp. 289–300, 2020.

[2] X. Zhou, W. Liang, K. I. K. Wang et al., "Academic influence aware and multidimensional network analysis for research collaboration navigation based on scholarly big data," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 1, pp. 246–257, 2021.

[3] W. Liang, Y. Hu, X. Zhou, Y. Pan, and K. Wang, "Variational few-shot learning for microservice-oriented intrusion detection in distributed industrial IoT," *IEEE Transactions on Industrial Informatics*, 2021.

[4] X. Zhou, X. Yang, J. Ma, and K. Wang, "Energy efficient smart routing based on link correlation mining for wireless edge computing in IoT," *IEEE Internet of Things Journal*, vol. 99, 2021.

[5] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 144–153, Dallas, TX, USA, 2019.

[6] X. Zhou, X. Xu, W. Liang, Z. Zeng, and Z. Yan, "Deep-learning-enhanced multitarget detection for end-edge-cloud surveillance in smart IoT," *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 12588–12596, 2021.

[7] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[8] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–38, 2021.

[9] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[10] X. Zhou, W. Liang, S. Shimizu, J. Ma, and Q. Jin, "Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5790–5798, 2021.

[11] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[12] P. Ferrari, S. Rinaldi, E. Sisinni et al., "Performance evaluation of full-cloud and edge-cloud architectures for Industrial IoT anomaly detection based on deep learning," in *2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4.0&IoT)*, pp. 420–425, Naples, Italy, 2019.

[13] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[14] D. Cozzolino, G. Poggi, and L. Verdoliva, "Copy-move forgery detection based on PatchMatch," in *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 5312–5316, Izmir, Turkey, 2014.

[15] A. J. Fridrich, B. D. Soukal, and A. J. Lukáš, "Detection of copy-move forgery in digital images," *Proceedings of Digital Forensic Research Workshop*, , pp. 289–302, Springer-Verlag Press, Berlin, 2003.

[16] A. C. Popescu and H. Farid, *Exposing Digital Forgeries by Detecting Duplicated Image Regions*, Dartmouth Computer Science Technical Report TR2004-515, USA, 2004.

[17] X. Kang and S. Wei, "Identifying tampered regions using singular value decomposition in digital image forensics," in *International Conference on Computer Science and Software Engineering*, pp. 926–930, Wuhan, China, 2008.

[18] M. Bashar, K. Noda, N. Ohnishi, and K. Mori, "Exploring duplicated regions in natural images," *IEEE Transactions on Image Processing*, vol. 99, pp. 1–40, 2010.

[19] B. Mahdian and S. Saic, "Detection of copy-move forgery using a method based on blur moment invariants," *Forensic Science International*, vol. 171, no. 2-3, pp. 180–189, 2007.

[20] Y. Zhu, X. Shen, and H. Chen, "Covert copy-move forgery detection based on color LBP," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 43, no. 3, pp. 390–397, 2017.

[21] S. Bayram, H. T. Sencar, and N. Memon, "An efficient and robust method for detecting copy-move forgery," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1053–1056, Taipei, Taiwan, 2009.

[22] S. J. Ryu, M. J. Lee, and H. K. Lee, "Detection of copy-rotate-move forgery using Zernike moments," in *International Workshop on Information Hiding*, pp. 51–65, Springer, 2010.

[23] S. J. Ryu, M. Kirchner, M. J. Lee, and H. K. Lee, "Rotation invariant localization of duplicated image regions based on Zernike moments," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 8, pp. 1355–1370, 2013.

[24] Y. Li, "Image copy-move forgery detection based on polar cosine transform and approximate nearest neighbor searching," *Forensic Science International*, vol. 224, no. 1-3, pp. 59–67, 2013.

[25] S. Bravo-Solorio and A. K. Nandi, "Exposing duplicated regions affected by reflection, rotation and scaling," *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 1880–1883, Prague, Czech Republic, 2011.

[26] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "PatchMatch," *ACM Transactions on Graphics*, vol. 28, no. 3, pp. 1–11, 2009.

[27] D. Cozzolino, G. Poggi, and L. Verdoliva, "Efficient dense-field copy-move forgery detection," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 11, pp. 2284–2297, 2015.

[28] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, pp. 1150–1157, Kerkyra, Greece, 1999.

[29] W. Luo, J. Huang, and G. Qiu, "Robust detection of region-duplication forgery in digital image," in *18th International Conference on Pattern Recognition (ICPR'06)*, pp. 746–749, Hong Kong, 2006.

[30] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: speeded up robust features," in *Computer Vision-ECCV*, pp. 404–417, Springer, Graz, Austria, 2006.

[31] Y. Zhu, X. Shen, and H. Chen, "Copy-move forgery detection based on scaled ORB," *Multimedia Tools and Applications*, vol. 75, no. 6, pp. 3221–3233, 2016.

[32] O. Miksik and K. Mikolajczyk, "Evaluation of local detectors and descriptors for fast feature matching," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pp. 2681–2684, Tsukuba, Japan, 2012.

[33] Y. Xu, C. Zhang, G. Wang, Z. Qin, and Q. Zeng, "A blockchain-enabled deduplicatable data auditing mechanism for network storage services," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1421–1432, 2021.

[34] X. Yan, B. Cui, Y. Xu, P. Shi, and Z. Wang, "A method of information protection for collaborative deep learning under GAN model attack," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 3, pp. 871–881, 2021.

[35] X. Zhou, Y. Li, and W. Liang, "CNN-RNN based intelligent recommendation for online medical pre-diagnosis support," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 3, pp. 912–921, 2021.

[36] A. Parashar, A. K. Upadhyay, and K. Gupta, "An effectual classification approach to detect copy-move forgery using support vector machines," *Multimedia Tools and Applications*, vol. 78, no. 20, pp. 29413–29429, 2019.

[37] C. Pun, X. Yuan, and X. Bi, "Image forgery detection using adaptive over-segmentation and feature points matching," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 8, pp. 1705–1716, 2015.

[38] G. Tahaoglu, G. Ulutas, B. Ustubioglu, and V. V. Nabiyev, "Improved copy move forgery detection method via L∗a∗b∗ color space and enhanced localization technique," *Multimedia Tools and Applications*, vol. 80, no. 15, pp. 23419–23456, 2021.

[39] M. Zandi, A. Mahmoudi-Aznaveh, and A. Talebpour, "Iterative copy-move forgery detection based on a new interest point detector," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 11, pp. 2499–2512, 2016.

[40] Y. Xu, J. Ren, G. Wang, C. Zhang, J. Yang, and Y. Zhang, "A blockchain-based nonrepudiation network computing service scheme for industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3632–3641, 2019.

[41] Y. Xu, Z. Liu, C. Zhang, J. Ren, Y. Zhang, and X. Shen, "Blockchain-based trustworthy energy dispatching approach for high renewable energy penetrated power systems," *IEEE Internet of Things Journal*, 2021.

[42] X. Yan, Y. Xu, X. Xing, B. Cui, and T. Guo, "Trustworthy network anomaly detection based on an adaptive learning rate and momentum in IIoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6182–6192, 2020.

[43] C. Zhang, Y. Xu, Y. Hu, J. Wu, J. Ren, and Y. Zhang, "A blockchain-based multi-cloud storage data auditing scheme to locate faults," *IEEE Transactions on Cloud Computing*, 2021.

[44] Y. Xu, Q. Zeng, G. Wang, C. Zhang, J. Ren, and Y. Zhang, "An efficient privacy-enhanced attribute-based access control mechanism," *Concurrency and Computation Practice and Experience*, vol. 32, no. 5, 2020.

[45] Y. Li and J. Zhou, "Fast and effective image copy-move forgery detection via hierarchical feature point matching," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1307–1322, 2019.

[46] A. Vedaldi and B. Fulkerson, "Vlfeat: an open and portable library of computer vision algorithms," in *International Conference on Multimedia*, pp. 1469–1472, Firenze, Italy, 2010.

[47] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra, "A SIFT-based forensic method for copy-move attack detection and transformation recovery," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 1099–1110, 2011.

[48] I. Amerini, L. Ballan, R. Caldelli, A. del Bimbo, L. del Tongo, and G. Serra, "Copy-move forgery detection and localization by means of robust clustering with J-linkage," *Signal Processing: Image Communication*, vol. 28, no. 6, pp. 659–1669, 2013.

[49] Jian Li, Xiaolong Li, Bin Yang, and Xingming Sun, "Segmentation-based image copy-move forgery detection scheme," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 507–518, 2015.

[50] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou, "An evaluation of popular copy-move forgery detection approaches," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 6, pp. 1841–1854, 2012.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 2372748, 11 pages
https://doi.org/10.1155/2022/2372748

Research Article

Short-Term Solar Irradiance Prediction Based on Multichannel LSTM Neural Networks Using Edge-Based IoT System

Maozheng Pi,[1] Ning Jin,[1] Dongxiao Chen,[1] and Bing Lou[2]

[1]*Key Laboratory of Electromagnetic Wave Information Technology and Metrology of Zhejiang Province, College of Information Engineering, China Jiliang University, Hangzhou, China 310018*
[2]*Zhejiang Huayun Information Technology Co. Ltd, Hangzhou, China*

Correspondence should be addressed to Dongxiao Chen; chendx@cjlu.edu.cn

Received 12 November 2021; Revised 10 December 2021; Accepted 27 December 2021; Published 24 January 2022

Academic Editor: Yingjie Wang

Most photovoltaic power generation methods use global level irradiance (GHI) as the main input and output. However, randomness, instability, and intermittency are the main factors that seriously degrade the solar irradiance prediction results. Traditional data-driven prediction models are difficult for accurate predictions. In this study, a multichannel deep learning model named multichannel, wavelet transform combining convolutional neural network and bidirectional long short-term memory (MC-WT-CBiLSTM) framework-based edge computing and IoT system is proposed to improve the GHI prediction accuracy. The solar irradiance data is decomposed by wavelet transform to reduce data complexity. Each decomposed component is inputted into the multichannel MC-CBiLSTM deep learning framework for forecasting and combined to produce the final results. The comparison with existing solar irradiance forecasting methods shows that the proposed MC-WT-CBiLSTM deep learning framework has obvious advantages in the prediction of various time horizons.

1. Introduction

As one of the green, clean, and sustainable energy, solar energy accounts for an increasing proportion of the current world energy structure. Accurate and reliable solar irradiance prediction brings significant benefits to the construction of modern smart grids [1–3]. For effective design and control of the photovoltaic (PV) energy, it is necessary to accurately predict solar irradiance in advance [4]. PV power generation and irradiance are positively correlated. However, the irradiance is affected by various external factors such as temperature, weather, and seasonality. These complex factors interact with each other to make GHI change irregularly, making the prediction difficult for traditional methods [5].

Data-driven prediction and forecasting methods, including machine learning (ML), edge computing (EC), and internet of things (IoT), are crucial methods in the field and important for the operation and dispatch of Industry 4.0 [6–9]. A recent study shows that ML, EC, and IoT methods have significant advantages in irradiance data prediction compared with physics-based models. Precise irradiance prediction provides the approximation of the expected PV output power for the dispatching plan of the grid company's operators [10].

The recent development of various DL methods seriously influences the issues of time-series data analysis and forecasting [11–15]. Long short-term memory (LSTM) has been widely applied to different time-series data analysis fields, including air quality prediction [16], short-term load prediction [17], irradiance prediction [18], and cyber-physics systems [19, 20]. While the original RNN cannot consider the dependence between long-term sequences, causing problems such as gradient disappearance or gradient explosion, LSTM cleverly solves the problems of gradient disappearance and gradient explosion by increasing the selectivity of the unique gating unit structure control information [21].

Difficulties exist for traditional LSTM neural networks for the solar irradiance forecasting problem. First of all, the irradiance data presents high volatility with weather changes, which are difficult to accurately capture by the

neural network. Secondly, a variety of external factors such as temperature, wind speed, and cloud density may have a certain impact on the irradiance prediction. Therefore, it is necessary to consider the mutual influence of multiple feature data to make more accurate predictions. For these neural network methods based on historical data, in order to make more accurate irradiance predictions, the structural complexity of the neural network must be increased. While remembering the characteristics of the longer-term sequence, the mutual influence between variables should also be considered [22].

Taking into account the shortcomings in the current studies in the field, this paper proposes a multichannel, wavelet transform combining convolutional neural network and bidirectional long short-term memory (MC-WT-CBiLSTM) framework for solar irradiance forecasting. BiLSTM is improved from LSTM by combining an LSTM moving from the beginning of the sequence and an LSTM moving from the end of the sequence to the beginning of the sequence [23]. In addition to the BiLSTM model, a one-dimensional convolutional neural network (CNN) is also used to further extract data features. Wavelet transform (WT) is introduced to decompose the original input data into multiple subsequences with different frequencies. Then, each subsequence is individually connected to a CNN-BiLSTM module for short-term GHI prediction. Experimental results show that wavelet decomposition can effectively reduce data complexity and improve prediction performance. At the same time, BiLSTM combined with CNN learns more sequence features from different dimensions and improves the prediction performance. According to experiments, the proposed MC-WT-CBiLSTM depth model framework has the following advantages over the existing methods:

(i) *A data preprocessing step with wavelet transform.* As GHI data is affected by many factors, the solar irradiance data fluctuates greatly. The wavelet transform preprocessing step effectively reduces the data complexity and improves the prediction ability of the multichannel CNN-BiLSTM model

(ii) *A sophisticated multi-input multichannel network structure*. The proposed framework takes the mutual influence of temperature factors and GHI data into consideration and proposes to use multiple channels for parallel learning

(iii) *A deep network framework integrating CNN and BiLSTM*. To the best of our knowledge, it is the first time that WT-CBiLSTM is combined with multichannel ideas for GHI prediction. Comparative experiments show that the prediction performance of the framework is due to the current advanced prediction methods

2. Related Works

The problem of time-series data prediction has always been one of the important topics in the field of artificial intelligence (AI). A lot of work has been done in the field of time-series forecasting. LSTM is one of the most popular deep learning models. Compared with traditional neural networks, the unique gated unit structure enables LSTM to remember information for a longer period of time [24–27]. Wen et al. [28] implemented the LSTM model to predict photovoltaic power generation and power load. The prediction performance of the proposed LSTM neural network is significantly better than the ML model. Yan et al. [29, 30] proposed a hybrid deep learning neural network framework that combines LSTM neural network and CNN to solve the problem of single household electricity consumption prediction. The use of CNN adds a preprocessing stage and extends the traditional LSTM neural network. Combined with CNN's LSTM can predict sequence changes more accurately. This research proves the advantage of one-dimensional convolution in processing time-series data. Zhou et al. [31] proposed an LSTM model combined with an attention mechanism to predict photovoltaic power generation. Taking into account the impact of temperature data on photovoltaic power generation, the attention mechanism adaptively focuses on more important input features, and the prediction effect is better than the comparison model of each time field of view. A large number of prediction studies have proved that a variety of data preprocessing strategies have greatly improved the prediction ability of the neural network model. Zheng et al. [32] proposed a hybrid deep learning model that combines empirical model decomposition (EMD) and LSTM to decompose the original data into multiple intrinsic mode functions (IMF) for better predictive analysis. It can be known from the research results that the decomposition of the waveform has a good effect on the prediction of time series. Wu et al. [33] realized singular value decomposition, reconstructed the original cutting force signal of the tool, and then used BiLSTM to predict the feature subsignal, thereby effectively improving the prediction accuracy.

While irradiance forecasting has received increasing number of attentions, people have adopted a variety of forecasting methods for irradiance forecasting. In [34], Yan et al. added the Inception-ResNet network for feature extraction and then input the extracted features into the GRU-Attention network for training prediction. The fusion of complex structures increased the complexity of the network. Zhao et al. [35] proposed 3D-CNN to perform feature analysis on ground cloud images for irradiance prediction and achieved very good prediction results.

The surveyed works show that for the nonlinearity and instability of the current time-series data, adopting a variety of data preprocessing strategies can effectively improve the prediction performance of the neural network model [36, 37]. The multichannel complex neural network fusion model proposed in this paper shows the effectiveness of predicting unstable irradiance data. Different from the conventional stacked CNN-LSTM, in the proposed hybrid model, CNN and LSTM extracted features in parallel, which results in more robust features with less loss in terms of data information. In [38], a multichannel DL framework was proposed for electrical load time-series prediction. The

framework consists of two parallel channels and a feature fusion module. One of the channels is composed of the CNN layer, and the other is the LSTM layer. These two channels are connected in the feature fusion module, and then, the final output is set. The final prediction result is better than most deep models.

3. Methodology

The experimental flowchart of the proposed MC-WT-CBiLSTM model is shown in Figure 1. The features used to predict GHI include irradiance and temperature data. After normalizing each feature, a three-layer wavelet transform is performed separately to reduce the complexity of the input data to obtain a more predictable subsequence. The subsequence is trained by the proposed MC-CBiLSTM framework, and the final prediction result is obtained. The experiment uses five evaluation indicators to evaluate the predictive performance of the proposed model.

3.1. Data Source and Preprocessing. The data used in this article comes from a comprehensive set of solar irradiance, imaging, and prediction data released by Pedro et al. [39] in 2019. The data includes three-year (2014-2016) quality control, 1-minute resolution global level irradiance, and direct ground measurement of normal irradiance in California. In addition, it also provides overlapping data from commonly used exogenous variables, including sky images, satellite images, and numerical weather forecast predictions. The experimenter selects global level irradiance and temperature data. The data for the three years from 2014 to 2016 are selected according to the training set, and the test set ratio is 4 : 1. The experiment chooses the z-score normalization method to preprocess all input data, and the calculation formula is as follows:

$$X^* = \frac{(X - \mu)}{\sigma}, \tag{1}$$

where μ is the average of all sample data and σ is the standard deviation of all sample data.

3.2. Wavelet Transform. Due to the severe volatility of the original GHI data set, this paper proposes WT's data processing method to decompose the original solar irradiance series data into multiple subsequences of different frequencies. These subsequences include a stable part (low-frequency signal) and a fluctuating part (high-frequency signal). These decomposed subsequences have better behavior in terms of rules. The wavelet transform decomposes the input data into multiple subcomponents, reducing the complexity and nonlinearity of the input data. These relatively stable simple subsequences are more stable, which is conducive to model training.

Generally speaking, the irradiance sequence data always presents high volatility, variability, and randomness due to its correlation with nonstationary weather conditions. Therefore, the original solar irradiance sequence may include nonlinear and dynamic components in the form of spikes and fluctuations [39]. WT is a decomposition method of discrete sampling of the input sequence. The key advantage of WT over Fourier transform is that WT can simultaneously capture frequency and position information (position in time). In addition, it is also good at multiscale information processing [40]. These advantages make WT an effective tool for complex data sequence analysis.

The main feature of wavelet transform is that the transformation can fully highlight the characteristics of certain aspects of the problem, localized analysis of time (space) and frequency, and gradually multiscale refinement of the signal (function) through the expansion and translation operation and finally reach the high-frequency time subdivision and low-frequency subdivision, which can automatically adapt to the requirements of time-frequency signal analysis, so that you can focus on any details of the signal. CWT is to select a center frequency and then obtain a large number of center frequencies through scale transformation and then obtain a series of basic functions in different intervals through time shift and then integrate the products of a certain segment of the original signal (corresponding to the interval of the basis function), respectively, and the result is the frequency corresponding to the extreme value is the frequency contained in this interval of the original signal. Since CWT requires a continuous signal, but the actual sampled signal is often discrete, we cannot directly perform CWT on the actual signal. In order to perform wavelet transformation on the irradiance sequence, the discrete wavelet transform (DWT) needs to be introduced. The discrete wavelet transform is obtained by discretizing the scale and displacement of the continuous wavelet transform according to the power of 2. The characteristics of the irradiance time series determine that the discrete wavelet transform is more suitable for decomposition.

There are many types of wavelet basis functions, such as Hear wavelet, Symlet wavelet, and *dbN* wavelet. In this study, wavelet transform (WT) with *db1* wavelet basis function is implemented to decompose the original data into multiple subsignals, including denoising low-frequency components and denoising high-frequency components. The decomposition evidently improves the learning ability of the subsequent neural network models. Wavelet transform is a localized analysis of time and frequency. It gradually refines the sequence in multiple scales through the expansion and translation operation. It can automatically adapt to the requirements of time-frequency sequence analysis, subdividing time at high frequencies and subdividing frequencies at low frequencies. In this way, the time-frequency variation characteristics of the irradiance time series are analyzed.

Given a mother wavelet function $\psi(t)$ and its corresponding reduced order function $\varphi(t)$, calculate the wavelet $\psi j, k(t)$ and the binary reduced order function $\phi_{j,k}(t)$:

$$\begin{aligned} \psi_{j,k}(t) &= 2^{\frac{j}{2}}\psi\left(2^j t - k\right), \\ \varphi_{j,k}(t) &= 2^{\frac{j}{2}}\varphi\left(2^j t - k\right), \end{aligned} \tag{2}$$

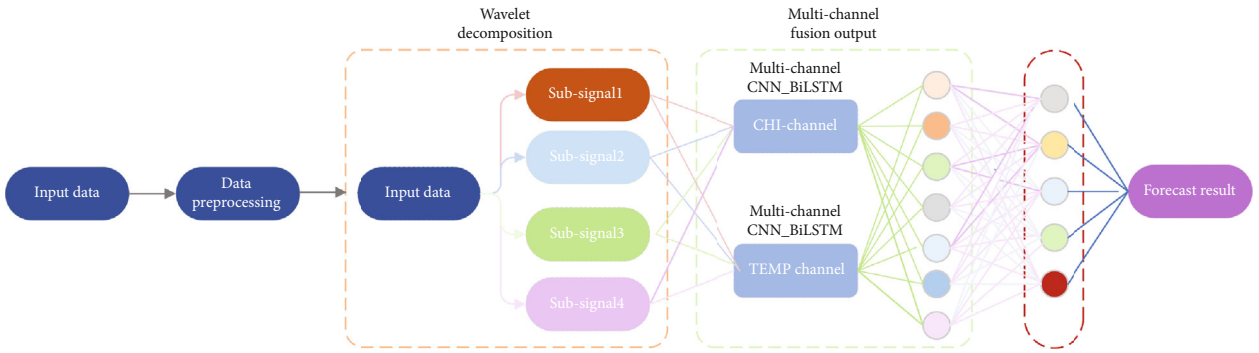


Figure 1: The overall flow chart of all proposed GHI prediction frameworks.

where t represents the time index, j represents the zoom-in variable, and k represents the translation variable. After the original sequence is decomposed n times, multiple components are obtained:

$$y(t) = A_{nt} + \sum_{i=1}^{n} D_{it}. \tag{3}$$

Through multiple decompositions, the low-frequency component A_{nt} is decomposed into the next layer of low-frequency components $A_{(n+1)t}$ and high-frequency components $D_{(n+1)t}$. The WT level in this paper is three. The original data is decomposed into $A3$, $D1$, $D2$, and $D3$. The decomposition sequence is directly input into the model framework for training. The wavelet decomposition process is shown in Figure 2.

3.3. Convolutional Neural Network. CNN is an emerging branch of DL. Different from traditional ways of feature extractions, CNN automatically generates useful and discerning features from raw data. This efficient feature extraction feature has been widely used in image recognition, speech recognition, and natural language processing [40].

Each subsequence decomposed from the original solar irradiance data set sequence is a one-dimensional sequence. A one-dimensional CNN is used as a local feature extractor. CNN adds a preprocessing stage and extends the BiLSTM neural network. In the processing stage, useful features are extracted from the original data, which improves the accuracy of subsequent predictions.

CNN can recognize simple patterns in data well and then use them to form more complex patterns in higher layers. One-dimensional CNN obtains more detailed features from a shorter (fixed-length) segment of the overall irradiance data set, and the position of the feature in the sequence segment is not correlated; the one-dimensional CNN will be very effective. In this paper, CNN is used to extract the features of each subsequence of wavelet transform, which further optimizes the learning of data features and facilitates the improvement of the prediction accuracy of subsequent neural network models.

3.4. Bidirectional Long Short Memory Neural Network (BiLSTM). The long-term short-term memory (LSTM) model is a special form of recurrent neural network (RNN) that provides feedback on each neuron. The unique gating unit solves the problem of gradient disappearance and gradient explosion when RNN processes long sequences. In the traditional RNN model and the long-term memory recurrent neural network (LSTM) model, information can only be propagated forward. This makes the current sequence state of the model only relate to the previous state. The bidirectional LSTM is an extension of the traditional LSTM, which combines two sets of LSTM in an opposite manner. This two-way structure facilitates simultaneous learning of forward and reverse sequence information, making the prediction results more integrated. BiLSTM not only considers the before and after correlation of the sequence but also solves the problem of prediction lag that may exist in one-way LSTM. The structure of BiLSTM is shown in Figure 3.

Since GHI data fluctuates significantly over time, the characteristics of the data before and after are closely related. The BiLSTM model is selected to predict the irradiance data, combined with the before and after correlation of GHI. Relying on this two-way characteristic, more detailed data characteristics are obtained. BiLSTM effectively improves the prediction accuracy of GHI.

The final prediction output is determined by the two values of the hidden layer of the bidirectional network. The formulas for the gating units of the BiLSTM model are as follows:

$$\begin{aligned} i_t &= \sigma_t(x_t W_{xi} + h_{t-1} W_{hi} + b_i), \\ f_t &= \sigma_f\left(x_t W_{xf} + h_{t-1} W_{hf} + b_f\right), \\ c_t &= f_t \odot c_{t-1} + i_t \odot \sigma_c(x_t W_{xc} + h_{t-1} W_{hc} + b_c), \\ \sigma_t &= \sigma_o(x_t W_{xo} + h_{t-1} W_{ho} + b_o), \\ h_t &= o_t \odot \sigma_h(c_t). \end{aligned} \tag{4}$$

3.5. MC-WT-CBiLSTM Frame Structure. The proposed MC-WT-CBiLSTM deep neural network framework is introduced in this subsection. Considering the internal correlation with temperature factors, temperature data is selected as an additional input feature. The overall flowchart of the

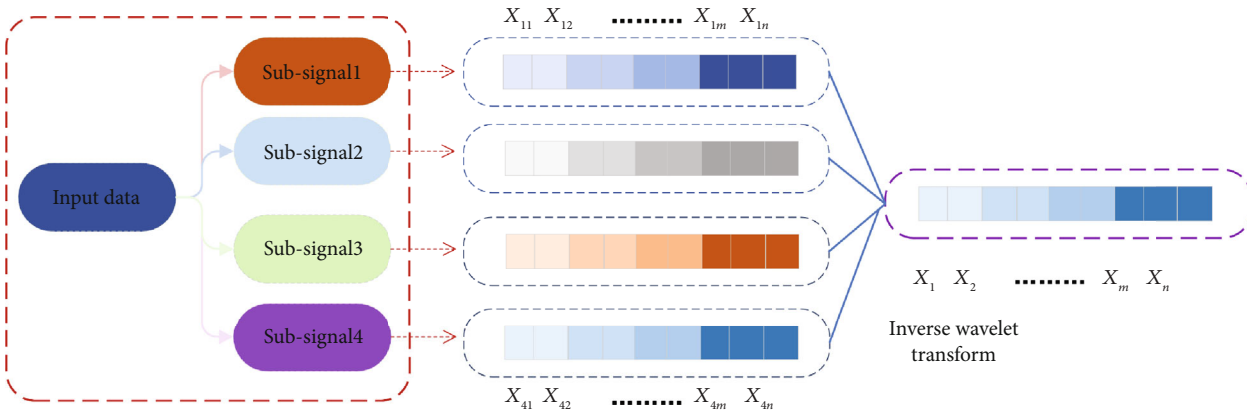


FIGURE 2: Schematic diagram of wavelet transforms. The left side is the sample raw data, and the right side is the decomposed subsequence. From top to bottom are *A*3, *D*1, *D*2, and *D*3.

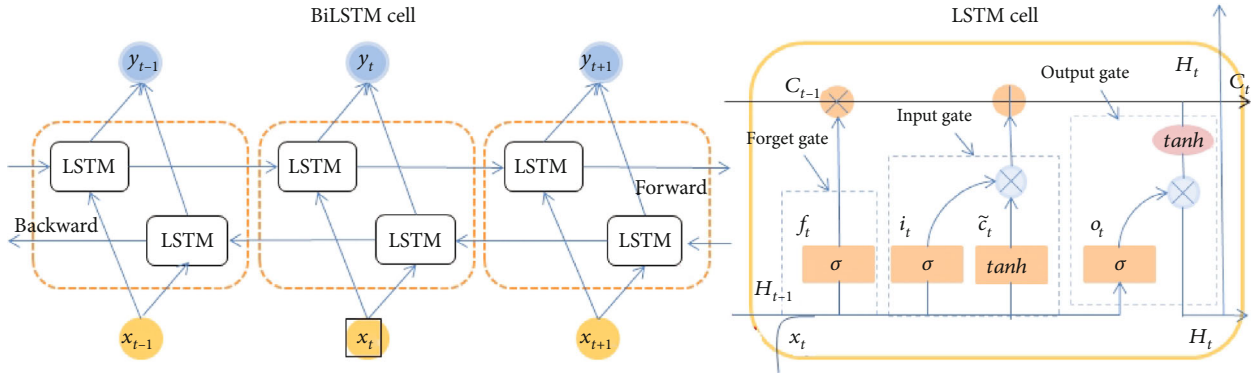


FIGURE 3: BiLSTM structure and its internal LSTM cell unit structure.

proposed framework is shown in Figure 4. Each input sequence is decomposed into multiple subsequences using WT. Then, each subsequence is inputted into the MC-CBiLSTM framework. Each subsequence is individually connected to a CNN-BiLSTM channel, and the channel parameters are adjusted according to the complexity of the subsequence to achieve the best prediction effect. The input GHI and temperature data are learned separately in two parallel channels. Each channel is connected by a feature fusion layer. In the feature fusion layer, the feature information of each channel is shared, and the prediction results are output together. Experimental results show that the output of GHI is affected by the temperature component. In view of actual experience, it is known that the irradiance and temperature do have certain internal influences. The interaction between the two can achieve more accurate prediction results than single-sequence prediction.

In Figure 4, the multichannel training layer is divided into GHI channels and TEMP channels, and the subsequence data after wavelet transformation are input, respectively. Each sequence is individually input to a CNN_BiLSTM model. Taking into account the internal correlation between components, the correlation effect may improve the accuracy of GHI prediction. One-dimensional CNN is designed for local feature extraction to improve prediction accuracy. For different input features, the number of filters can be flexibly adjusted to achieve the best feature extraction effect. The RMSprop optimizer is used to minimize the mean square error (MSE) loss function. The forecast steps are 10 minutes, 30 minutes, and 60 minutes. The neural network model was trained for 16 iterations. The BiLSTM unit of each channel has 100, 64, 64, and 32 neural units, respectively. The remaining hyperparameters include activation = "linear," validation_split = 0.05. Each channel is connected to the feature fusion layer for information sharing and finally undergoes wavelet inverse transformation to obtain the final prediction result.

The proposed MC-WT-CBiLSTM multichannel deep network framework consists of two parallel input channels, and two input features are trained separately. With edge computing solutions, input channels can be placed in different positions. For each input feature channel, four subchannels are connected, and the subchannels are used to train the subsignals after wavelet decomposition. Each subchannel consists of a CNN-BiLSTM layer, a feature fusion layer, and an output layer. The four subsequences after wavelet decomposition are input into one subchannel, respectively, and the CNN and LSTM parameters of each subchannel are different. The purpose is to train the model from different depths and finally perform overall prediction through feature fusion.

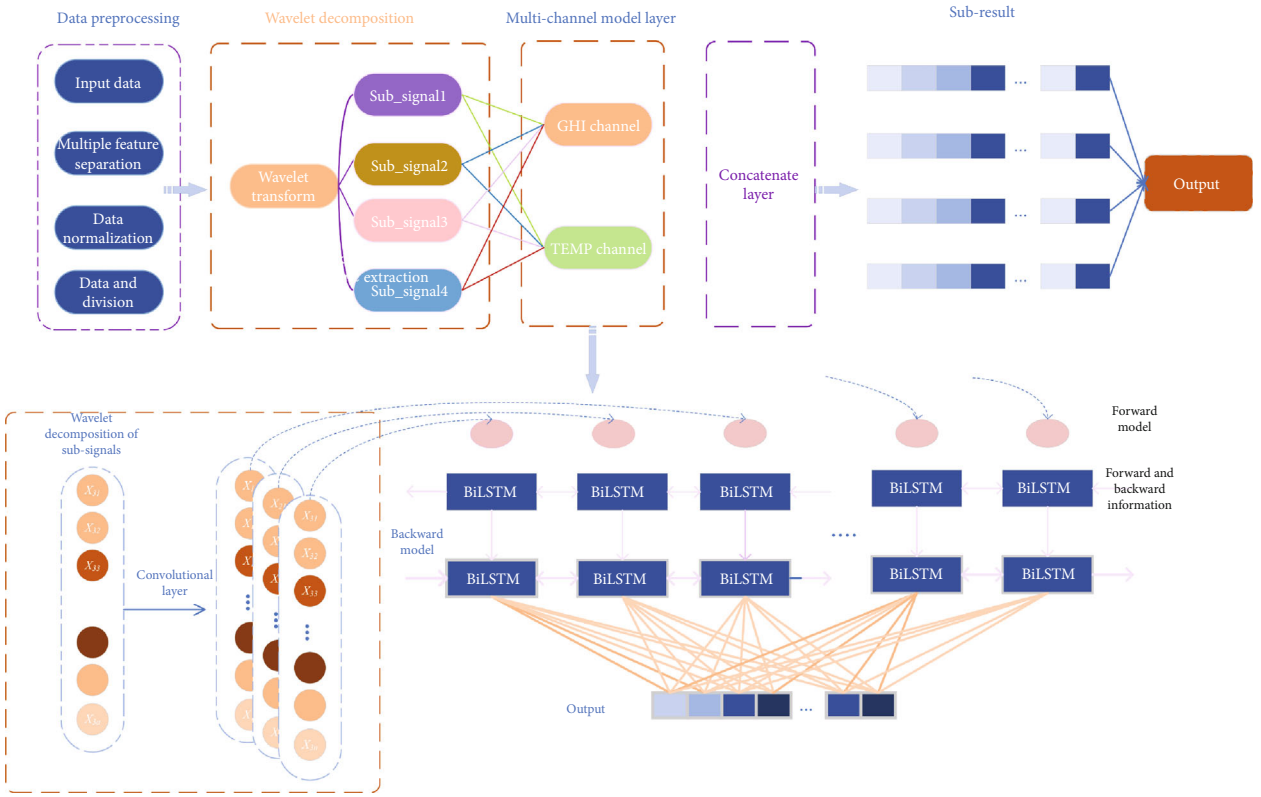


Figure 4: The proposed MC-WT-CBiLSTM overall framework flow chart.

Compared with the existing methods, the proposed framework not only considers the internal influence of temperature factors and GHI data but also is equipped with different channels, and multiple channels are connected for parallel learning of decomposed subsequences. Compare this multichannel model with a single-channel model. It can learn the characteristics of the input sequence in more detail. Compared with the existing single-channel model, the time dependence between features can be captured more accurately, and the decomposition of the input signal enables the framework to understand data fluctuations in more detail. The effective local feature extraction ability of one-dimensional convolution will further improve the predictive ability of the model. In some cases, BiLSTM considers the overall correlation of the sequence and is more suitable for predicting irradiance data, such as periodic fluctuations, than traditional LSTM.

4. Results

4.1. Evaluation Metrics. In this experiment, five error evaluation indexes of absolute error (MAE), root mean square error (RMSE), average absolute percentage error (MAPE), coefficient of determination (R^2), and symmetric average absolute percentage error (SMAPE) are selected to evaluate the accuracy of prediction. The specific formulas of the 5 indicators are as follows:

$$\begin{aligned}
\mathrm{RMSE}(y_i,\widehat{y}_i) &= \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y\wedge_i - y_i)^2},\\
\mathrm{MAE}(y_i,\widehat{y}_i) &= \frac{1}{n}\sum_{i=1}^{n}|\widehat{y}_i - y_i|,\\
\mathrm{MAPE}(y_i,\widehat{y}_i) &= \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\widehat{y}_i - y_i}{y_i}\right|,\\
\mathrm{SMAPE}(y_i,\widehat{y}_i) &= \frac{100\%}{N}\sum_{i=1}^{n}\frac{|\widehat{y}_i - y_i|}{(|\widehat{y}_i| + |y_i|)/2},\\
R^2(y_i,\widehat{y}_i) &= 1 - \frac{\sum_{i=1}^{n}(y_i - y\wedge_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}.
\end{aligned} \tag{5}$$

4.2. Input Feature Selection. In the experimental preprocessing stage, the original data is normalized. In the decomposition prediction stage, the processed data is decomposed into four subsignals by wavelet transform and input the proposed MC-CBiLSTM model for prediction. The prediction results are analyzed with five evaluation indicators: MAE, RMSE, MAPE, SMAPE, and R^2.

The data set selected in this paper contains multiple sets of features such as temperature and wind speed. The multichannel and multifeature prediction model proposed in this

Table 1: Single feature data prediction result evaluation index.

	10 min					30 min					60 min				
	MAE	RMSE	R^2	SMAPE	MAPE	MAE	RMSE	R^2	SMAPE	MAPE	MAE	RMSE	R^2	SMAPE	MAPE
Bagging	30.1	68.24	0.942	13.74	19.32	106.2	132.9	0.9	22.4	38.28	46.28	90.2	0.896	23.8	75.1
MLP	30.9	66.7	0.943	15.82	44.02	44.15	97.23	0.91	27.5	59.46	47.47	88.4	0.902	24.4	62.2
LSTM	33	68.25	0.943	18.61	24.89	60.4	97.14	0.895	27.6	57.13	127.4	97.1	0.895	27.6	57.13
BiLSTM	34.1	68.25	0.944	19.06	28.67	51.54	91.1	0.901	24.7	64.12	51.54	91.1	0.901	24.7	64.12
CNN-LSTM	32.6	67.65	0.944	15.15	22.72	51.14	81.11	0.861	30	53.38	51.14	81.1	0.861	30	53.38
CNN-BiLSTM	30.1	66.62	0.946	15.14	18.72	48.04	82.74	0.912	26	49.69	48.04	82.7	0.912	26	49.69
WT-LSTM	22.9	31.24	0.987	8.78	11.83	30.79	40.71	0.982	13.7	25.13	24.57	35.7	0.985	15.7	26.11
WT-BiLSTM	16.6	23.17	0.994	9.33	11.39	19.46	27.66	0.991	12.8	25.47	25.05	34.1	0.985	12.1	24.16
Proposed	10.1	13	0.998	8.94	9.3	11.38	17.86	0.996	10.8	15.24	29.48	38.4	0.98	13.1	21.82

paper inputs different features in different channels to improve the prediction accuracy. Consider the internal correlation between features. Through experiments, the selection of temperature characteristics effectively improves the prediction accuracy of GHI. In order to verify the influence of multidimensional features on the prediction results, the experiment selects a single GHI data for experimentation. The experimental results are shown in Table 1, comparing the single feature (GHI) prediction results of three-time intervals. Each model in the table uses a single GHI data for training.

4.3. Comparative Experiment. In order to further verify the prediction performance of the proposed MC-WT-CBiLSTM model, a variety of existing prediction models were selected for comparative research. For comparison experiments, more advanced machine learning models and deep fusion models in the field of time-series forecasting were selected. In this article, experiments are conducted in time steps of 10 minutes, 30 minutes, and 60 minutes. This experiment selects five evaluation indicators to evaluate the prediction results and compare the prediction performance of various models. The machine learning models to be compared include Bagging and MLP. In order to further verify the advantages of the MC-WT-CBiLSTM fusion framework proposed in this paper, deep learning models such as LSTM, BiLSTM, CNN-LSTM, CNN-BiLSTM, WT-LSTM, and WT-BiLSTM are selected for comparison. The prediction performance of the three-time interval model: the evaluation results are shown in Table 2. Each model in the table is trained using GHI and TEMP feature data.

Comparing the prediction performance index tables of the three time periods, it shows that the prediction performance of each model decreases significantly as the time interval increases. The prediction results show that the MC-WT-CBiLSTM model proposed in this paper still maintains good prediction performance. Compared with machine learning models, machine learning may be better than some deep learning models in short-term predictions such as 10 min predictions. However, as the time interval increases, the performance of machine learning prediction decreases significantly. This article carried out multiple sets of comparative experiments. It can be seen from the experimental results. The prediction results of LSTM or BiLSTM alone are poor, because the network structure is relatively simple and cannot learn more detailed features. The feature extraction ability of CNN can improve the learning ability of the model to a certain extent, but it has limited processing ability for complex data volatility. At the same time, the wavelet transform is added to reduce the complexity of the irradiance data. The results show that the wavelet has a great improvement in the predictive ability of the neural network. This article starts from multiple angles. On the one hand, wavelet transform is introduced to reduce the data complexity, and on the other hand, CNN is introduced for feature extraction. The results show that CNN and wavelet transform alone have certain limitations, and the combination of the two can more effectively improve the prediction accuracy.

The prediction results of the proposed MC-WT-CBiLSTM depth model and multiple comparison models are shown in Figure 5. Based on the last year's full-year data microtest set, the following picture shows the forecast results of the four seasons of spring, summer, autumn, and winter. It can be seen from the prediction effect graph that the proposed model has a good learning ability against various fluctuations of GHI data and has a better learning ability than other models. Figures 5(a) and 5(d) show the 10-minute time interval forecast. Due to the short time interval and the relatively smooth GHI data, all models have achieved good prediction results.

However, most model predictions generally have a certain lag. The highest and lowest points of the irradiance data cannot be accurately fitted. And the prediction result graph shows that the model after adding the waveform decomposition can capture more fluctuation information. From the fitting curve in the figure, the prediction effect of each model can be observed more intuitively. Only LSTM and BiLSTM have the worst fitting results. Compared with a single neural network, the prediction effect of CNN-LSTM and CNN-BiLSTM has been improved to a certain extent, but it still falls short of expectations. Due to complex data fluctuations, the neural network cannot learn accurate information, so wavelet transform is introduced for this purpose. The ability of wavelet transforms to reduce the complexity of the frequency domain effectively reduces the learning difficulty of neural networks. But

Table 2: Multifeature input model prediction result evaluation index.

	10 min					30 min					60 min				
	MAE	RMSE	R^2	SMAPE	MAPE	MAE	RMSE	R^2	SMAPE	MAPE	MAE	RMSE	R^2	SMAPE	MAPE
Bagging	29.33	68.9	0.942	12.42	17.04	42	84.42	0.9	20.62	36	48.63	90.83	0.9	23.8	75.1
MLP	28.28	65.4	0.948	15.82	24.92	41	79.91	0.91	24.46	53	47.47	88.36	0.9	24.4	62.2
LSTM	33.04	68.3	0.943	18.61	24.89	51	83.77	0.902	25.02	47	54.31	92.63	0.89	26.3	77.27
BiLSTM	34.14	68.3	0.944	19.06	28.67	46	81.04	0.908	25.85	47	56.99	90.91	0.89	27.4	57.34
CNN-LSTM	32.58	67.7	0.944	15.15	22.72	44	78.48	0.914	23.38	45	57.62	89.98	0.9	27	48.82
CNN-BiLSTM	30.11	66.6	0.946	15.14	18.72	46	79.82	0.911	25.61	42	48.04	82.74	0.91	26	49.69
WT-LSTM	11.67	15.8	0.996	6.72	9.62	18	24.61	0.991	12.32	16	23.2	33.76	0.99	13.3	17.25
WT-BiLSTM	11.26	15.7	0.996	8.12	10.51	17	22.45	0.992	15.49	22	25.05	34.08	0.99	12.1	24.16
Proposed	7.66	10.5	0.998	6.13	8.54	9.2	13.98	0.997	8.71	9.5	18.13	27.98	0.99	10.97	15.63

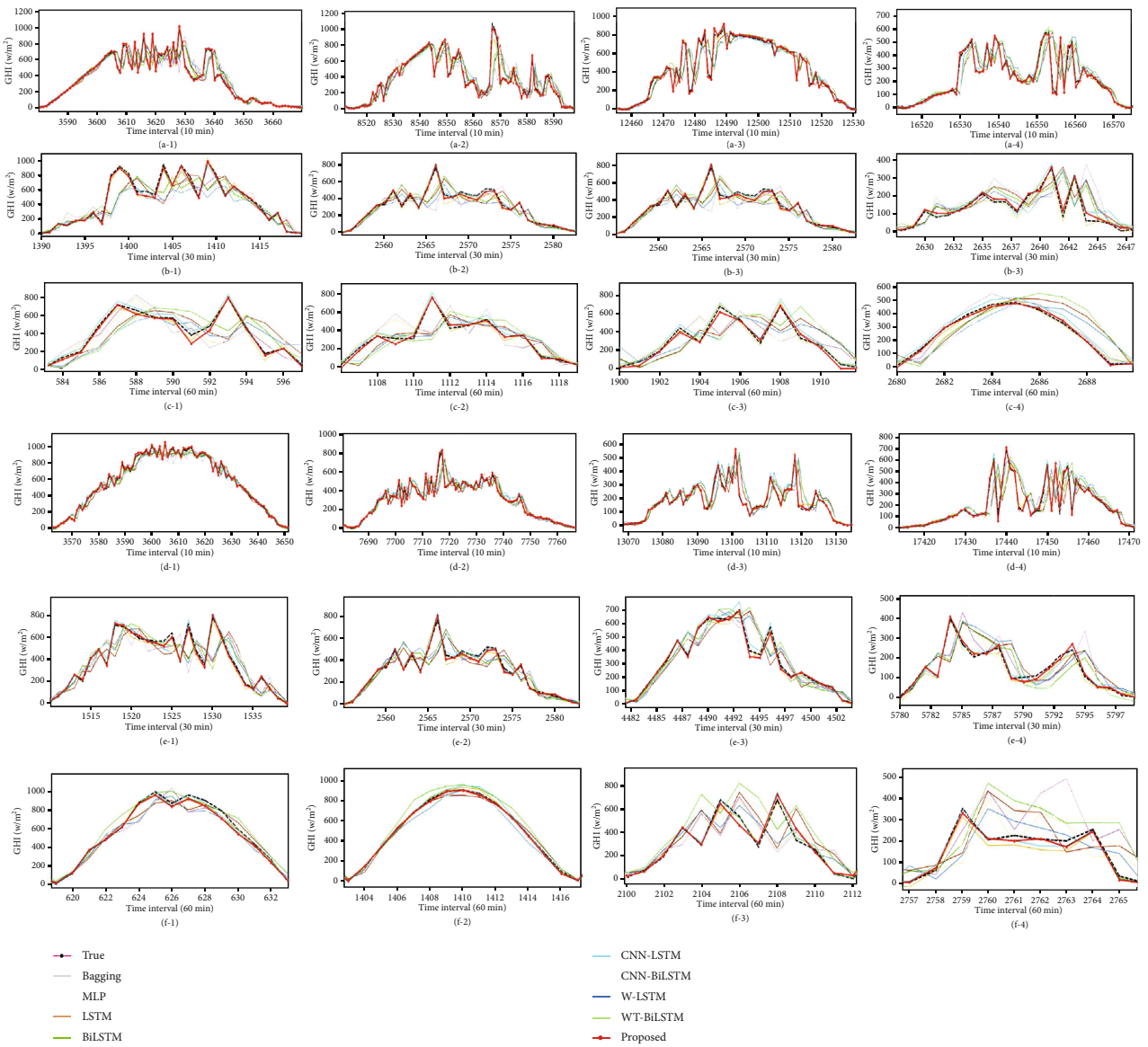


Figure 5: Single-feature and multifeature prediction effect diagram. Among them, (a), (b), and (c) are single feature 10 min, 30 min, and 60 min predictive effect. Contains the prediction results of the four seasons of spring, summer, autumn, and winter. (d), (e), and (f) are multifeature prediction results, which also contain four seasonal prediction results.

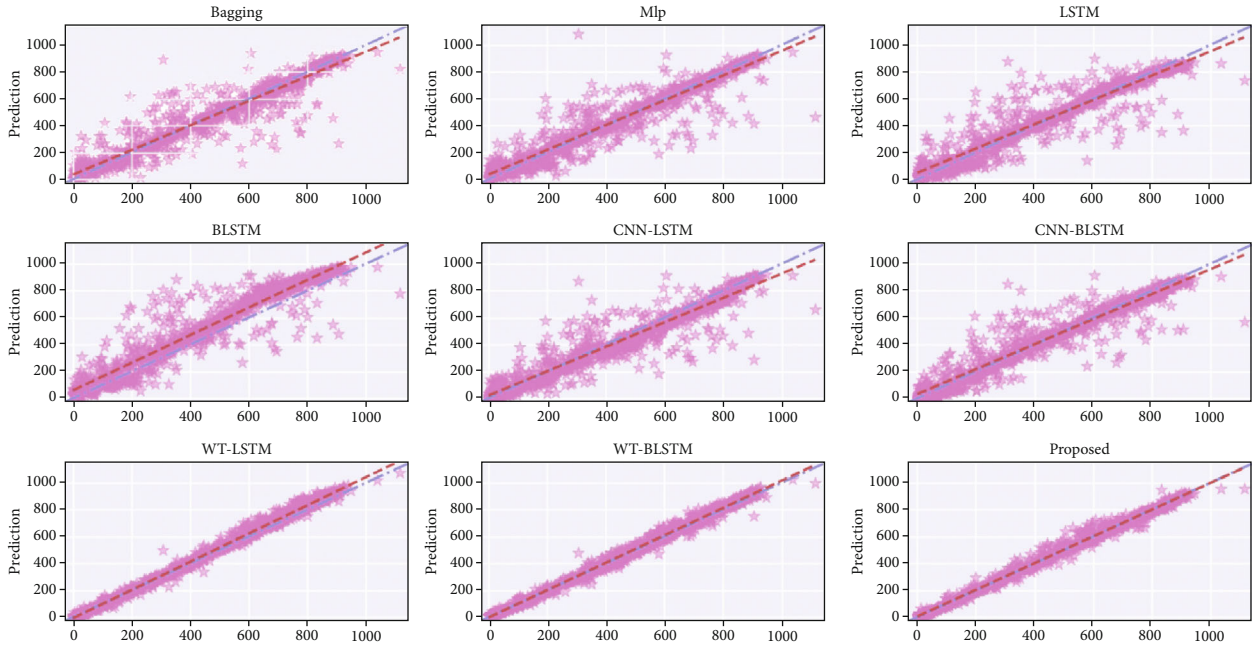


Figure 6: The rectangular coordinate distribution diagram of the prediction results of each model.

compared with WT-BiLSTM and WT-LSTM, the multifeature channel model proposed in this paper further improves the prediction ability. Although wavelet transform is used to reduce the complexity of data, for the irradiance sequence, the randomness of weather changes increases the complexity of the sequence. For single-channel neural networks, it is more difficult to learn sequence features in a single channel, which increases the difficulty of neural network training. The different channels of the multichannel BiLSTM model are trained at the same time, which reduces the difficulty of neural network training. BiLSTM neurons with different numbers of multiple channels are used to obtain sequence information of multiple depths and finally be fused. Obviously, more feature information can be obtained, and the prediction result is more accurate. Figures (b) and (d) show the 30-minute interval forecast. The fitting curve in the figure shows that as the time interval increases, the prediction effect becomes significantly worse. There are certain errors in the prediction of peaks and valleys. Figures (e) and (f) show the prediction results of 60 minutes, and the prediction effects of all prediction models have been reduced. However, the framework proposed in this paper still has high predictive power. The model accurately captures data fluctuations over multiple time periods. However, other relatively simple comparison models cannot capture too much fluctuation information when the time interval increases, and the prediction effect is poor. The six-day forecast results of the last month of 2016 selected in Figure 5 show that the MC-WT-CBiLSTM proposed in this paper can better fit the original GHI data. In order to see the prediction performance of each model more clearly, the prediction effect in the blue dashed box in the figure has been partially enlarged.

Figure 5 shows that the model proposed in this paper has significant advantages whether it is the overall prediction effect or the partial detailed prediction effect. The model proposed in this paper accurately predicts the fluctuation of data in three time periods. The prediction effect of each model shows that adding a series of data processing strategies to the irradiance prediction can effectively improve the prediction accuracy. For example, in the comparison model in this article, the fusion of CNN or WT obtains more accurate prediction results than the traditional single model.

Both the evaluation index and the fitting effect diagram prove the superiority of the model proposed in this paper. It can be seen from the fitting results in the figure that a single LSTM and BiLSTM model has certain difficulties in processing such complex irradiance data. This is because a single neural network cannot learn more in-depth data features, and at the same time, the neural structure is simple, and there is a certain performance bottleneck in the prediction of complex data. And from the results, most of the excellent prediction performance is due to the parallel learning of multiple channels. Multiple channels learn features of different depths. Compared with single-channel learning, deeper learning features can make more accurate predictions. At the same time, the bidirectional learning ability of BiLSTM enables the model to learn sequence features from two directions. In some of these specific scenarios, such as the irradiance sequence, BiLSTM is more practical than LSTM in this case due to the front-to-back correlation. Wavelet decomposition has also made a great contribution, and its ability to reduce data complexity can improve the predictive ability of neural networks. But the disadvantage is that the decomposition of the waveform makes the amount of training data extremely large, and the training time is significantly increased.

A rectangular graph of the 60 min prediction is shown for the further evaluation of the prediction performance of the MC-WT-BiLSTM model (Figure 6). The horizontal axis in the figure represents the real data, and the vertical axis

represents the predicted value of each model. The blue line in the figure represents the best fitting effect of 100% perfect prediction under ideal conditions. The red line indicates the approximate fitting effect of the model predicted value. The closer the blue line is to the red line, the better the forecasting effect. The prediction and fitting effect of each model is shown in the figure. It is obvious that the model proposed in this paper is closer to the ideal value. It is concluded from the distribution of the forecast data in the figure that the distribution of the forecast results of the model proposed in this paper is closer to the ideal straight line. This tightly distributed data indicates that the predicted result is closer to the true value.

5. Conclusion and Discussion

Solar irradiance prediction adopting AI and IoT technologies is of great importance for smart grid and city designs. In this study, considering the nonstationary and nonlinearity of GHI data, a multichannel multimodel fusion framework MC-WT-BiLSTM is proposed on the edge for accurate and effective solar irradiance prediction using cutting-edge edge computing and IoT technologies. The most advanced DL technology was adopted. A multichannel hybrid network model combining CNN and BiLSTM is proposed. The wavelet decomposition strategy is selected to process the irradiance data. The experiment utilizes a comprehensive solar irradiance data released by Pedro et al. in 2019. A comprehensive comparison with a variety of advanced depth models proves the effectiveness of the MC-WT-CBiLSTM model. Through comparison and prediction of multiple time intervals, it is evident that the proposed DL model has the most superior performance over the existing approaches. The fitting effect diagram in Figure 6 shows that the prediction method proposed in this article has a smaller prediction error. The results of various comparative experiments show that the various methods combined with the MC-WT-CBiLSTM model have the effect of improving the prediction ability.

The experiment takes into account the internal correlation between temperature data and GHI. At the same time, multichannel parallel learning enables the model to learn more data features. Summarizing the forecasting method of this article draws the following conclusion. First of all, for complex and nonstationary data, the waveform decomposition strategy is an effective way to reduce the complexity of the data. Moreover, one-dimensional convolution has excellent feature extraction capabilities and can achieve good feature extraction effects in the prediction of time-series data with greater volatility. As a variant of LSTM, BiLSTM is widely used in the field of NLP, mainly due to its bidirectional learning ability. For irradiance data with certain periodicity, it has an excellent predictive effect.

A future working direction of this study is to add more features to make more complex predictions. At the same time, the generalization ability of most of the current forecasting methods in the literature is poor, and only good results can be achieved in a small range. The next work is to improve the model in this paper and improve its generalization to be applied to more time-series forecasting fields.

Data Availability

The data used in this study is confidential.

Conflicts of Interest

The authors declare that there is no competing interest.

Acknowledgments

This study is fully supported by the Key Laboratory of Electromagnetic Wave Information Technology and Metrology of Zhejiang Province, College of Information Engineering, China Jiliang University, Hangzhou, China.

References

[1] F. Wang, Z. Zhen, Z. Mi, H. Sun, S. Su, and G. Yang, "Solar irradiance feature extraction and support vector machines based weather status pattern recognition model for short-term photovoltaic power forecasting," *Energy and Buildings*, vol. 86, pp. 427–438, 2015.

[2] E. Scolari, L. Reyes-Chamorro, F. Sossan, and M. Paolone, "A comprehensive assessment of the short-term uncertainty of grid-connected PV systems," *IEEE Transactions on Sustainable Energy*, vol. 9, no. 3, pp. 1458–1467, 2018.

[3] W. Wang, H. Chen, B. Lou, N. Jin, X. Lou, and K. Yan, "Data-driven intelligent maintenance planning of smart meter reparations for large-scale smart electric power grid," in *2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pp. 1929–1935, Guangzhou, China, 2018.

[4] A. García-Olivares, J. Solé, and O. Osychenko, "Transportation in a 100% renewable energy system," *Energy Conversion and Management*, vol. 158, pp. 266–285, 2018.

[5] X. Lü, T. Lu, C. J. Kibert, and M. Viljanen, "Modeling and forecasting energy consumption for heterogeneous buildings using a physical-statistical approach," *Applied Energy*, vol. 144, pp. 261–275, 2015.

[6] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 1–590, 2016.

[7] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 144–153, Dallas, TX, USA, 2019.

[8] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[9] X. Zhou, X. Xu, W. Liang et al., "Intelligent small object detection based on digital twinning for smart manufacturing in industrial CPS," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 2, pp. 1377–1386, 2021.

[10] R. H. Inman, H. T. Pedro, and C. F. Coimbra, "Solar forecasting methods for renewable energy integration," *Progress in Energy and Combustion Science*, vol. 39, no. 6, pp. 535–576, 2013.

[11] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks: a survey towards private and secure applications," 2021, http://arxiv.org/abs/2106.03785.

[12] X. Zhou, Y. Li, and W. Liang, "CNN-RNN based intelligent recommendation for online medical pre-diagnosis support," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 3, pp. 912–921, 2020.

[13] X. Zhou, X. Xu, W. Liang, Z. Zeng, and Z. Yan, "Deep learning enhanced multi-target detection for end-edge-cloud surveillance in smart IoT," *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 12588–12596, 2021.

[14] Y. Cao, X. Zhou, and K. Yan, "Deep learning neural network model for tunnel ground surface settlement prediction based on sensor data," *Mathematical Problems in Engineering*, vol. 2021, 14 pages, 2021.

[15] H. Zhou, Q. Liu, K. Yan, and Y. Du, "Deep learning enhanced solar energy forecasting with AI-driven IoT," *Wireless Communications and Mobile Computing*, vol. 2021, 11 pages, 2021.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] X. Song, J. Huang, and D. Song, "Air quality prediction based on LSTM-Kalman model," in *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, pp. 695–699, Chongqing, China, 2019.

[18] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 841–851, 2019.

[19] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[20] X. Zhou, W. Liang, S. Shimizu, J. Ma, and Q. Jin, "Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5790–5798, 2021.

[21] M. Husein and I. Y. Chung, "Day-ahead solar irradiance forecasting for microgrids using a long short-term memory recurrent neural network: a deep learning approach," *Energies*, vol. 12, no. 10, p. 1856, 2019.

[22] F. A. Gers and J. Schmidhuber, "Recurrent nets that time and count," in *Proceedings of the IEEE-INNS-ENNS international joint conference on neural networks. IJCNN 2000. Neural computing: new challenges and perspectives for the new millennium*, Como, Italy, 2000.

[23] B. Maag, Z. Zhou, and L. Thiele, "A survey on sensor calibration in air pollution monitoring deployments," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4857–4870, 2018.

[24] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The performance of LSTM and BiLSTM in forecasting time series," in *2019 IEEE International Conference on Big Data (Big Data)*, pp. 3285–3292, Los Angeles, CA, USA, 2019.

[25] L. Yufang, C. Mingnuo, and Z. Wanzhong, "Investigating long-term vehicle speed prediction based on BP-LSTM algorithms," *IET Intelligent Transport Systems*, vol. 13, no. 8, pp. 1281–1290, 2019.

[26] Y. Xing and X. Jiaxiang, "Research on photovoltaic power generation prediction of improved LSTM network," *China Test*, vol. 45, no. 11, pp. 14–20, 2019.

[27] H. Zhao, Z. Zhao, H. Wang, and Y. Yue, "Short-term photovoltaic power prediction based on DE-GWO-LSTM," in *2020 IEEE International Conference on Mechatronics and Automation (ICMA)*, pp. 1681–1686, Beijing, China, 2020.

[28] L. Wen, K. Zhou, S. Yang, and X. Lu, "Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting," *Energy*, vol. 171, pp. 1053–1065, 2019.

[29] K. Yan, W. Li, Z. Ji, M. Qi, and Y. Du, "A hybrid lstm neural network for energy consumption forecasting of individual households," *IEEE Access*, vol. 7, no. 1, pp. 157633–157642, 2019.

[30] K. Yan, X. Wang, Y. Du, N. Jin, H. Huang, and H. Zhou, "Multi-step short-term power consumption forecasting with a hybrid deep learning strategy," *Energies*, vol. 11, no. 11, p. 3089, 2018.

[31] H. Zhou, Y. Zhang, L. Yang, Q. Liu, K. Yan, and Y. Du, "Short-term photovoltaic power forecasting based on long short term memory neural network and attention mechanism," *IEEE Access*, vol. 7, pp. 78063–78074, 2019.

[32] H. Zheng, J. Yuan, and L. Chen, "Short-term load forecasting using EMD-LSTM neural networks with a Xgboost algorithm for feature importance evaluation," *Energies*, vol. 10, no. 8, p. 1168, 2017.

[33] X. Wu, J. Li, Y. Jin, and S. Zheng, "Modeling and analysis of tool wear prediction based on SVD and BiLSTM," *The International Journal of Advanced Manufacturing Technology*, vol. 106, no. 9-10, pp. 4391–4399, 2020.

[34] K. Yan, H. Shen, L. Wang, H. Zhou, M. Xu, and Y. Mo, "Short-term solar irradiance forecasting based on a hybrid deep learning methodology," *Information*, vol. 11, no. 1, p. 32, 2020.

[35] X. Zhao, H. Wei, H. Wang, T. Zhu, and K. Zhang, "3D-CNN-based feature extraction of ground-based cloud images for direct normal irradiance prediction," *Solar Energy*, vol. 181, pp. 510–518, 2019.

[36] X. Zhou, X. Yang, J. Ma, and K. I. -K. Wang, "Energy efficient smart routing based on link correlation mining for wireless edge computing in IoT," *IEEE Internet of Things Journal*, 2021.

[37] W. Liang, Y. Hu, X. Zhou, Y. Pan, and K. I. -K. Wang, "Variational few-shot learning for microservice-oriented intrusion detection in distributed industrial IoT," *IEEE Transactions on Industrial Informatics*, p. 1, 2021.

[38] C. Tian, J. Ma, C. Zhang, and P. Zhan, "A deep neural network model for short-term load forecast based on long short-term memory network and convolutional neural network," *Energies*, vol. 11, no. 12, p. 3493, 2018.

[39] H. T. Pedro, D. P. Larson, and C. F. Coimbra, "A comprehensive dataset for the accelerated development and benchmarking of solar forecasting methods," *Journal of Renewable and Sustainable Energy*, vol. 11, no. 3, article 036102, 2019.

[40] I. P. Panapakidis and A. S. Dagoumas, "Day-ahead natural gas demand forecasting based on the combination of wavelet transform and ANFIS/genetic algorithm/neural network model," *Energy*, vol. 118, pp. 231–245, 2017.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8653083, 22 pages
https://doi.org/10.1155/2021/8653083

Research Article

QoE-Oriented Cooperative Broadcast Optimization for Vehicular Video Streaming

Jingyao Liu, Guangsheng Feng, Jiayu Sun, Liying Zheng, and Huiqiang Wang

College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China

Correspondence should be addressed to Guangsheng Feng; fengguangsheng@hrbeu.edu.cn

Received 2 September 2021; Accepted 21 October 2021; Published 23 December 2021

Academic Editor: Yingjie Wang

The popularity of online vehicular video has caused enormous information requests in Internet of vehicles (IoV), which brings huge challenges to cellular networks. To alleviate the pressure of base station (BS), Roadside Units (RSUs) and vehicle peers are introduced to collaboratively provide broadcast services to vehicle requesters where vehicles act as both service providers and service requesters. In this paper, we propose an efficient framework leveraging scalable video coding (SVC) technique to improve quality of experience (QoE) from two perspectives: (1) maximizing the data volume received by all requesters and (2) determining buffer action based on playback fluency and average playback quality. For (1), potential providers cooperate to determine the precached video content and delivery policy with the consideration of vehicular mobility and wireless channel status. If one provider fails, other sources will complement to provide requested content delivery. Therefore, their cooperation can improve the QoE and enhance the service reliability. For (2), according to buffer occupancy status, vehicle requesters manage buffer action whether to buffer new segments or upgrade the enhancement level of unplayed segment. Furthermore, the optimization of the data volume is formulated as an integer nonlinear programming (INLP) problem, which can be converted into some linear integer programming subproblems through McCormick envelope method and Lagrange relaxation. Numerical simulation results show that our algorithm is effective in improving total data throughput and QoE.

1. Introduction

As forecasted by Cisco, mobile video will account for 79% of total mobile data traffic by 2022 [1]. Traditional data transmission mainly relies on cellular data; however, enormous information requests will bring huge challenge to BS suffering from bandwidth competition, transmission interference, and high transmission delay [2, 3]. In this case, the performance of HD video may be deteriorated. Hence, it is inevitable to leverage RSU and vehicles as alternative potential transmission sources to improve user's QoE in vehicular networks. There are some issues faced in the transmission process, such as privacy protection [4–8] and task allocation [9–13]. Vehicle requesters are equipped with multiple network interfaces to support different transmission techniques [14–16], and multisources can compensate each other if one candidate path suffering congestion fails to provide service. Potential providers cooperate to provide desired content, overcoming the limitation of only relying on origin content server. Hence, designing a cooperative transmission mechanism is necessary for improving QoE in vehicular networks [17–20].

It is noteworthy that there are two conditions that must be satisfied to be a potential provider: desired content is exactly cached and the distance is within transmission range. To address this issue, designing caching placement policies under the limitation of storage capacity has been extensively studied [21–26]. However, most previous caching strategies do not take the mobility of vehicles and cooperate broadcast into consideration, which is not suitable to deploy in vehicular ad hoc networks (VANET). In addition, different from the existing work where vehicle client can only receive content from one single source [27–29], we consider a cooperate broadcast mechanism where multisources can transmit different parts of one video segment simultaneously. Multisources cooperating to provide broadcast service can enhance the reliability of transmission, reduce transmission latency, and increase the received data volume.

Considering the mobility characteristic of vehicular network, transmission channel status is variable with the frequent changes of connection with candidate providers. To overcome the disadvantages of bandwidth variation, SVC is an attractive technology with layered feature to support dynamic adaptive resolution [29–35]. According to the current buffer status, vehicle requesters make buffer action to adjust appropriate video quality level. To achieve the objective of maximizing QoE, there are several important parameters which need to be balanced: (1) maintaining adequate segments to avoid playback interruption, (2) achieving better playback smoothness to support high resolution, and (3) avoiding progressive download to reduce bandwidth competition and waste. Most of the previous works do not consider the cache and delivery cooperation among the transmission sources. Hence, the optimization scheme in above works is different from our case where the content caching and content delivery are coupled together. In this paper, we propose a cooperative broadcast mechanism where multisources provide desired content to vehicle requester simultaneously. In addition, vehicle clients can determine buffer actions supporting bit-rate adaption to improve QoE. The contributions of our work can be summarized in the following:

(i) We propose a broadcast mechanism where multisources cooperatively provide service combining content caching and content delivery policy

(ii) We design an optimization model to not only maximize the volume of received data which is formulated as an INLP problem but also improve the QoE of desired content by dynamically determining buffer action

(iii) We apply the McCormick envelope method and Lagrange relaxation to convert the initial INLP problem into several decoupled subproblems which can be solved through a distributed algorithm

(iv) Simulation results show that our cooperative broadcast schemes, distributed algorithm, and buffer action determination algorithm can promote QoE of desired content

The rest of the paper is organized as follows. In Section 2, we provide detailed related works. In Section 3, we introduce the system model and formulate the problem. In Section 4, we propose distributed algorithm and buffer action determination algorithm to solve the integer nonlinear programming problem. Numerical experiments and result analysis are carried out in Section 5. Section 6 concludes this paper.

2. Related Work

We introduce the related work from three research directions. The first one focuses on content cache strategies, and the second one deals with the data delivery to vehicle clients, while the last one works on adaptive quality-level selections for video requesters. However, to the best of our knowledge, few works consider cooperative caching and content delivery to improve the QoE of vehicle users, which may reduce the utilization of cache capacity and limit transmission performance.

2.1. Content Caching. Due to the limited cache capacity, it is important to design appropriate content caching policy to satisfy user's requests. Zhao et al. designed a dynamic probabilistic caching scheme and propose a caching vehicle selection method to reduce the average transmission delay and improve the cache hit ratio [21]. Kumar and Misra proposed an incentive mechanism to encourage users to share the cached content, which can minimize both users' total cost and communication delay [22]. Zhang et al. predicted the requests of vehicle users applying autoregressive neural network and optimize the content cache aiming to minimize energy cost [23]. Li et al. presented a caching optimization model to minimize the average transmit power with the limited cache capacity [24]. Zhu et al. proposed a strategy combining power allocation and layer-wise caching to maximize users' QoE [25]. Hu et al. explored an in-vehicle caching framework to keep the cached data survival in a designated region considering the vehicle mobility [26].

2.2. Content Delivery. Since multiple potential providers are available for transmission, there have been some researchers dealing with content delivery problem. Xu et al. analyzed content mobility and adopted multimedia content delivery approach to achieve high-quality multimedia services [14]. Su et al. developed a pricing model to maximize total utilities by determining content delivery from moving vehicles and RSUs according to the competition and cooperation [16]. Chen et al. proposed to rely on gathered vehicles for task execution, and the task offloading scheme could minimize the task executing time by cooperative computing among vehicles [27]. In [28], a pricing model is performed to motivate vehicle peers to cooperatively provide content delivery aiming to improve both RSU revenue and quality level of vehicle customers. Sun et al. introduced that vehicles and RSUs cooperatively distribute online video to vehicle users, where vehicle providers can be content carriers and relays [29]. Zhu et al. investigated a cooperative content delivery mechanism to motivate parking vehicles and RSUs to contribute cached contents with the auction game [36].

2.3. Adaptive Quality-Level Determination. Considering the fact that the playback buffer states are different for vehicle requesters, users can dynamically select the video quality according to their own situation. There are some works covering this topic. Kumar et al. presented an adaptive QoE management mechanism deciding whether to download a new video segment or smooth the definition level of unplayed video segments [37]. And the proposed strategy can improve delivery satisfactory QoE in highly varying bandwidth state conditions. Xing et al. introduced a best-action search algorithm to improve the quality of services from the perspectives of playback stall, average quality, clarity fluctuation, and service cost [38]. Zhao et al. developed a cross-layer optimization scheme to optimize the QoE of

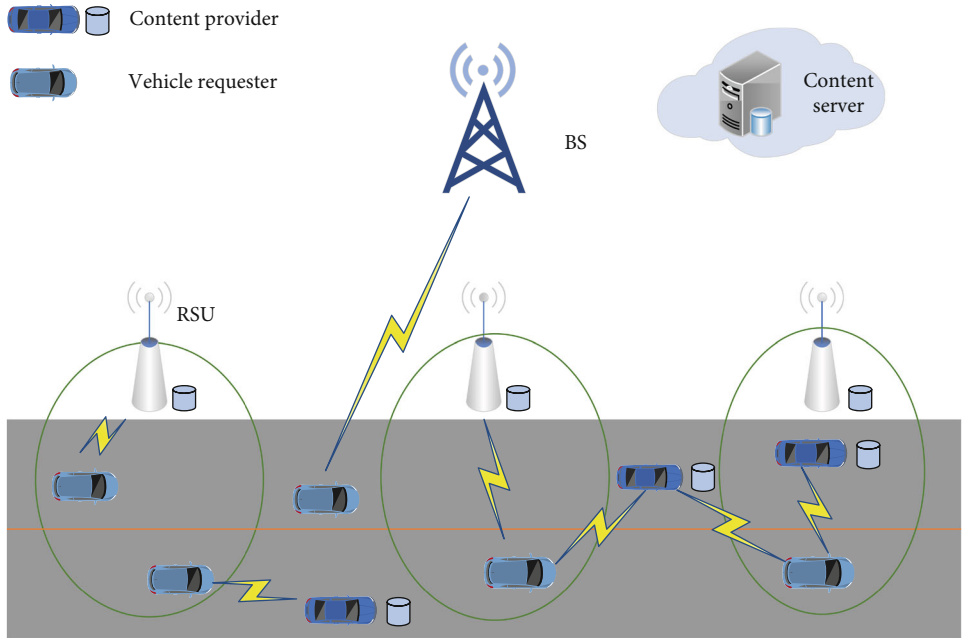


Figure 1: The cooperative broadcast model combining content caching and content delivery.

multiple clients by considering the video rate adaptation and the wireless resource allocation [39]. Cofano et al. designed a hybrid dynamical system to maximize the video bit rate while controlling the quality level switching frequency and minimizing the amount of playback buffer [40]. Bezerra et al. presented a control system estimating the ideal video quality and analyzing buffer state to improve QoE in mobile environment [41].

3. System Model and Problem Formulation

3.1. System Architecture. We consider a scenario combining cellular network and VANET, where vehicle peers and RSUs can be served as transmission sources to provide vehicle requesters with video streaming services. As depicted in Figure 1, vehicles travel on a road where RSUs are deployed and the entire road is covered by BS (cellular network). Vehicle users request online video contents while driving along the road, and surrounding potential providers make decisions including content caching and delivery policy to improve the QoE of the video streaming. Due to the dynamic nature of VANET, we consider discretized time slots, $t \in \{1, 2, \cdots\}$, where content caching and transmission connection can be updated for every slot to cope with the changing position of vehicles. Compared with vehicular position, the content request is almost unchanged, so the content caching is updated less frequently than the transmission connection. Due to the time-varying channel qualities of users, we apply SVC technology to satisfy the various requirements of users for video definition. SVC video consists of a base layer and several enhancement layers, and video definition level increases with the increase of the number of enhancement layers. At the beginning of each slot, vehicle users determine the buffer action according to their current buffer states. Furthermore, the buffer action for each requester is determined depending on the number of video segments in playback buffer and the quality level. The object of buffer action determination is to maximize the quality level while ensuring playback fluency. No matter to continue buffering new segments or to improve quality, it is necessary to transmit as much data as possible.

To maximize the data volume received by all requesters, potential providers need to determine content caching and content delivery policy. RSUs can prefetch video files to provide transmission services for vehicle requesters under their coverage. Besides, the vehicle storage area is divided into two parts: video caching area (providing broadcast service for other vehicle requesters) and the playback buffer area (buffering desired video content to watch online). There are different wireless network interfaces on vehicles to support different transmission techniques. Hence, BS, RSUs, and vehicle peers all can serve as candidate transmission sources to provide broadcast services for target vehicles. Each vehicle requester can communicate with multisources simultaneously, and desired video cached on RSUs and vehicular peers can increase the number of candidate providers. When the number of transmission requests exceeds the service capacity, it will bring enormous transmission pressure on BS. The cooperative transmission of multiple alternative transmission sources can enhance the quality of transmission services. When transmitting through V2V links, multiple vehicle peers can transmit different parts of video segment to the same target vehicle; meanwhile, a vehicle provider can provide content transmission services for multiple vehicle requesters at the same time. However, there are some service/receive upper limitation on content delivery policy. On the one hand, each transmission source can only service a certain number of requesters; on the other hand, one requester can only receive desired content from limited number of vehicle peers. When large amount of vehicle users

request the same video content, the broadcast strategy is more efficient in improving the average QoE of all vehicle clients.

Hence, the following issues need to be resolved: (1) content cache strategy of potential providers, (2) content delivery policy of multisources, and (3) buffer action determination for vehicle requesters.

Let $\mathscr{K} = \{1, 2, \cdots, K\}$ represent the set of RSU, $\mathscr{M} = \{1, 2, \cdots, M\}$ the set of vehicle providers, and $\mathscr{N} = \{1, 2, \cdots, N\}$ the set of vehicle requesters. Define $\mathscr{F} = \{1, 2, \cdots, f, \cdots, F\}$ the whole content library in the content server, in which f is the fth content descending by popularity. Then, the probability of video file f requested by vehicle i is

$$p_i^f = \frac{f^{-r}}{\sum_{n=1}^{F} n^{-r}}. \tag{1}$$

The probability follows the Zipf distribution with the skewness parameter r, in which the larger r indicates the desired contents are more influenced by the concentration of the popularity. When $r = 0$, the distribution of popularity follows the uniform distribution. The symbols mainly used in this paper are summarized in Table 1.

3.2. Content Caching Model. To mitigate the pressure of BS caused by generous transmission requests, RSUs and vehicle peers are served as potential providers, which collaborate to determine what to cache. Let $c_{\text{if}}^{\text{V}}(t) \subseteq \{0, 1\}$ be a binary decision variable which represents whether fth file is cached in vehicle i, $c_{\text{if}}^{\text{V}}(t) = 1$ if vehicle i caches fth file, and $c_{i,f}^{\text{V}}(t) = 0$ otherwise. Let $c_{kf}^{\text{R}}(t) \subseteq \{0, 1\}$ be a binary decision variable which represents whether fth file is cached in RSU k, $c_{kf}^{\text{R}}(t) = 1$ if RSU k caches fth file, and $c_{kf}^{\text{R}}(t) = 0$ otherwise. Due to the limitation of storage capacity, the content cached in RSU and vehicles can only occupy part of the content library. Therefore, cache policy must satisfy the following constraints:

$$\sum_{f=1}^{F} c_{\text{if}}^{\text{V}}(t) \le C_i^{\text{V}}, \quad \forall i \in \mathscr{M}, \tag{2}$$

$$\sum_{f=1}^{F} c_{kf}^{\text{R}}(t) \le C_k^{\text{R}}, \quad \forall k \in \mathscr{K}, \tag{3}$$

where C_i^{V} represents the upper bound of the cache capacity of vehicle i and C_k^{R} represents the upper bound of the cache capacity of RSU k.

3.3. Content Delivery Model. Vehicle requesters can receive different information of one segment from different transmission sources (if a transmission source fails, the others can compensate the fault). In addition, an important condition as a potential provider is that the distance from the target vehicle must be within the transmission range, and the maximal transmission rate between them is a unary function of distance. The position of vehicle nodes can be approximately predicted by existing vehicular mobility models [42]. Multicandidate sources can simultaneously provide content delivery to the target vehicle within their transmission range.

Table 1: Notation table.

Notation	Context
$\mathscr{K}$	Set of RSUs
$\mathscr{M}$	Set of vehicle providers
$\mathscr{N}$	Set of vehicle requesters
p_i^f	Probability of the vehicle i requests the video file f
c_{if}^{V}	If video file f is cached in vehicle i
c_{kf}^{R}	If video file f is cached in RSU k
C_i^{V}	Caching capacity of vehicle i
C_k^{R}	Caching capacity of RSU k
b_{ij}	BLER from provider i to requester j
$v_{ij\,\max}(t)$	Maximal transmission rate under definite distance
$v_{ij}(t)$	Practical transmission rate under definite distance
$v_{0j}(t)$	Practical transmission rate from BS to requester j
$r_{ijf}^{\text{V}}(t)$	If vehicle provider i delivers content f to requester j
$r_{kjf}^{\text{R}}(t)$	If RSU k deliver content f to vehicle requester j
B_i^{V}	Service upper limitation of vehicle provider i
B_k^{R}	Service upper limitation of RSU k
B_{receive}	Receive upper limitation of vehicle requester
Δt	Duration of one-time slot
$\Delta t'$	Duration of one segment
$w_j(t)$	Total data volume received by vehicle requester j
r'	Bit rate of perceived video segment
$r_{\max}$	The highest bit rate of one segment
$S_i(t)$	Initial state of the playback buffer in vehicle i for tth slot
$s_i(t)$	Number of segments in current playback buffer
$a_{i1}(t) \cdots a_{i4}(t)$	Video quality level in four regions, respectively

In addition, the two transmission modes of V2R and V2V are supported by 802.11p using a 6 GHz radio spectrum, where wireless links may suffer from shadowing, interference, and congestion. There is a block error rate (BLER) in the practical transmission rate between nodes. It has been widely acknowledged that the Gilbert channel model is used to describe the packet loss characteristics through wireless links. The wireless channel state is calculated through the Markov chain, where the state is only determined by its immediately previous state. The BLER from provider i to requester j, denoted by b_{ij} [43, 44], is

$$b_{ij} = 1 - \left(1 - \pi_{ij}^{Bad}\right)^M, \tag{4}$$

and we assume that the BLER will not be changed within one time slot. Hence, the practical transmission bit rate between provider i and requester j in each time slot is

$$v_{ij}(t) = \left(1 - b_{ij}\right) * v_{ij\,\max}(t), \tag{5}$$

where $v_{ij\,\max}$ is the maximal transmission rate under the current node transmission distance.

Let $r^{\mathrm{V}}_{ijf}(t), r^{\mathrm{R}}_{kjf}(t) \subseteq \{0, 1\}$ be binary decision variables, which represent whether vehicular source i provides content f to vehicle requester j and whether RSU source k provides content f to vehicle requester j. Limited by the transmission bandwidth, each transmission source can only provide services for a certain amount of vehicle requesters. Meanwhile, the number of vehicle peers providing desired content to one vehicle requester is also limited, which is given by

$$\sum_{j=1}^{N} \max\left(r^{\mathrm{V}}_{ijf}(t), \forall f\right) \le B^{\mathrm{V}}_{i}, \quad \forall i \in \mathscr{M}, \tag{6}$$

$$\sum_{j=1}^{N} \max\left(r^{\mathrm{R}}_{kjf}(t), \forall f\right) \le B^{\mathrm{R}}_{k}, \quad \forall k \in \mathscr{K}, \tag{7}$$

$$\sum_{i=1}^{M} r^{\mathrm{V}}_{ijf}(t) \le B_{\mathrm{receive}}, \quad \forall j \in \mathscr{N}, f \in \mathscr{F}, \tag{8}$$

where B^{V}_{i} and B^{R}_{k} are the upper bound of the number of requesters served by vehicle provider i or RSU k and B_{receive} is the upper bound of the number of vehicle peers providing content to one target vehicle.

It is worth notice that one necessary prerequisite for potential provider vehicle i or RSU k to broadcast content f is

$$c^{\mathrm{V}}_{\mathrm{if}}(t) \ge r^{\mathrm{V}}_{ijf}(t), \quad \forall i \in \mathscr{M}, j \in \mathscr{N}, f \in \mathscr{F}, \tag{9}$$

$$c^{\mathrm{R}}_{kf}(t) \ge r^{\mathrm{R}}_{kjf}(t), \quad \forall k \in \mathscr{K}, j \in \mathscr{N}, f \in \mathscr{F}, \tag{10}$$

which means that content f being cached in candidate i is a necessary condition given that candidate i or k tends to provide transmission service.

Therefore, in each time slot, the data volume received by vehicle requester j can be expressed as

$$w_j(t) = \sum_{f=1}^{F} p^{f}_{j} \left(\sum_{i=1}^{M} c^{\mathrm{V}}_{\mathrm{if}}(t) r^{\mathrm{V}}_{ijf}(t) v^{\mathrm{V}}_{ij}(t) \Delta t + \sum_{k=1}^{K} c^{\mathrm{R}}_{kf}(t) r^{\mathrm{R}}_{kjf}(t) v^{\mathrm{R}}_{kj}(t) \Delta t + \left(1 - g\left(\sum_{i=1}^{M} r^{\mathrm{V}}_{ijf}(t) + \sum_{k=1}^{K} r^{\mathrm{R}}_{kjf}(t)\right)\right) v_{0j}(t) \Delta t \right). \tag{11}$$

In equation (11), $g(x)$ is the function of x and equals to 1 if $x > 0$ or 0 if $x = 0$, and $v_{0j}(t)$ is the transmission rate of BS. The previous two polynomials represent the amount of data transmitted by neighboring vehicles and RSU, respectively. And the last polynomial means the data volume transmitted by BS if there is no other candidate provider.

3.4. QoE Calculation Model. In order to alleviate the pressure of the BS, vehicle sources and RSU sources are preferred to broadcast. If there are no well-conditioned channels from RSUs and vehicle peers to requesters, then BS steps in as candidate source to provide service. Considering the high dynamics characteristic of the IoV (Internet of vehicles) channel, the flexibility of SVC technology can be utilized to solve different resolution requirements of vehicle requesters. There is a significant advantage for SVC in that it allows quality upgradation of the video segments (unwatched in current time slot) in buffer blocks. SVC technology is adaptive and does not require decoding and reencoding of the signal. It supports buffering more enhancement layers to upgrade the quality of video on basis of the original buffered video layers.

For each video segment, the video bit rate of different quality level is denoted by $R = \{r_1, r_2, \cdots, r_l\}$ (l is the best possible enhancement level). We apply the mean opinion score (MOS) to measure the perceived video quality. Based on literature [45], the formula evaluating average QoE can be expressed as

$$\mathrm{QoE} = 4 * \exp\left(-c_1 \left(\frac{r'}{r_{\max}}\right)^{-c_2} + c_1\right) + 1, \tag{12}$$

where r' represents the bit rate of perceived video segment and $r_{\max} = r_l$ is the highest bit rate of one segment.

We mainly improve the QoE of video desired by vehicle requesters from the following parameters: (1) maintaining sufficient number of video segments in the playback buffer in order to reduce playback stall time, (2) maximizing the bit rates of video segments in the playback buffer to support better resolution, and (3) avoiding progressive download which will result in the waste of transmission resources and affect other vehicle requesters.

3.5. Buffer Action Model. The status of vehicular playback buffer is shown in Figure 2. Each video is divided into several segments with the same duration $\Delta t'$, and the number of segments played in each time slot is fixed.

The upper limit of the entire playback buffer length is L, which is divided into 4 regions. The video segments buffered in region $[0, S_1]$ are prepared for playing in the current time slot, the video segments buffered in region $[S_1, S_2]$ are reserved for the next time slot, and regions $[S_2, S_3]$ and $[S_3, S_4]$ are provided for the future time slots. The number of segments buffered in the region $[0, S_1]$, $[S_1, S_2]$, $[S_2, S_3]$, and $[S_3, S_4]$ is the same, whose duration is the length of video playback in each time slot. Let $S_i(t)$ denote the initial state of the playback buffer in vehicle i for tth time slot, and client i dynamically makes a decision whether to continue downloading a new segment or to enhance the quality level of the unplayed video segment according to its buffer state. The buffer status $S_i(t)$ contains two important factors: the number of video segments in playback buffer and the quality

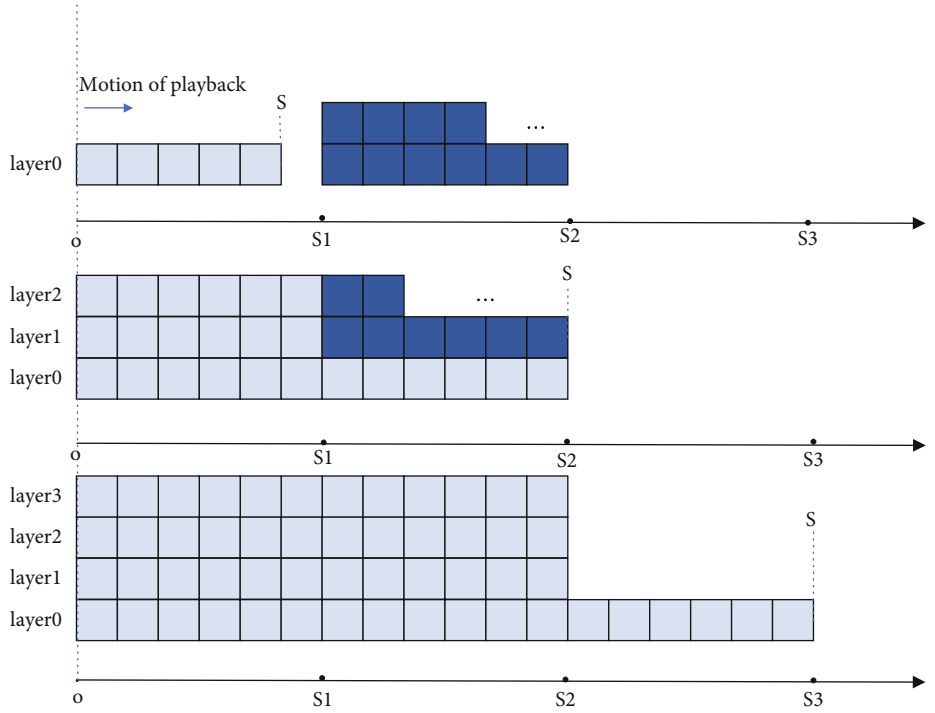


Figure 2: Buffer action for vehicle requesters under a different status.

level of the video segments. s_i represents the number of segments in current playback buffer, and a_{i1}, a_{i2}, a_{i3}, and a_{i4} are the video quality level in the aforementioned four regions, respectively. It is noteworthy that the segments buffered in region $[0, S_1]$ are played in the current time slot, so in each time slot, the incremental data cannot change the state of video segments in this region.

Since the video segments buffered in region $[0, S_1]$ are played in the current time slot, when $s_i(t) < S_1$, the current playback will be interrupted and the user's perception will be seriously affected. Hence, buffer action a is taken: continuing to download new segments to prevent an interruption occurring in next time slots. When $S_1 \le s_i(t) < S_2$, the current time slot can playback smoothly, but the next time slot video playback is at risk of interruption, so it is necessary to continue downloading new segments. When $S_2 \le s_i(t) < S_3$, there is no possibility of interruption in current and next slots, so the buffer action determination is b: upgrading the quality level of video segments reserved for subsequent time slots and then downloading new segments. When $s_i(t) \ge S_3$, it means that there will be no playback interruption in a short time and the resolution quality is high. So buffer action decision is c: stopping the transmission service to vehicle requester i to prevent competition with other users for resource, which can be represented by the following constraint:

$$\sum_{j=1}^{M} r_{jif}^{\mathrm{V}}(t) + \sum_{k=1}^{K} r_{kif}^{\mathrm{R}}(t) = 0, \text{ if } s_i(t) \ge S_3. \tag{13}$$

To maximize the average QoE of desired video required by all vehicle clients, we propose to maximize the received data volume first.

$$\begin{aligned} \mathrm{P1}: \text{Maxmize} \sum_{j=1}^{N} w_j'(t) = \sum_{j=1}^{N}\sum_{f=1}^{F} p_j^f \\ \cdot \left(\sum_{i=1}^{M} c_{\mathrm{if}}^{\mathrm{V}}(t) r_{ijf}^{\mathrm{V}}(t) v_{ij}^{\mathrm{V}}(t) \Delta t + \sum_{k=1}^{K} c_{k,f}^{\mathrm{R}}(t) r_{kjf}^{\mathrm{R}}(t) v_{kj}^{\mathrm{R}}(t) \Delta t \right). \end{aligned} \tag{14}$$

As mentioned above, if neither RSU nor vehicle peers can provide transmission for the target vehicle, then the service source will be transferred to the BS. Therefore, after the optimization operation is performed, according to the result, if $\sum_{i=1}^{M} r_{ijf}^{\mathrm{V}}(t) + \sum_{k=1}^{K} r_{kjf}^{\mathrm{R}}(t) = 0$, $s_i(t) < S_3$, $w_j^f(t)$ will be updated:

$$w_j^f(t) = p_j^f v_{0j}(t) \Delta t. \tag{15}$$

And we can obtain the final optimal data throughput w_j. According to the initial status of playback buffer and the optimization result w_j, the new buffer status after the execution of above buffer action is calculated:

For buffer action a, the number of incremental video segments and the corresponding quality level can be represented by the following formulation:

$$\begin{aligned} s_i(t+1) = S_1 f\left(\frac{w_i(t)}{r_1 \Delta t' (S_2 - S_1)} \right) \\ + (S_2 - S_1) f\left(\frac{w_i(t)}{r_l \Delta t' (S_2 - S_1) + r_1 \Delta t' (S_3 - S_2)} \right) \\ + (S_3 - S_2) f\left(\frac{w_i(t)}{r_l \Delta t' (S_3 - S_1) + r_1 \Delta t' (S_4 - S_3)} \right). \end{aligned} \tag{16}$$

In equation (16), $f(x)$ is a function of x, where $f(x)$ equals to 1 if $x \geq 1$ or 0 if $x < 1$.

$$a_{i2}(t) = \begin{cases} 1, r_1 \Delta t'(S_2 - S_1) \leq w_i(t) < r_2 \Delta t'(S_2 - S_1), \\ 2, r_2 \Delta t'(S_2 - S_1) \leq w_i(t) < r_3 \Delta t'(S_2 - S_1), \\ \cdots \\ l-1, r_{l-1} \Delta t'(S_2 - S_1) \leq w_i(t) < r_l \Delta t'(S_2 - S_1), \\ l, \text{otherwise}. \end{cases} \tag{17}$$

The calculation of quality level a_{i3}, a_{i4} of video segments in region $[S_2, S_3], [S_3, S_4]$ is similar to a_{i2}.

For buffer action b, according to the video quality level of the current buffer region $[S_1, S_2]$, we need to compare the relationship between the received data w_j and the bit rate required for additional enhancement. Users will buffer video data of the next enhancement layer in sequence until the highest definition is reached. After meeting the conditions of the highest quality layer, users continue to buffer new video segments in the next region. The specific calculation formula is similar to formula (17).

For buffer action c, it means that the number of video segments in the playback buffer is sufficient and the quality level of video is high. In order to prevent resource waste caused by excessive buffering, it is necessary to stop providing transmission services to requesters. To decouple the two variables in the constraint (13), the formula can be equivalently transformed into the following constraint:

$$\sum_{j=1}^{M} r_{jif}^{\mathrm{V}}(t) = 0, \text{ if } s_i(t) \geq S_3, \tag{18}$$

$$\sum_{k=1}^{K} r_{kif}^{\mathrm{R}}(t) = 0, \text{ if } s_i(t) \geq S_3. \tag{19}$$

4. Algorithm Design and Algorithm Analysis

In this section, we introduce the proposed joint optimization algorithm and then analyze time complexity and convergence of the algorithm.

4.1. Distributed Algorithm Design. To simplify our algorithm design, we introduce a set $\mathscr{I} = \mathscr{K} \cup \mathscr{M} = \{1, \cdots, K, K+1, \cdots, K+M\}$, where the 1st to Kth elements represent RSUs, and the $(K+1)$-th to $(K+M)$-th elements represent vehicle providers. We denote variable $c_{ij}^{f}(t)$ to replace $c_{\mathrm{if}}^{\mathrm{V}}(t), c_{kf}^{\mathrm{R}}(t)$, and variable $r_{ij}^{f}(t)$ to replace $r_{ijf}^{\mathrm{V}}(t), r_{kjf}^{\mathrm{R}}(t)$. In the preceding section, since there are two variables in formula (P1) that are tightly coupled, the aforementioned optimization of received data volume is an integer nonlinear programming (INLP) problem which is well known NP-hard. To solve this problem, firstly, we employ the McCormick envelope method to relax problem (P1) by introducing a set of additional variables $z_{ij}^{f}(t)\{i \in \mathscr{I}, j \in \mathscr{N}, f \in \mathscr{F}\}$ to replace $c_{\mathrm{if}}^{\mathrm{V}}(t) r_{ijf}^{\mathrm{V}}(t)$ and $c_{kf}^{\mathrm{R}}(t) r_{kjf}^{\mathrm{R}}(t)$. Secondly, we apply Lagrange relaxation to decompose the optimization problem into several subproblems.

To convert the original problem (P1) into several integer linear problems, several constraints will be added with the introduced variables z_{ij}^{f} [46, 47], which are given by

$$z_{ij}^{f}(t) \geq r_{ij}^{f}(t) + c_{i}^{f}(t) - 1, \quad \forall i \in \mathscr{I}, j \in \mathscr{N}, f \in \mathscr{F}, \tag{20}$$

$$z_{ij}^{f}(t) \leq r_{ij}^{f}(t), \quad \forall i \in \mathscr{I}, j \in \mathscr{N}, f \in \mathscr{F}, \tag{21}$$

$$z_{ij}^{f}(t) \leq c_{i}^{f}(t), \quad \forall i \in \mathscr{I}, j \in \mathscr{N}, f \in \mathscr{F}, \tag{22}$$

$$z_{ij}^{f}(t) \geq 0, \quad \forall i \in \mathscr{I}, j \in \mathscr{N}, f \in \mathscr{F}. \tag{23}$$

Therefore, P1 can be converted to the following formula:

$$\begin{aligned} \mathrm{P2}: \text{Maxmize} \sum_{j=1}^{N} w_j'(t) = \sum_{j=1}^{N} \sum_{f=1}^{F} p_j^f \sum_{i=1}^{I} z_{ij}^{f}(t) \\ \cdot v_{ij}(t) \Delta t \text{s.t.} (2)(3), (6) \sim (9), (18)(19), (20) \sim (23). \end{aligned} \tag{24}$$

Hence, the Lagrangian function decoupling the associated variables of P2 is expressed as

$$\begin{aligned} L(\lambda, \beta, \gamma, \mu, z, r, c) = \sum_{j=1}^{N} \sum_{f=1}^{F} p_j^f \Bigg(\sum_{i=1}^{I} - z_{ij}^{f}(t) v_{ij}(t) \Delta t + \lambda_{ij}^{f} \Big(r_{ij}^{f}(t) \\ + c_{i}^{f}(t) - 1 - z_{ij}^{f}(t) \Big) + \beta_{ij}^{f} \Big(r_{ij}^{f}(t) - c_{i}^{f}(t) \Big) \\ + \gamma_{ij}^{f} \Big(z_{ij}^{f}(t) - r_{ij}^{f}(t) \Big) + \mu_{ij}^{f} \Big(z_{ij}^{f}(t) - c_{i}^{f}(t) \Big) \Bigg), \end{aligned} \tag{25}$$

where λ, β, γ, and μ are the introduced Lagrange multipliers:

$$\lambda_{ij}^{f} \geq 0, \beta_{ij}^{f} \geq 0, \gamma_{ij}^{f} \geq 0, \mu_{ij}^{f} \geq 0, \quad \forall i \in \mathscr{I}, \forall j \in \mathscr{N}, \forall f \in \mathscr{F}. \tag{26}$$

Correspondingly, the dual of P2 is

$$\begin{aligned} \mathrm{P3}: \max_{\lambda, \beta, \gamma, \mu} \min_{z, r, c} L(\lambda, \beta, \gamma, \mu, z, r, c) \\ \text{s.t.} (2)(3), (6) \sim (10), (18)(19), (20) \sim (23), (26). \end{aligned} \tag{27}$$

Then, P3 can be decomposed into three independent subproblems, which is given as follows:

$$L(\lambda, \beta, \gamma, \mu, z, r, c) = f_1(z) + f_2(r) + f_3(c), \tag{28}$$

where $f_1(z), f_2(r)$, and $f_3(c)$ are the objective functions of

1: **Input:** $p_j^f, b_{ij} loca[m]$, $\varepsilon = 0.01$, $d_{\max} = 1000$
2: **Output:** $opt(c_i^f(t), r_{ij}^f(t), z_{ij}^f(t), w_j(t))$
3: Initialize $\lambda, \beta, \gamma, \mu, Z_{UP}(t)$
4: **while** $\|S^d\| \geq \varepsilon$ and $d \leq d_{\max}$ **do**
5: Solve $P3.1$ and get the values of $z_{ij}^f(t)$;
6: Solve $P3.2$ and get the values of $r_{ij}^f(t)$;
7: Solve $P3.3$ and get the values of $c_i^f(t)$;
8: Vehicle requester j calculates its received data w_j' using (14);
9: Update step size $\sigma(d)$;
10: Update $\|S^d\|$;
11: Update Lagrange Multipliers $\lambda, \beta, \gamma, \mu$ using (32)–(35);
12: $d = d + 1$;
13: **end while**
14: if $\sum_{i=1}^{I} r_{ij}^f(t) = 0, s_i(t) < S_3$ **then**
15: Vehicle requester j turns to be broadcasted by BS and updates received data $w_j(t)$ using (15);
16: **end if**
17: return the data volume $w_j(t)$ for vehicle requester;

Algorithm 1: Distributed algorithm for primal optimization problem.

subproblems P3.1, P3.2, and P3.3, respectively. The subproblems are given as follows:

$$\begin{aligned} \text{P3.1}: \min_{z} & \sum_{j=1}^{N}\sum_{f=1}^{F}\sum_{i=1}^{I} -p_j^f z_{ij}^f(t) v_{ij}(t)\Delta t - \lambda_{ij}^f z_{ij}^f(t) \\ & + \gamma_{ij}^f z_{ij}^f(t) + \mu_{ij}^f z_{ij}^f(t) \text{s.t.}(23), \end{aligned} \tag{29}$$

$$\begin{aligned} \text{P3.2}: \min_{r} & \sum_{j=1}^{N}\sum_{f=1}^{F}\sum_{i=1}^{I} \lambda_{ij}^f r_{ij}^f(t) + \beta_{ij}^f r_{ij}^f(t) - \gamma_{ij}^f r_{ij}^f(t) \\ & \text{s.t.}(6) \sim (8), (18)(19), \end{aligned} \tag{30}$$

$$\begin{aligned} \text{P3.3}: \min_{c} & \sum_{j=1}^{N}\sum_{f=1}^{F}\sum_{i=1}^{I} \lambda_{ij}^f c_i^f(t) - \beta_{ij}^f c_i^f(t) - \mu_{ij}^f c_i^f(t) \\ & \text{s.t.}(2)(3). \end{aligned} \tag{31}$$

All of these subproblems are integer linear problems, which can be solved by the generic linear integer programming method [48]. We employ the subgradient method to update the Lagrangian multipliers λ, β, γ, and μ iteratively:

$$\lambda_{ij}^f(d+1) = \left[\lambda_{ij}^f(d) + \sigma(d)\left(r_{ij}^f(t) + c_i^f(t) - 1 - z_{ij}^f(t)\right)\right]^+, \tag{32}$$

$$\beta_{ij}^f(d+1) = \left[\beta_{ij}^f(d) + \sigma(d)\left(r_{ij}^f(t) - c_i^f(t)\right)\right]^+, \tag{33}$$

$$\gamma_{ij}^f(d+1) = \left[\gamma_{ij}^f(d) + \sigma(d)\left(z_{ij}^f(t) - r_{ij}^f(t)\right)\right]^+, \tag{34}$$

$$\mu_{ij}^f(d+1) = \left[\mu_{ij}^f(d) + \sigma(d)\left(z_{ij}^f(t) - c_i^f(t)\right)\right]^+, \tag{35}$$

where d is the iteration number, $\sigma(d)$ is the step size, and $[.]^+ = \max\{0, .\}$. The formulation of step size $\sigma(d)$ is

$$\sigma(d) = \frac{Z_{\text{UP}}(d) - Z_{LB}(d)}{\left\|S^d\right\|^2}\varphi, \tag{36}$$

where $S^d = [d(\lambda(d)), d(\beta(d)), d(\gamma(d)), d(\mu(d))]^T$, $Z_{\text{UP}}(d)$ is the upper bound which is a feasible solution of the primary problem, $Z_{LB}(d)$ is the received data volume in the dth iteration, and φ is a positive constant. Notice that the algorithm will converge when $\sigma(d)$ has no significant change anymore. The detailed method is summarized in Algorithm 1.

The procedure of choosing suitable buffer action and calculating QoE of played video segments is illustrated in Algorithm 2. According to the current buffer state, each time slot executes the following loop. It is easy to note that the initial state of each time slot is determined by the received data throughput and buffer action of the previous time slot. First, for each slot, the algorithm attains the initial buffer states of each requesters, and then users determine buffer action. There is a key part in line 4, if user i has already buffered relatively sufficient segments, we need to stop its transmission service. So it is necessary to add constraint (18) and (19) to Algorithm 1. Next, the actual received data volume of each vehicle requesters can be calculated through Algorithm 1. After that, from steps 10 to 20, each requester calculates the new buffered segments and updates the quality level (a_{i2}, a_{i3}, a_{i4}) in corresponding regions. Since each time slot can only play video segments in region $[0, S_1]$, it results in that QoE is determined by the quality level a_{i1}. Finally, the QoE of video segments in region $[0, S_1]$ is calculated and buffer state is updated in each for loop.

1: Input: current buffer state $S_i(t)$, received data throughput w_i
2: Output: buffer action, updated buffer state $S_i(t+1)$,
QoE of video segment in region $[0, S_1]$
3: **for** each time slot tdo**do**
4: **if**$s_i(t) \geq S_3$**then**
5: choose buffer action c, no buffer for vehicle requester i;
6: Adding constraints $\sum_{i=1}^{M} r_{ij}^f(t) = 0$, $\sum_{k=1}^{K} r_{kj}^f(t) = 0$ to algorithm 1;
7: $w_i(t) = 0$;
8: **end if**
9: obtain the data throughput w_i through algorithm 1;
10: **if**$s_i(t) < S_2$**then**
11: choose buffer action a and calculate the new buffered segments and quality level;
12: update $s_i(t)$ and video quality level $a_{i2}(t)$, $a_{i3}(t)$, $a_{i4}(t)$ in next intervals respectively;
13: **end if**
14: **if**$S_2 \leq s_i(t) < S_3$ then
15: choose buffer action b and upgrade the quality level in region $[S_1, S_2]$ then buffer new segments;
16: update $s_i(t)$ and video quality level $a_{i2}(t)$, $a_{i3}(t)$, $a_{i4}(t)$ in next intervals respectively;
17: **end if**
18: **if**$s_i(t) \geq S_3$**then**
19: no buffer state changed;
20: **end if**
21: calculate the QoE of video segment in region $[0, S_1]$;
22: update buffer states for next time slot $a_{i2}(t) = a_{i1}(t+1)$, $a_{i3}(t) = a_{i2}(t+1)$, $a_{i4}(t) = a_{i3}(t+1)$;
23: **end for**
24: return buffer action and QoE for each time slot;

Algorithm 2: QoE-oriented buffer action determination mechanism.

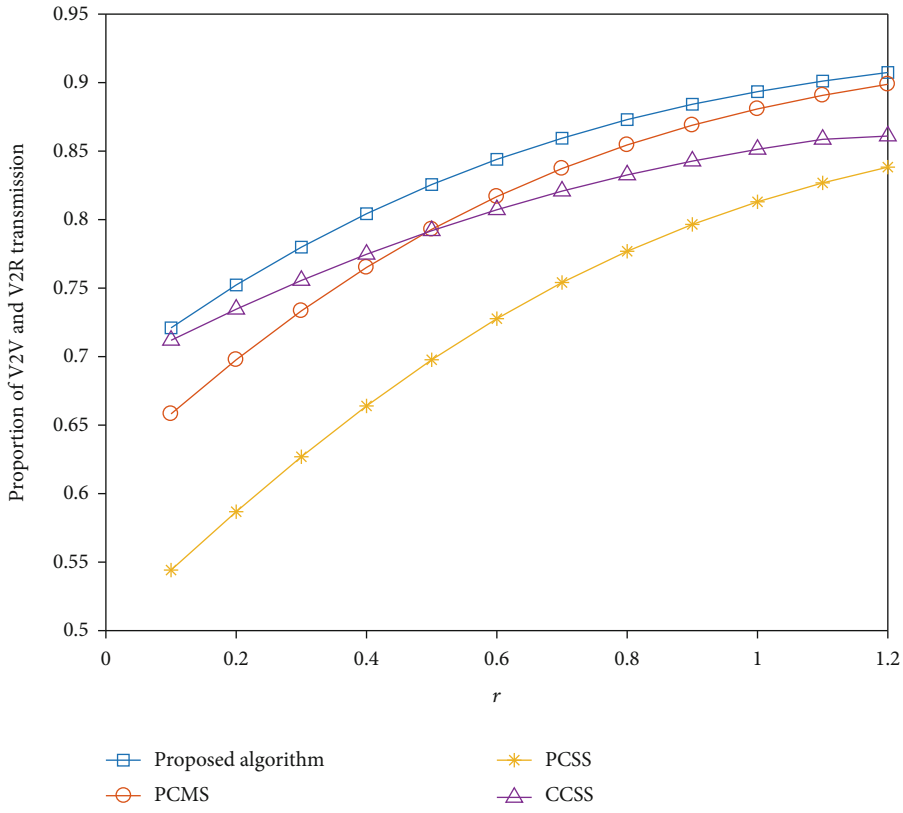
Table 2: Simulation parameters.

Parameter	Value
Number of vehicle requesters	10
Number of vehicle providers	3
Number of video content in library	50
Layers of video segment	4
Number of segments for each time slot	6
Duration of one time slot Δt	6 s
Duration of one video segment $\Delta t'$	1 s
Popularity parameter r	0.6
Upper cache capacity for RSU	10 video contents
Upper cache capacity for vehicle	5 video contents
Max transmission rate for B2R	2 Mbps
Max transmission rate for R2V	4 Mbps
Max transmission rate for V2V	6 Mbps
Average bit-rate for different layers w_0, w_1, w_2, w_3	2, 3, 4, 7 Mbps
c_1, c_2	0.16, 0.66

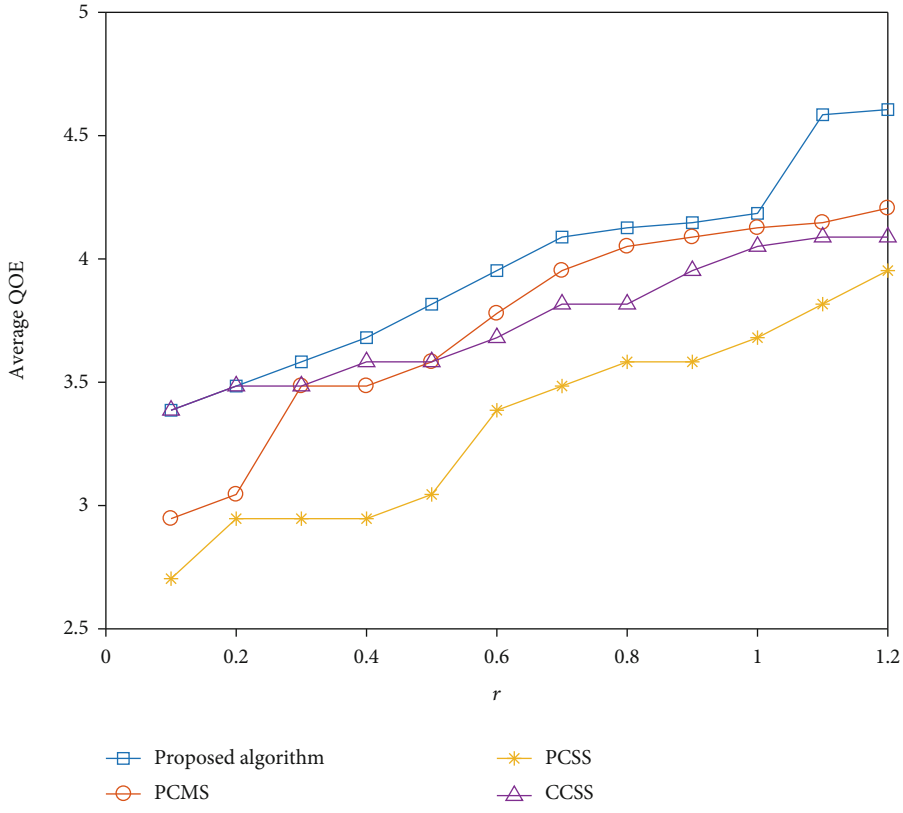
4.2. Algorithm Analysis. In this part, we analyze the convergence and computational complexity of the algorithm.

4.2.1. Convergence. Based on the proof in [49, 50], the proposed algorithm can gradually improve the optimization result and converge after a limited number of iterations. Furthermore, different feasible solutions $Z_{\mathrm{UP}}(d)$ affect the convergence speed but do not change the convergence result. The closer the feasible solution $Z_{\mathrm{UP}}(d)$ is to the optimal solution, the faster the convergence speed of the algorithm.

4.2.2. Computational Complexity. The proposed algorithm executes $O(1/\varepsilon^2)$ iterations to satisfy the accuracy ε ($\varepsilon > 0$ and is a small parameter) of the subgradient method [51]. Besides, the computational complexity of each subproblem solved by the generic linear integer programming method

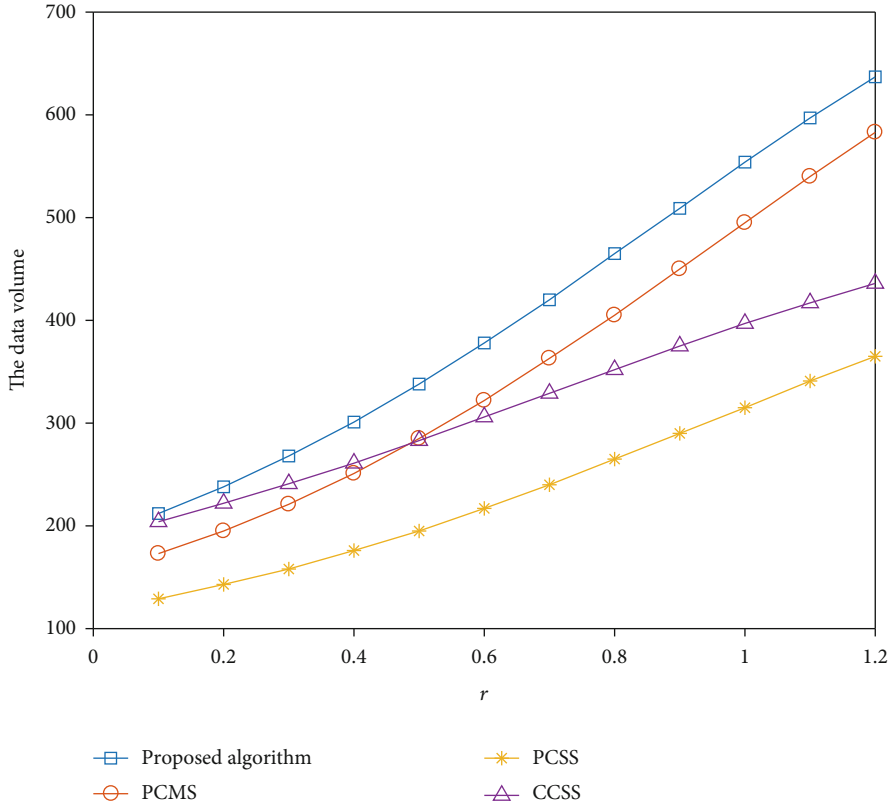


(a) Proportion of V2V and V2R transmission

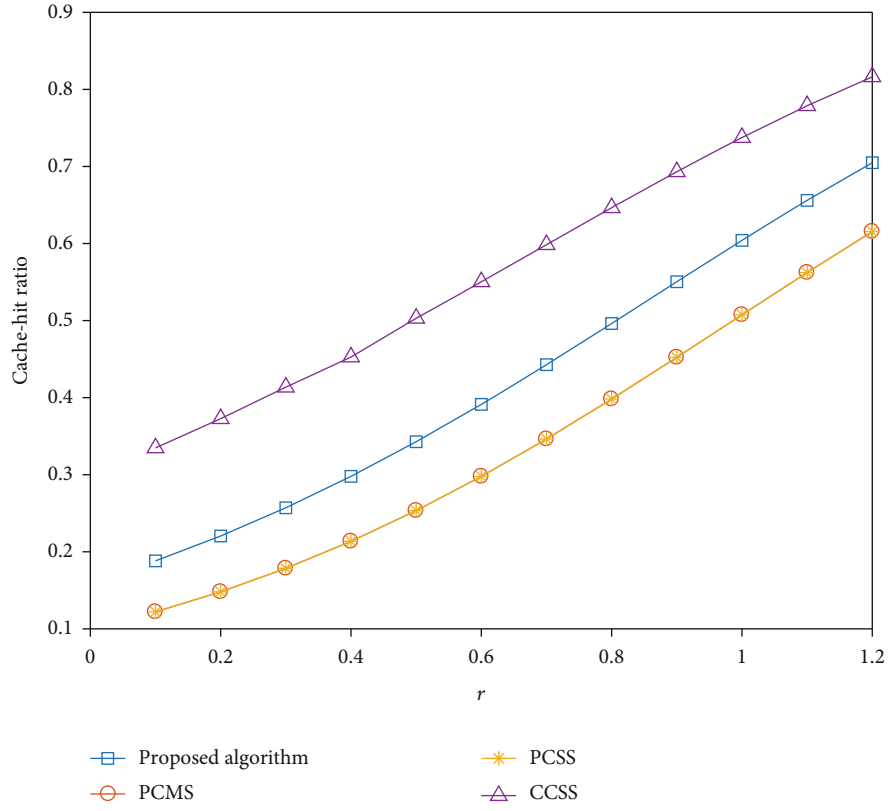


(b) Average QoE

Figure 3: Continued.

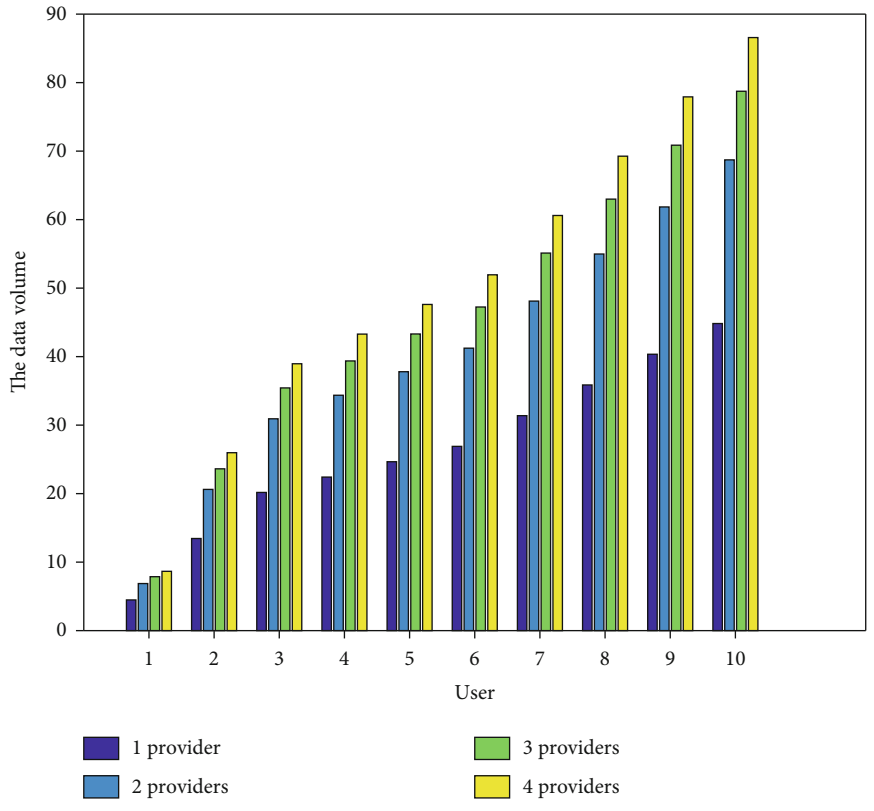


(c) The data volume

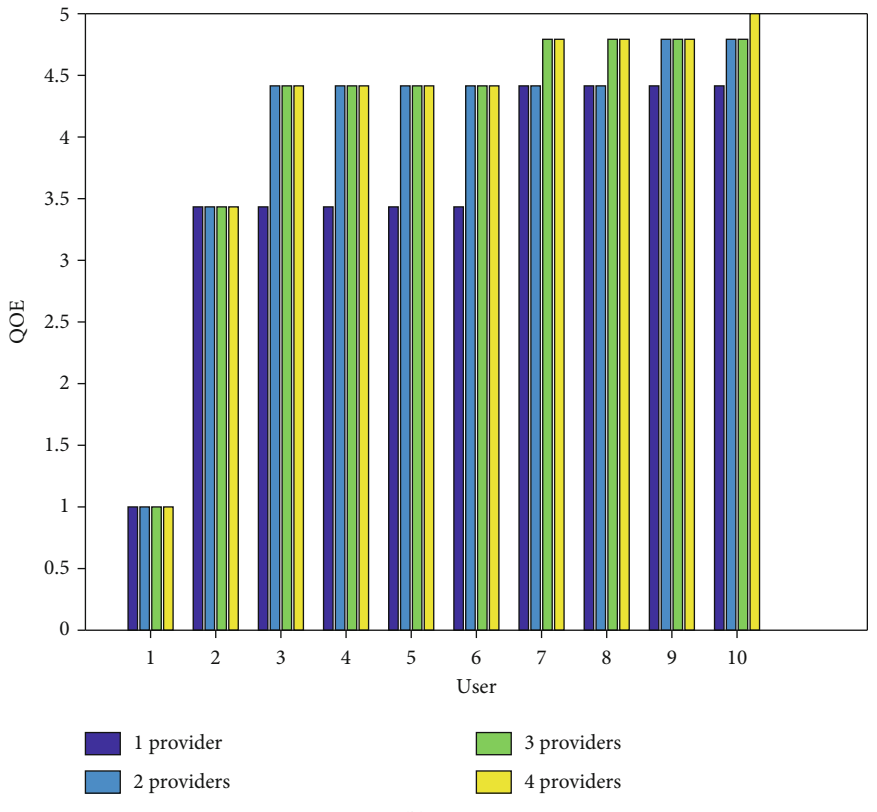


(d) Cache hit ratio

Figure 3: Impact of popularity parameter r.



(a) The data volume



(b) QoE

Figure 4: Impact of the number of vehicle providers.

Table 3: The BLER of users.

(a)

$User_1$	$User_2$	$User_3$	$User_4$	$User_5$
[0.9,1]	[0.8,0.9]	[0.7,0.8]	[0.6,0.7]	[0.55,0.6]

(b)

$User_6$	$User_7$	$User_8$	$User_9$	$User_{10}$
[0.5,0.55]	[0.45,0.5]	[0.4,0.45]	[0.2,0.3]	[0,0.1]

is $O(N^{3.5}L^2)$, where N is the dimension of the variable, and L is a positive constant [48]. Therefore, the computational complexity of the proposed algorithm is $O(N^{3.5}L^2/\varepsilon^2)$.

5. Numerical Experiments and Results

In this section, we conduct comprehensive simulations to evaluate the effectiveness of our proposed cooperative broadcast optimization algorithm.

5.1. Simulation Scenarios. Similar to [29], we perform the simulation in a scenario with multilane road and RSUs whose coverage is 400 m located on road at equal distance. We assume the video library contains 50 video files, and each video is divided into fixed duration segments. Besides, each video segment is encoded into 4 layers, and the bit rate of each enhancement layer is set as w_0, w_1, w_2, and w_3 separately [37]. All the significant simulation parameters are collected in Table 2.

We use several quantitative evaluations to measure the optimization result compared with other schemes including popularity-based caching and multiple transmission sources (PCMS), popularity-based caching and single transmission sources (PCSS), and cooperative caching and single transmission source (CCSS).

(1) Data volume received: according to Algorithm 1, calculating the total amount of data successfully received by all requesters through joint optimization of caching and transmission association, users can decode corresponding resolution level video after receiving sufficient data

(2) Average QoE of all vehicle requesters: according to the total data volume received and Algorithm 2, calculating QoE of video segments in buffer region [0, $S1$] which directly reflects the video definition level

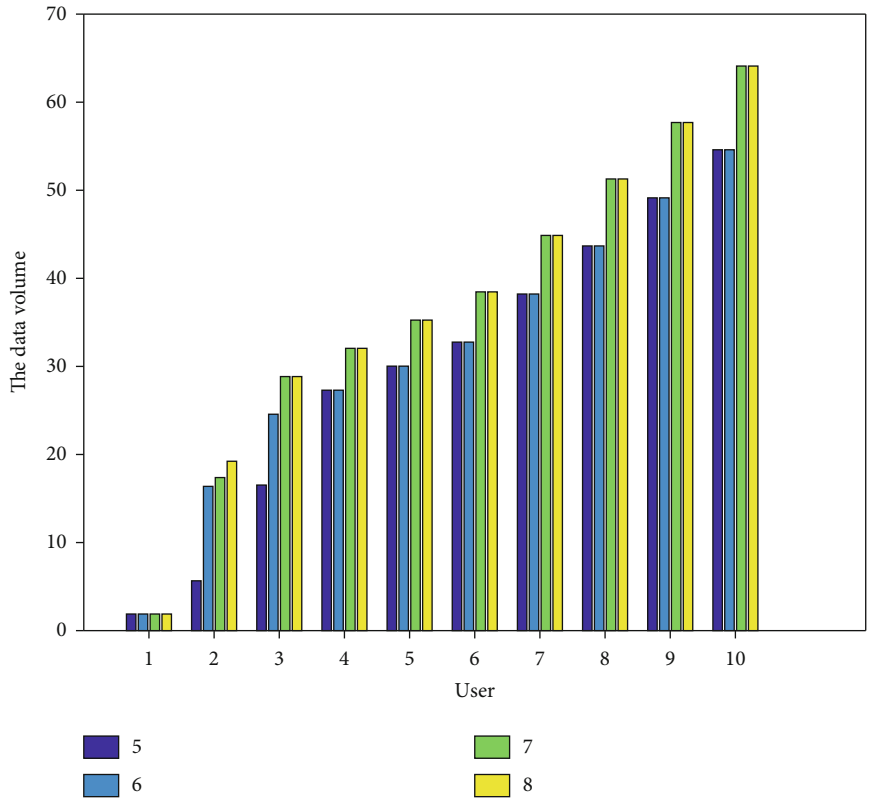
(3) Proportion of V2V and V2R transmission: total volume of data received by vehicle requesters come from three kinds of sources, where a larger proportion of V2V and V2R transmission represents better to alleviate the pressure of BS

(4) Cache hit ratio: desired video content can be satisfied by vehicle and RSU cache, so it is not necessary to attain transmission service from a remote server

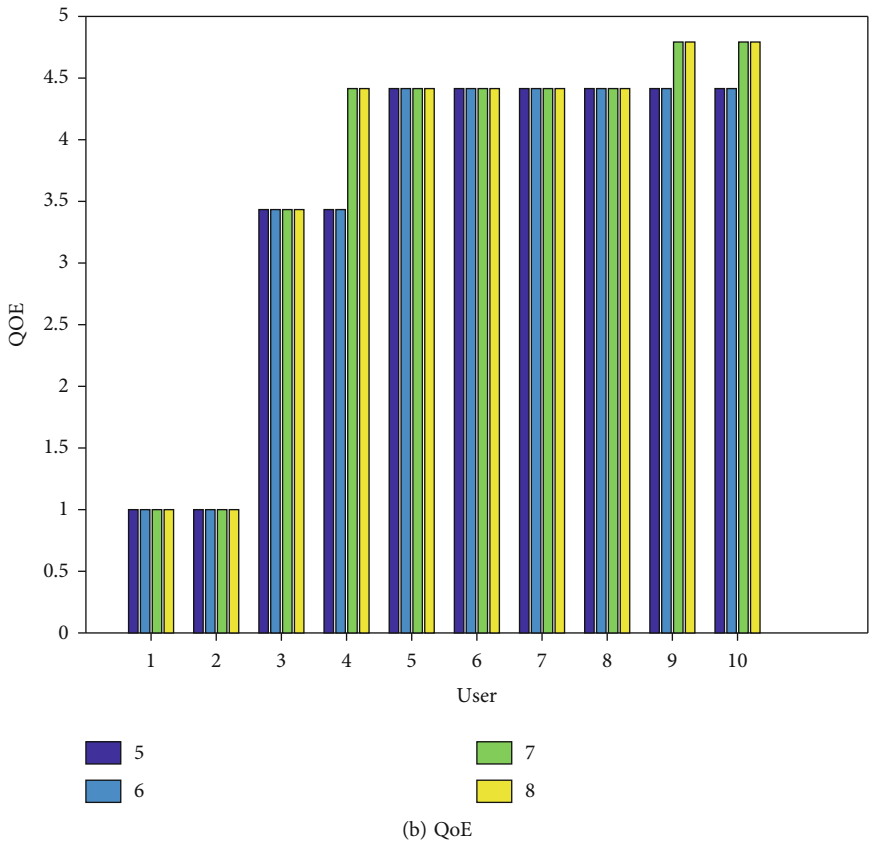
5.2. Impact of Popularity Parameter r. The tendency of above quantitative measurements under different popularity parameter r is shown in Figure 3. It can be found that the proposed algorithm outperforms the others. Furthermore, with the increase of popularity parameter r, all of the measurements, including the total data volume, average QoE of all vehicle requesters, and proportion of V2V and V2R transmission and cache hit ratio, will increase. The reason can be summarized as follows: as mentioned earlier, a larger r means requesters concentrate more on the most popular video contents. Therefore, the content cached in providers can hit more desired files and RSU and vehicle peers can broadcast more data. Furthermore, no coordination in cache scheme based on popularity causes it to perform worse. Every sources cache the most popular video files, so identical content in the cache cannot complement each other and content provided by vehicles is very limited. It can be found in Figure 3(b); the growth trend of QoE is not obvious as other measurements; it is because in order to avoid continuous fluctuations in definition per time slot, the incremental amount must reach a certain level; then, the value of QoE will increase with the increase of the enhancement layer.

5.3. Impact of the Number of Vehicle Providers. The impact of the number of vehicle providers for various requesters over different BLERs is shown in Figure 4. The BLERs of users are listed in Table 3, and users are sorted according to channel status. In order to analyze the extreme channel scenario, we set BLERs for different users with a large gap which is usually quite small in practice. It can be found that with the increase of vehicle providers, the data volume received and average QoE will increase. Furthermore, the worse the channel quality, the smaller the increasing degree. The reason is that our target is to maximize the data volume received, but the reward of providing transmission for users with poor channel quality is small under the condition of limited number of transmissions. Therefore, it will provide priority to users with good channel quality. And when the number of vehicle providers reaches 4, for users with good channel conditions, the video quality level can be the highest, and it does not make much sense to continue to increase the number of transmission sources.

5.4. Impact of the Service Upper Limitation. To illustrate the impact of the service upper bound, we adjust the number of vehicle requesters that a vehicle provider can serve simultaneously from 5 to 8. Figure 5(a) shows that when the upper bound increases from 6 to 7, the data volume received for most users increases obviously. Besides the fluctuation range of poor channel state, a user is bigger than that of good channel state user with service upper limitation changing. And the tendency of QoE in Figure 5(b) is more stable. The reason behind this phenomenon is that the increasing point is dependent on cache capacity and the number of vehicle providers. If the cache diversity and providers are sufficient, service upper limitation will become bottlenecks and the total data volume received will increase with that. Otherwise, the total data volume received will not increase over the change of service upper limitation. Furthermore,

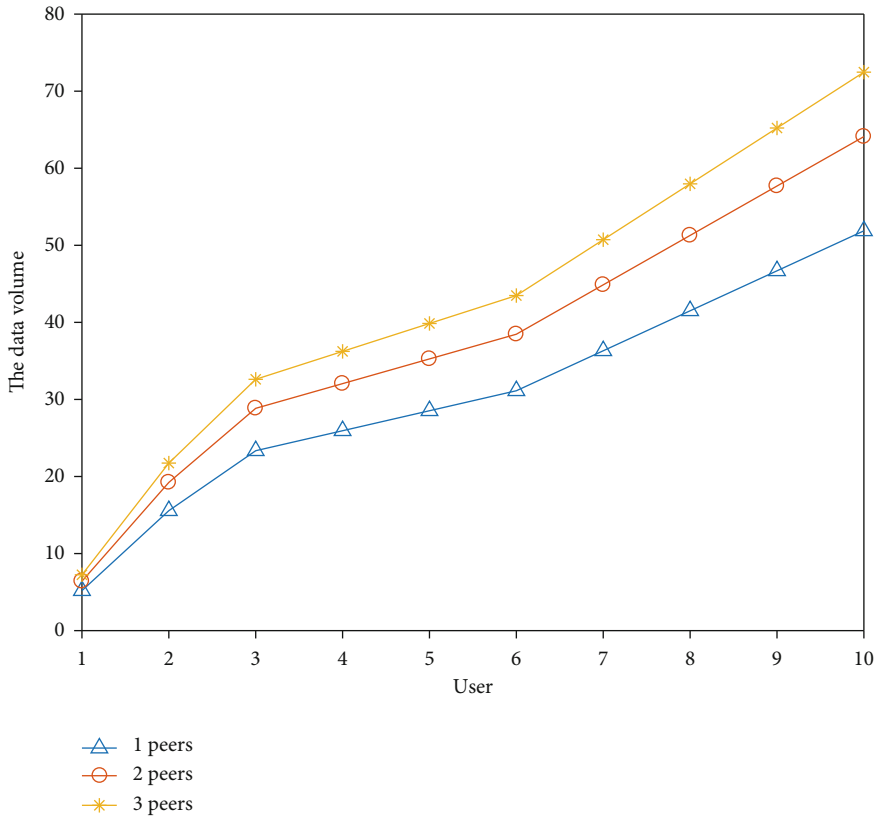


(a) The data volume

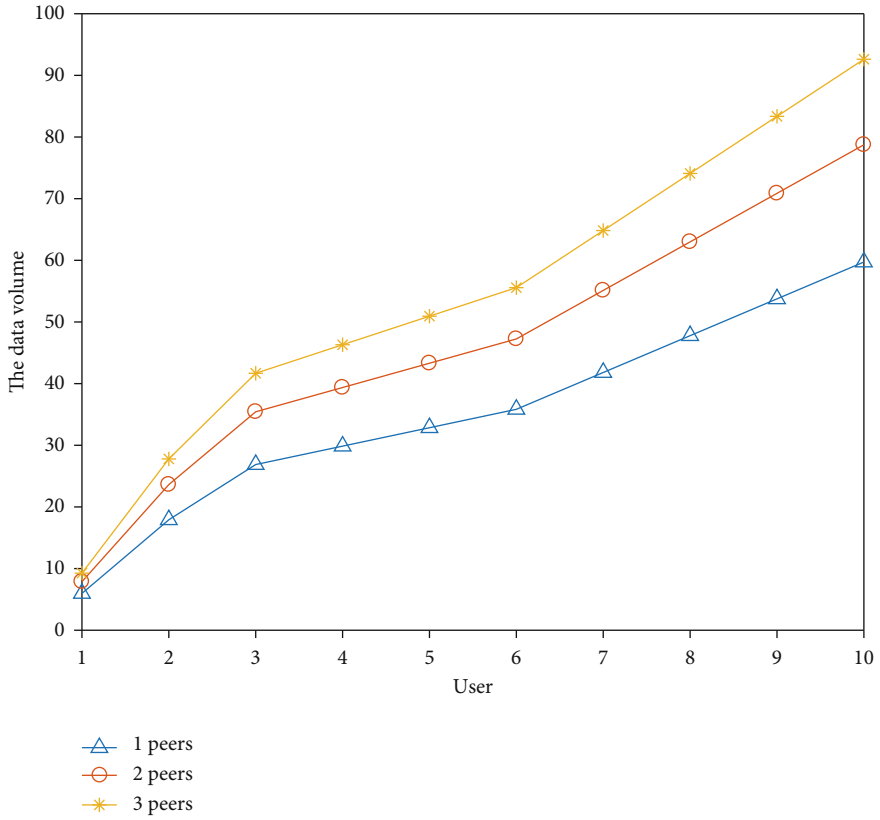


(b) QoE

Figure 5: Impact of the service upper bound.

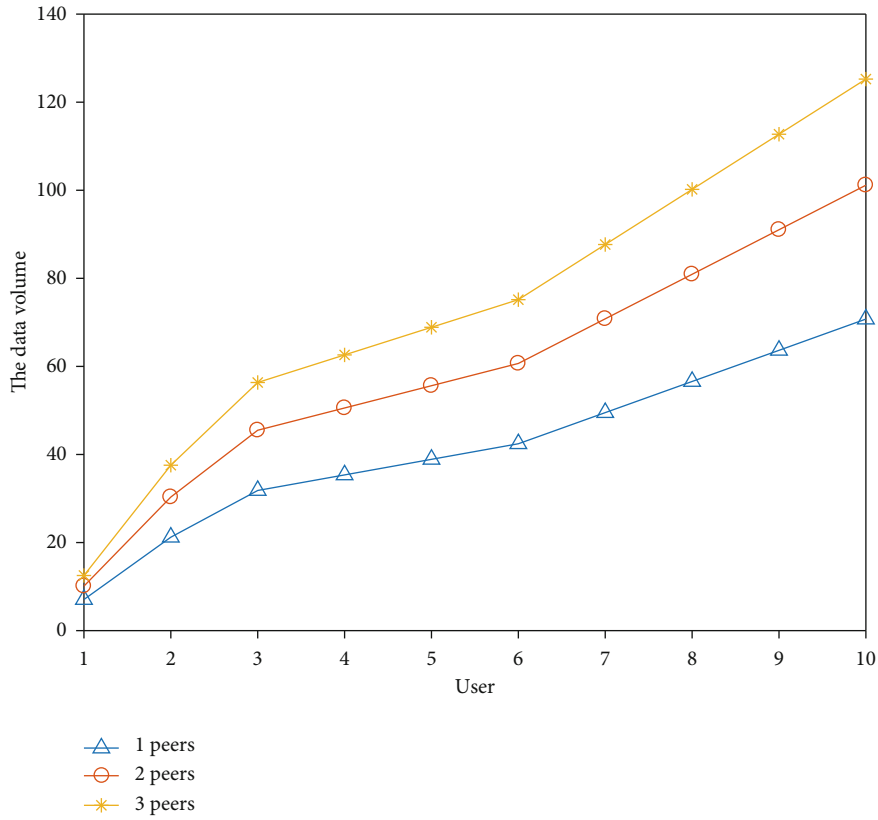


(a) $r = 0.6$

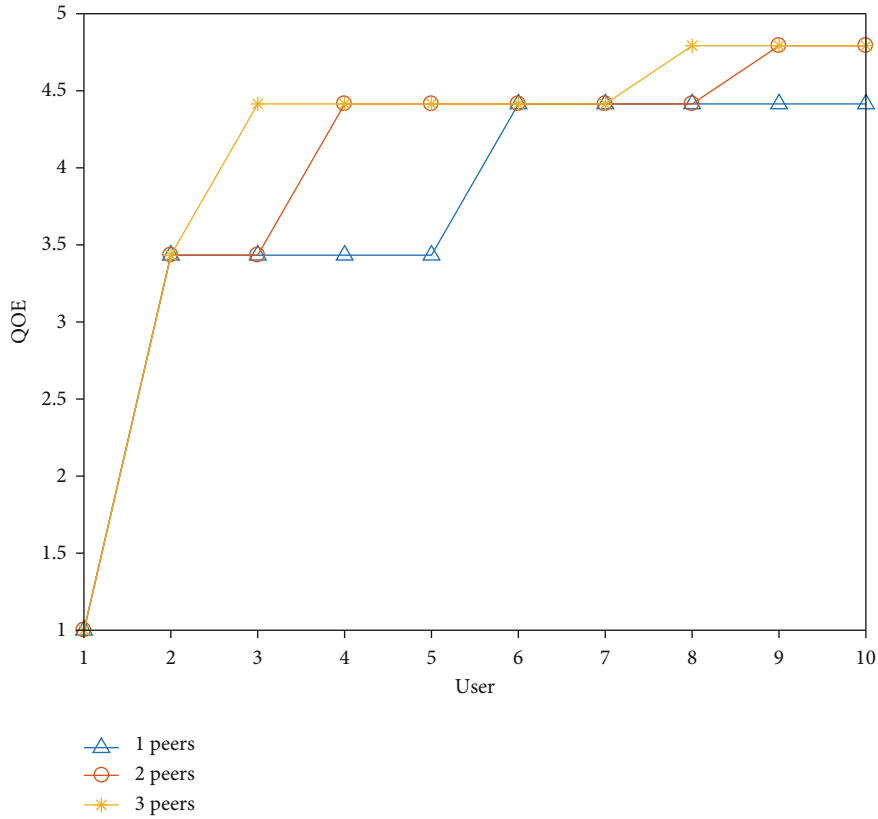


(b) $r = 0.8$

Figure 6: Continued.

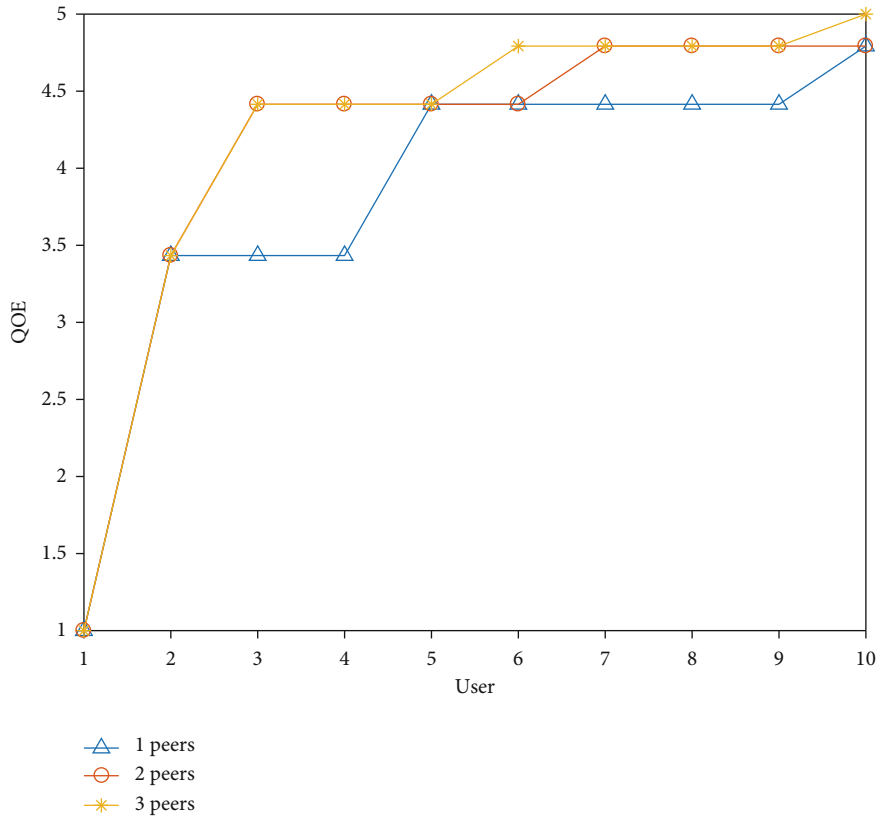


(c) $r = 1.2$

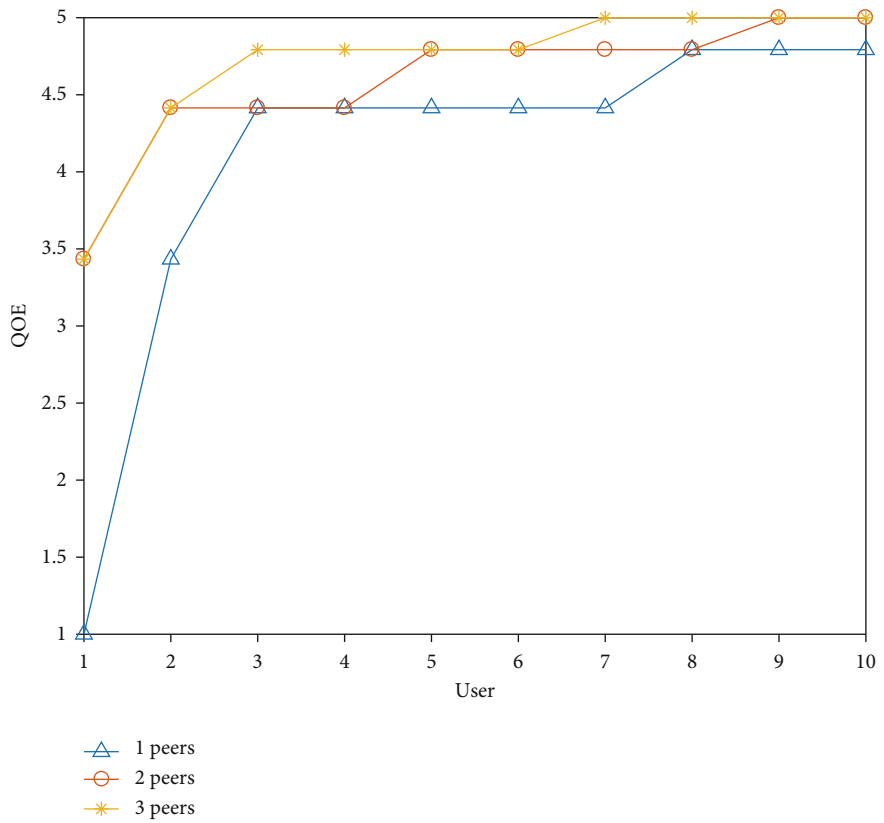


(d) $r = 0.6$

Figure 6: Continued.

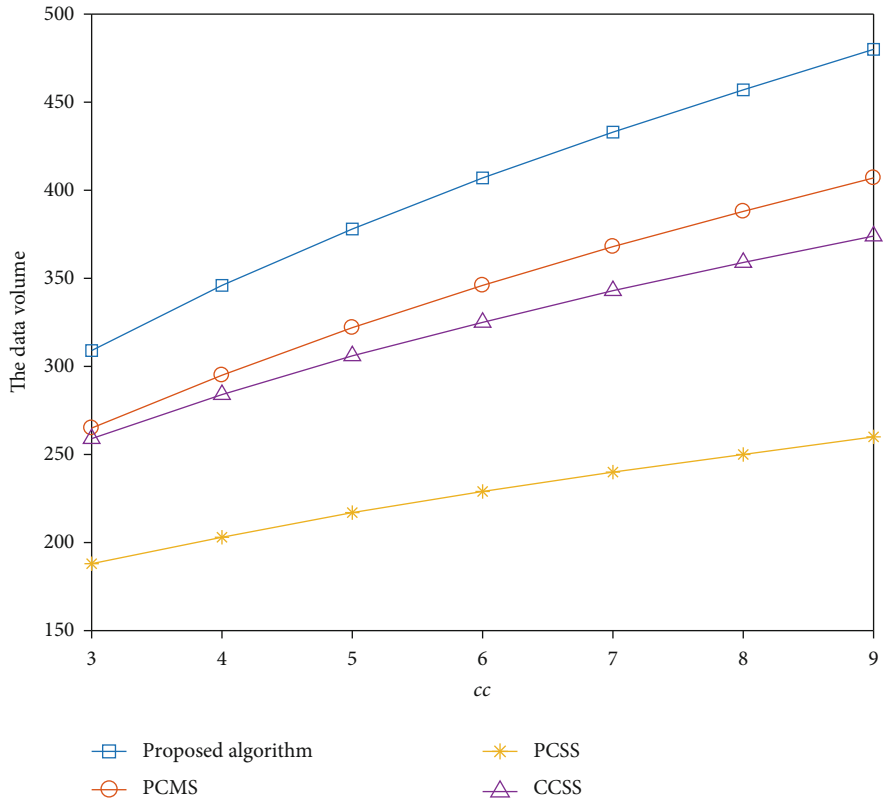


(e) $r = 0.8$

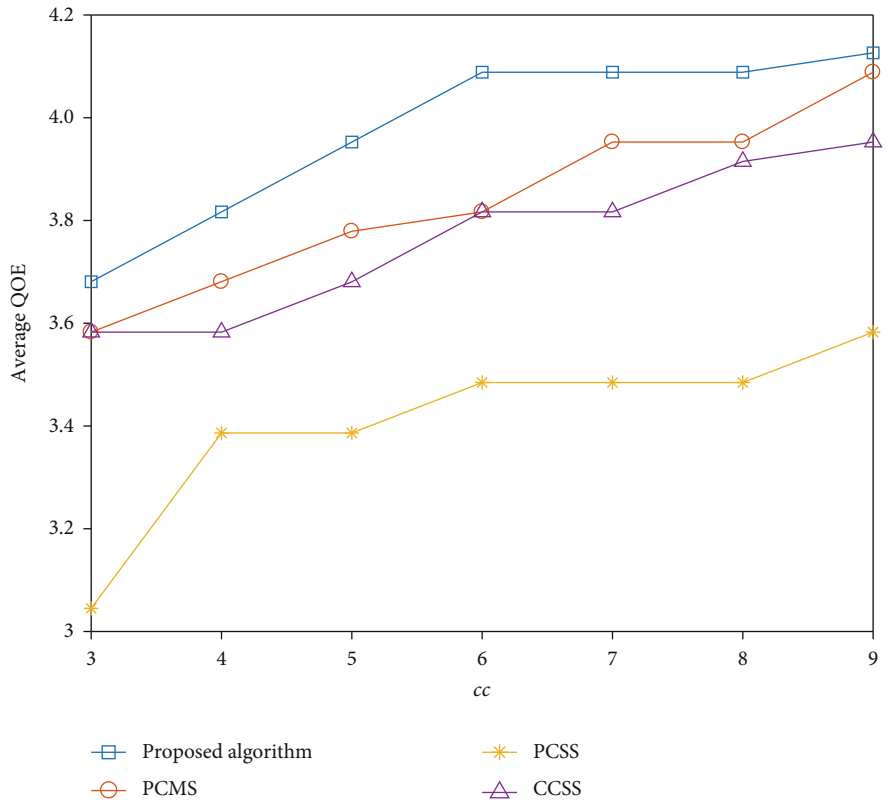


(f) $r = 1.2$

Figure 6: Impact of the receive upper bound.

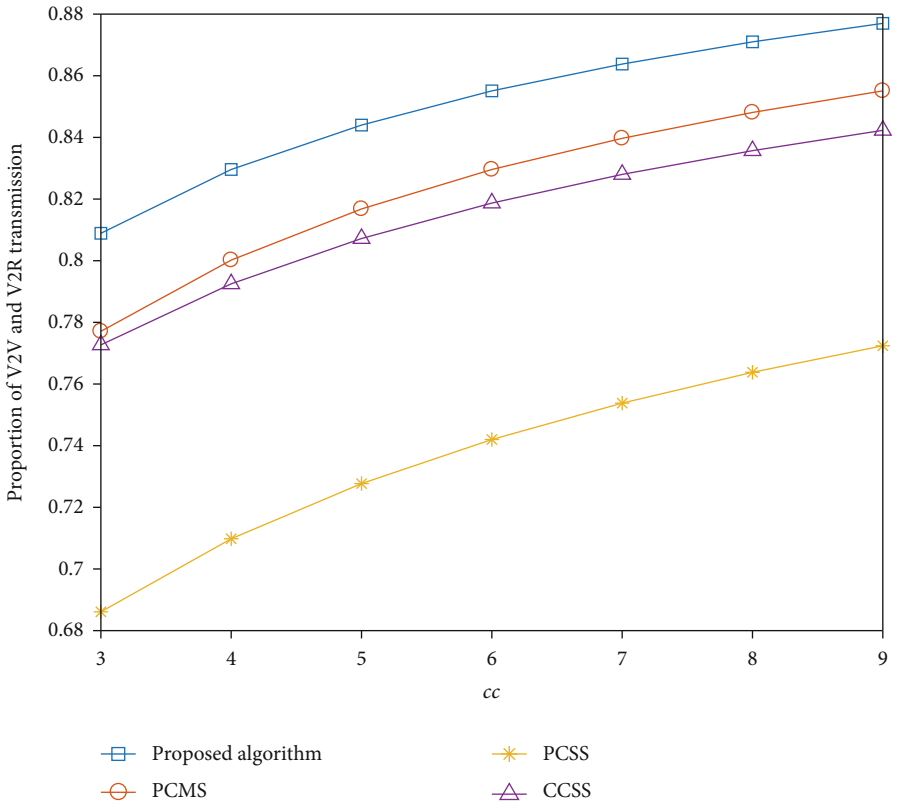


(a) The data volume

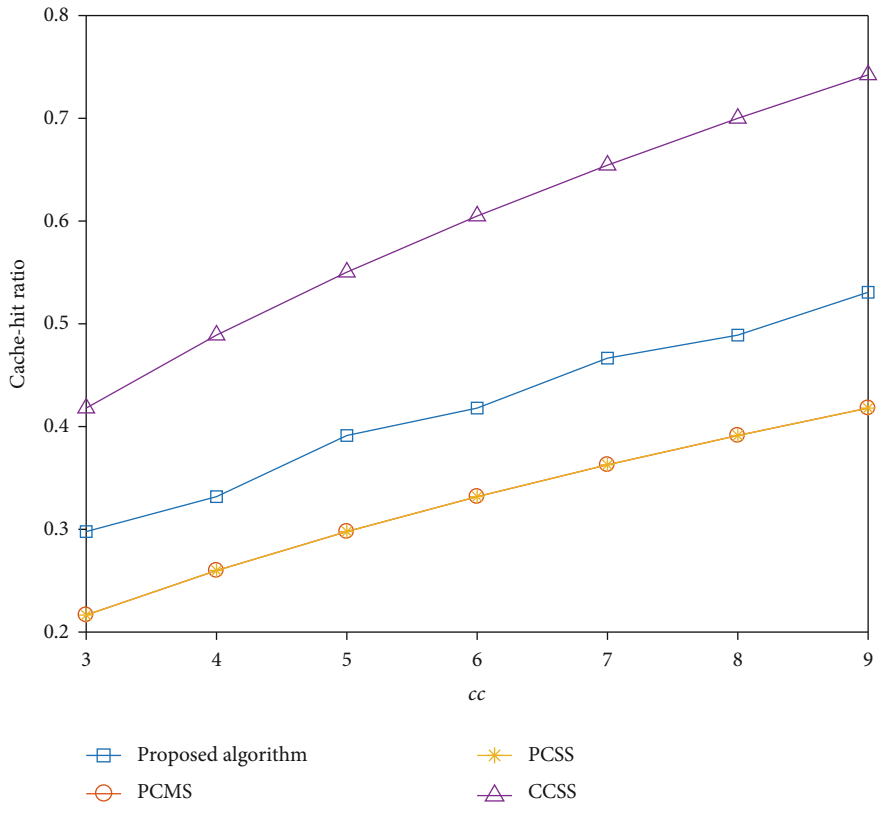


(b) Average QOE

Figure 7: Continued.



(c) Proportion of V2V and V2R transmission



(d) Cache hit ratio

Figure 7: Impact of vehicle cache capacity *cc*.

to attain more data, users with a good channel state take priority in the service quota; when the quota increases, the impact is less than other users.

5.5. Impact of the Receive Upper Limitation. Receive upper bound means the max number of vehicle peers that one requester can receive at the same time. The impact of the receive upper bound over different popularity parameter is shown in Figure 6. From these figures, it can be found that the data volume will increase with the increase of receive upper bound, and the smaller the popularity parameter r, the bigger the range of increase. The reason is that allowing a requester to receive from more vehicle peers can receive more data and ensure the adequacy of the playback buffer. Besides for users, the better the channel quality, the more obvious the data throughput will increase with the increase in the number of peers. When users' preferences are concentrated and the receive upper bound reaches 3, the video quality received by users with good channel quality can attain the high-definition layers. So continuing to increase vehicle peers will not bring significant effects in improving video quality. However, in practical applications, a larger number of vehicle peers providing transmission services to requesters will cause higher cost. Therefore, it is necessary to achieve a good trade-off between high-definition level and the number of vehicle peers.

5.6. Impact of Vehicle Cache Capacity. Figure 7 shows the influence of vehicle cache capacity cc on different transmission strategies in terms of quantitative measurements. From the trend of increase, we can observe that total data throughput, proportion of V2V and V2R transmission, and cache hit ratio all increase with the increase of cc for different transmission strategies. Under the constraint that the received data throughput must reach a certain amount, the growth trend of QoE is slightly slower, but the overall trend is still on the rise. It is because with more caching capacity, more video contents can be satisfied by vehicle peers whose transmission rate is relatively high and less contents are requested from BS. And more cache capacity contributes to caching diversity which will improve the cache hit ratio. Besides, our proposed strategy always outperforms than others for all quantitative measurements regardless of the cache capacity. The reason is that, on the one hand, cache scheme based on popularity does not take coordination into consideration, and the identical content cache for all nodes cannot compensate for each other; on the other hand, the transmission capacity of single transmission source is limited and cannot support high-definition level video decoding. Furthermore, when it comes to the scenario where the capacity is adequate, the gap of QoE and proportion of V2V and V2R transmission for different caching strategy is not obvious. The larger the vehicle cache capacity cc, the closer the QoE value is to 5, the closer the transmission ratio is to 1, and the smaller the gap between different caching strategies.

6. Conclusion

In this paper, we propose a cooperative broadcast optimization to improve the QoE for vehicular video streaming. The cooperative broadcast optimization takes content cache and transmission scheduling policy into account to maximize total data volume received by all requesters. In addition, from the perspective of improving quality level, requesters determine the buffer action according to its current buffer status. The total data throughput can be constructed as an INLP problem which can be converted into some linear integer programming subproblems through the McCormick envelope method and Lagrange relaxation. Numerical simulations demonstrate that our algorithm outperforms other strategies in terms of the data volume, average QoE of all vehicle requesters, and proportion of V2V and V2R transmission and cache hit ratio. For the future work, we will take energy consumption between the requesters and different broadcast candidates into consideration to manage the transmission scheduling association.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the Natural Science Foundation of China (No. 61872104) and the Fundamental Research Funds for the Central Universities in China.

References

[1] GMDT Forecast, "Cisco visual networking index: global mobile data traffic forecast update, 2017–2022," *Update*, vol. 2017, p. 2022, 2019.

[2] S. Boussoufa-Lahlah, F. Semchedine, and L. Bouallouche-Medjkoune, "Geographic routing protocols for Vehicular Ad hoc NETworks (VANETs): a survey," *Vehicular Communications*, vol. 11, pp. 20–31, 2018.

[3] X. Ge, S. Tu, G. Mao, C.-X. Wang, and T. Han, "5g ultra-dense cellular networks," *IEEE Wireless Communications*, vol. 23, no. 1, pp. 72–79, 2016.

[4] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2016.

[5] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[6] Z. Cai and Z. He, "Trading private range counting over big iot data," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 144–153, Dallas, Texas, USA, 2019.

[7] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial iots," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[8] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks: a survey towards private and secure applications," 2021, https://arxiv.org/abs/1612.06637.

[9] Y. Wang, Y. Gao, Y. Li, and X. Tong, "A worker-selection incentive mechanism for optimizing platform-centric mobile crowdsourcing systems," *Computer Networks*, vol. 171, article 107144, 2020.

[10] Z. Sun, Y. Wang, Z. Cai, T. Liu, X. Tong, and N. Jiang, "A two-stage privacy protection mechanism based on blockchain in mobile crowdsourcing," *International Journal of Intelligent Systems*, vol. 36, no. 5, pp. 2058–2080, 2021.

[11] T. Liu, Y. Wang, Y. Li, X. Tong, L. Qi, and N. Jiang, "Privacy protection based on stream cipher for spatiotemporal data in iot," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 7928–7940, 2020.

[12] Y. Wang, Z. Cai, Z.-H. Zhan, B. Zhao, X. Tong, and L. Qi, "Walrasian equilibrium-based multiobjective optimization for task allocation in mobile crowdsourcing," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 4, pp. 1033–1046, 2020.

[13] L. Zhuoran, Y. L. YingjieWang, X. Tong, C. Mu, and C. Yu, "Data-driven many-objective crowd worker selection for mobile crowdsourcing in industrial iot," *IEEE Transactions on Industrial Informatics*, 2021.

[14] C. Xu, W. Quan, A. V. Vasilakos, H. Zhang, and G.-M. Muntean, "Information-centric cost-efficient optimization for multimedia content delivery in mobile vehicular networks," *Computer Communications*, vol. 99, pp. 93–106, 2017.

[15] J. A. F. F. Dias, J. J. P. C. Rodrigues, N. Kumar, and K. Saleem, "Cooperation strategies for vehicular delay-tolerant networks," *IEEE Communications Magazine*, vol. 53, no. 12, pp. 88–94, 2015.

[16] S. Zhou, X. Qichao, Y. Hui, M. Wen, and S. Guo, "A game theoretic approach to parked vehicle assisted content delivery in vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 7, pp. 6461–6474, 2017.

[17] L. Haodong, X. He, D. Miao, X. Ruan, Y. Sun, and K. Wang, "Edge qoe: computation offloading with deep reinforcement learning for internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9255–9265, 2020.

[18] X. He, L. Haodong, D. Miao, Y. Mao, and K. Wang, "Qoe-based task offloading with deep reinforcement learning in edge-enabled internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2252–2261, 2021.

[19] X. He, K. Wang, and W. Xu, "Qoe-driven content-centric caching with deep reinforcement learning in edge-enabled iot," *IEEE Computational Intelligence Magazine*, vol. 14, no. 4, pp. 12–20, 2019.

[20] X. He, K. Wang, H. Huang, and B. Liu, "Qoe-driven big data architecture for smart city," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 88–93, 2018.

[21] W. Zhao, Y. Qin, D. Gao, C. H. Foh, and H.-C. Chao, "An efficient cache strategy in information centric networking vehicle-to-vehicle scenario," *IEEE Access*, vol. 5, pp. 12657–12667, 2017.

[22] S. Kumar and S. Misra, "Joint content sharing and incentive mechanism for cache-enabled device-to-device networks," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 5, pp. 4993–5002, 2021.

[23] Y. Zhang, C. Li, T. H. Luan, Y. Fu, and H. Wang, "Prediction based vehicular caching: where and what to cache?," *Mobile Networks and Applications*, vol. 25, no. 2, pp. 760–771, 2020.

[24] Y. Li, M. Jiang, Q. Zhang, and J. Qin, "Cache content placement optimization in non-orthogonal multiple access networks," *IEEE Transactions on Communications*, vol. 68, no. 7, pp. 4580–4591, 2020.

[25] D. Zhu, L. Hancheng, G. Zhuojia, L. Yujiao, and F. Guo, "Joint power allocation and caching for SVC videos in heterogeneous networks," in *2018 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–7, Abu Dhabi, UAE, 2018.

[26] H. Binbin, L. Fang, X. Cheng, and L. Yang, "Invehicle caching (IV-cache) via dynamic distributed storage storage relay (D2SR) in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 843–855, 2019.

[27] C. Chen, L. Chen, L. Liu et al., "Delay-optimized v2v-based computation offloading in urban vehicular edge computing and networks," *IEEE Access*, vol. 8, pp. 18863–18873, 2020.

[28] A. Al-Hilo, D. Ebrahimi, S. Sharafeddine, and C. Assi, "Revenue-driven video delivery in vehicular networks with optimal resource scheduling," *Vehicular Communications*, vol. 23, article 100215, 2020.

[29] Y. Sun, L. Xu, Y. Tang, and W. Zhuang, "Traffic offloading for online video service in vehicular networks: a cooperative approach," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 8, pp. 7630–7642, 2018.

[30] X. Zeyu, Y. Cao, W. Wang, T. Jiang, and Q. Zhang, "Incentive mechanism for cooperative scalable video coding (svc) multicast based on contract theory," *IEEE Transactions on Multimedia*, vol. 22, no. 2, pp. 445–458, 2020.

[31] Y. Yan, B. Zhang, and C. Li, "Network coding aided collaborative real-time scalable video transmission in d2d communications," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 6203–6217, 2018.

[32] H. Zhou, X. Wang, Z. Liu, Y. Ji, and S. Yamada, "Resource allocation for svc streaming over cooperative vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 7924–7936, 2018.

[33] C. Xu, W. Ren, L. Yu, T. Zhu, and K.-K. R. Choo, "A hierarchical encryption and key management scheme for layered access control on h. 264/svc bitstream in the internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8932–8942, 2020.

[34] L. Wang, C.-S. Lam, and M.-C. Wong, "Multifunctional hybrid structure of svc and capacitive grid-connected inverter (svc// cgci) for active power injection and nonactive power compensation," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 3, pp. 1660–1670, 2019.

[35] S.-H. Shen, "Efficient svc multicast streaming for video conferencing with sdn control," *IEEE Transactions on Network and Service Management*, vol. 16, no. 2, pp. 403–416, 2019.

[36] Z. Zhu, Y. Xu, and Z. Su, "A reputation-based cooperative content delivery with parking vehicles in vehicular ad-hoc networks," *Peer-to-Peer Networking and Applications*, vol. 14, no. 3, pp. 1531–1547, 2021.

[37] S. Kumar, R. Devaraj, A. Sarkar, and A. Sur, "Client-side QoE management for SVC video streaming: an FSM supported design approach," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 1113–1126, 2019.

[38] M. Xing, S. Xiang, and L. Cai, "A real-time adaptive algorithm for video streaming over multiple wireless access networks,"

IEEE Journal on Selected Areas in Communications, vol. 32, no. 4, pp. 795–805, 2014.

[39] M. Zhao, X. Gong, J. Liang et al., "Qoe-driven optimization for cloud-assisted DASH-based scalable interactive multiview video streaming over wireless network," *Signal Processing: Image Communication*, vol. 57, pp. 157–172, 2017.

[40] G. Cofano, L. De Cicco, and S. Mascolo, "Modeling and design of adaptive video streaming control systems," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 1, pp. 548–559, 2018.

[41] D. Bezerra, M. Ito, W. Melo, D. Sadok, and J. Kelner, "Dbuffer: a state machine oriented control system for dash," in *2016 IEEE Symposium on Computers and Communication (ISCC)*, pp. 861–867, Messina, Italy, 2016.

[42] C. Celes, A. Boukerche, and A. A. F. Loureiro, "Mobility trace analysis for intelligent vehicular networks," *ACM Computing Surveys (CSUR)*, vol. 54, no. 3, pp. 1–38, 2021.

[43] W. Jiyan, C. Yuen, N.-M. Cheung, and J. Chen, "Delay-constrained high definition video transmission in heterogeneous wireless networks with multi-homed terminals," *IEEE Transactions on Mobile Computing*, vol. 15, no. 3, pp. 641–655, 2016.

[44] G. Feng, Y. Zhang, J. Lin, H. Wang, and L. Cai, "Joint optimization of downlink and D2D transmissions for SVC streaming in cooperative cellular networks," *Neurocomputing*, vol. 270, pp. 178–187, 2017.

[45] H. Hu, X. Zhu, Y. Wang, R. Pan, J. Zhu, and F. Bonomi, "Proxy-based multi-stream scalable video adaptation over wireless networks using subjective quality and rate models," *IEEE Transactions on Multimedia*, vol. 15, no. 7, pp. 1638–1652, 2013.

[46] T. Westerlund, A. Lundell, and J. Westerlund, "On convex relaxations in nonconvex optimization," *Chemical Engineering Transactions*, vol. 24, pp. 331–336, 2011.

[47] Q. Ploussard, L. Olmos, and A. Ramos, "An operational state aggregation technique for transmission expansion planning based on line benefits," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 2744–2755, 2017.

[48] K. Bernhard and J. Vygen, *Combinatorial Optimization: Theory and Algorithms*, vol. 2005Springer, Third Edition edition, 2008.

[49] D. P. Bertsekas, "Nonlinear programming," *Journal of the Operational Research Society*, vol. 48, no. 3, pp. 334–334, 1997.

[50] S. H. Low and D. E. Lapsley, "Optimization flow control. I. Basic algorithm and convergence," *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 861–874, 1999.

[51] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*, Cambridge university press, 2004.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 1567019, 16 pages
https://doi.org/10.1155/2021/1567019

Research Article

Broadcast Proxy Reencryption Based on Certificateless Public Key Cryptography for Secure Data Sharing

Won-Bin Kim,[1] Su-Hyun Kim,[2] Daehee Seo,[3] and Im-Yeong Lee[1]

[1]*Department of Software Convergence, Soonchunhyang University, Asan 31538, Republic of Korea*
[2]*National IT Industry Promotion Agency, Jincheon 27872, Republic of Korea*
[3]*Faculty of Artificial Intelligence and Data Engineering, Sangmyung University, Seoul 03016, Republic of Korea*

Correspondence should be addressed to Im-Yeong Lee; imylee@sch.ac.kr

Received 30 August 2021; Accepted 9 November 2021; Published 16 December 2021

Academic Editor: Yingjie Wang

Broadcast proxy reencryption (BPRE), which combines broadcast encryption (BE) and proxy reencryption (PRE), is a technology used for the redistribution of data uploaded on the cloud to multiple users. BPRE reencrypts data encrypted by the distributor and then uploads it to the cloud into a ciphertext that at a later stage targets multiple recipients. As a result of this, flexible data sharing is possible for multiple recipients. However, various inefficiencies and vulnerabilities of the BE, such as the recipient anonymity problem and the key escrow problem, also creep into BPRE. Our aim in this study was to address this problem of the existing BPRE technology. The partial key verification problem that appeared in the process of solving the key escrow problem was solved, and the computational efficiency was improved by not using bilinear pairing, which requires a lot of computation time.

1. Introduction

The cloud technology allows users to access a range of services anytime, anywhere. Depending on the requirement of users, the cloud provides a range of services, including software-as-a-service (SaaS), platform-as-a-service (PaaS), and infrastructure-as-a-service (IaaS). Recently, it has become easier to use the cloud owing to the rapid development of communication and computing technology, and consequently, the cloud has been introduced and used in various domains and environments.

In general, the cloud technology is recognized as a remote storage environment or as a software that can be used without the need for installing it on a local device. Microsoft's Office 365 products or Adobe's CC product line can be seen as cloud-based software, and Google Drive and Microsoft's One Drive are representative examples of cloud-based remote storage. These products, running on the Internet, provide functions that local storage fails to provide and can be easily used anytime, anywhere. However, the cloud is not limited to the range of services described above. Its scope has expanded into more diverse and extensive areas and domains in recent times.

Cloud technology requires an Internet connection to work. In other words, the cloud is an online technology. As a result of this, the working environment of the cloud can be shared online at any given time. For example, the cloud in an enterprise environment is not just for each employee to store their own data. It can also be used by employees as an efficient tool to share their work with each other. In light of this, there is an increasingly urgent need for technologies that can store and share data through such a cloud [1, 2].

The cloud essentially is a proxy server, that is, it is a remote server that can be accessed and used via a network. However, the proxy server responds to the request but it is always considered a semitrusted server because it always wants to know its contents. Therefore, for data to be stored safely in the cloud, data encryption is essential. In addition, to further share the data stored in the cloud, the recipient must be able to easily decrypt the encrypted data. Moreover, for the data sender to decrypt the encrypted data, a decryption key is required. However, the two most popular methods for this, symmetric key encryption and asymmetric key encryption, suffer from the key distribution problem.

Therefore, proxy reencryption (PRE) has been proposed to securely share data without exposing the data contents and decryption keys to risks during the data sharing process.

PRE reencrypts data encrypted using the sender's public key in the proxy so that the receiver can decrypt it by using their own private key. As a result, the private keys of the sender and receiver are not exposed to risks during data sharing and the cloud, too, has no access to the contents of the encrypted data. However, because this PRE is based on one-to-one transmission, it is not suitable for environments where the same data are distributed to multiple recipients (for example, environments such as update servers or secure e-mail). In such a scenario, if the existing proxy reencryption is used, reencryption key generation and reencryption must be performed as many times as the number of recipients.

Broadcast proxy reencryption (BPRE) combines broadcast encryption (BE), which enables sending of the same data to multiple recipients by using a single encryption, and PRE. Therefore, BPR can reencrypt encrypted data stored in the cloud and distribute it to multiple recipients, enabling flexible data sharing. However, because BPRE is an encryption method based on BE, it also suffers from some security vulnerabilities that are typical to BE. For example, the lack of receiver anonymity in BE, wherein the identity of a specific receiver in a communication channel gets exposed, leading to serious privacy issues, is also present in BPRE. In addition, the security threat caused by BE's public key generation method also appears in BPRE.

Typically, there is a key escrow problem that appears in ID-based cryptography (IBC) and the certificateless (CL) cryptography can be applied to solve this problem. In addition, the partial key verification problem of the CL cryptography must also be considered. Finally, existing BPRE schemes were designed using a bilinear pairing operation. However, bilinear pairing operation is a time-consuming process, which increases the computing cost in BPRE. Therefore, the goal of this study is to provide a relatively safe environment for data sharing by solving the security threats presented above, as well as to develop a more efficient BPRE technology by leaving out the bilinear pairing operation.

2. Related Works

This section describes related studies and theoretical constructs for understanding the concepts discussed in the present study.

2.1. Data Sharing. Data sharing refers to the sharing of data owned by the data owner (sender) with other users. Encryption becomes essential when sharing data safely over a network [3–5]. In the absence of encryption, data may be exposed or tampered with by eavesdropping events during the communication process. In addition, to share encrypted data, the receiver must be able to decrypt the data. If the sender encrypts data using the symmetric key method, the symmetric key must be safely delivered to the receiver. This, however, is difficult to achieve. Conversely, when using the asymmetric key (public key) method, the sender and receiver can share data by exchanging only the public key with each other. However, in both of the above two methods, the sender and the receiver must both remain online until the data transmission is completed, which is not possible at times. To solve this problem, a data sharing method using a cloud (proxy) has been proposed: after uploading data to the cloud, the sender can use the method of allowing access to the data stored in the cloud according to the request of the receiver. Encryption is indispensable even when using this method to ensure the contents of the data is not exposed. However, to decrypt data encrypted with the sender's public key using the receiver's private key, it is necessary to encrypt the data with the receiver's public key after decryption in the cloud; however, during this process, the data are inevitably exposed to the cloud. Therefore, PRE has been proposed to ensure that the private key or the contents of the data are not exposed while sharing data through the cloud.

2.2. Proxy Reencryption. PRE delegates decryption authority to other users by reencrypting data through a proxy represented by the cloud. As shown in Figure 1, the sender encrypts data with his/her public key, uploads it to the cloud, and generates a reencryption key at the request of the receiver and sends it to the cloud. Upon receiving the ciphertext and the reencryption key, the cloud reencrypts the ciphertext to generate a reencrypted ciphertext and transmits it to the receiver. PRE was first introduced in 1998 by Blaze et al. [6]. Since then, a number of traditional public key cryptography (PKC) PREs have been proposed [7–17, 18]. Such a PRE requires a public key certificate to prove the validity of the public key. However, the generation and storage of public key certificates involve considerable overhead. To address this, several ID-based PRE (IBPRE) methods [19] have been proposed [14, 20, 21]. However, the key generation center (KGC) generates the full private key of each user in IBPRE, which gives rise to the key escrow problem. To solve the key escrow problem, certificateless PRE (CL-PRE) has been proposed [22–24, 25]. In CL-PRE, the KGC does not generate the user's full private key but generates a partial key and delivers it to the user. As a result, the KGC cannot know the user's private key.

2.3. Broadcast Proxy Reencryption. Broadcast encryption (BE) is a technique first introduced by Berkovits [26]. BE is an access control technology designed to securely transmit data such as digital media, notifications or messages, and distance education to multiple recipients. BE can be divided into a symmetric broadcast encryption method and an asymmetric broadcast encryption method according to an encryption method. The symmetric broadcast encryption method is a method of delivering data to multiple receivers using a symmetric key method. Representative examples include Berkovits' scheme [26], Naor et al.'s scheme [27], and Halevy and Shamir's scheme [28]. This symmetric broadcast encryption method makes it difficult to distribute and manage keys.

On the other hand, asymmetric broadcast encryption is a method of delivering data to multiple recipients using a public key method. Therefore, the roles of encryption and decryption can be distinguished by utilizing the easy key

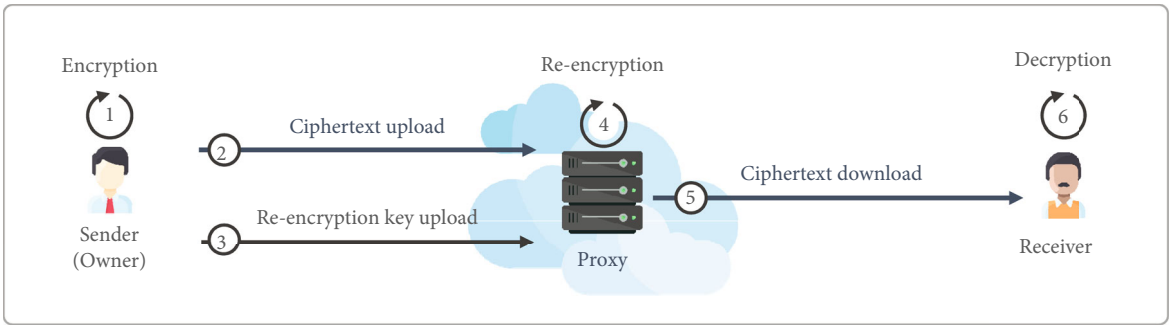


Figure 1: Basic form of proxy reencryption.

distribution and management, which are the advantages of the existing public key encryption method. Asymmetric broadcast encryption was first proposed by Dodis and Fazio [29] in 2002. However, this scheme has the disadvantage that the size of the encryption key is too large. In [30], Delerablée et al. proposed a scheme to perform BE using a dynamic method to target unpredictable users. Since then, BE schemes such as [31–34, 35] have been proposed.

Meanwhile, with the development of communication and storage technologies, the movement to utilize cloud storage has gradually increased. Also, a method for sharing data stored in cloud storage to other users was required. However, in order to make the encrypted data stored in the cloud available to other users, it is difficult to reencrypt after decryption or to deliver the decryption key. These problems increase the network load and reduce the efficiency. Therefore, PRE was proposed to solve this problem and research was conducted to deliver data to multiple recipients using cloud storage by combining this PRE with BE.

Chu et al. first proposed CPBRE by combining conditional proxy reencryption with BE [36]. Since then, various broadcast proxy reencryption (BPRE) has been proposed as shown in Figure 2. BPRE, proposed by Wang et al. in 2009, provides recipient anonymity. Also, data distribution is controlled by the KGC or broadcast center (BC). However, it requires a high computational overhead by using bilinear pairing and a key escrow problem also appears. Various methods of research were also conducted in the relatively recently proposed studies of Maiti and Misra [37], Sun et al. [38], Yin et al. [39], and Chunpeng et al. [40]. However, both of these methods use bilinear pairing and incur high computational overhead. In addition, there is a problem that the key escrow problem occurs.

3. Preliminaries

This section describes the basic environment and settings used to understand the scheme proposed in this study. To this end, the system model, security requirements, and algorithm used in the proposed scheme are explained.

3.1. System Model. The system model used in this study, shown in Figure 3, comprises a *sender*, *receiver*, *cloud* (proxy), and the *KGC*.

(i) *Sender*: the sender is the owner of the data and the user with whom the data are shared. The sender encrypts the plaintext with his/her public key and uploads it to the cloud. Then, to distribute the data, a reencryption key is generated and transmitted to the cloud. The sender can also download the data that he/she uploaded to the cloud and decrypt it with his/her private key to obtain the plaintext

(ii) *Receiver*: the receiver receives sender's data. The receiver may receive the reencrypted ciphertext from the cloud and decrypt the data with his/her private key to obtain the plaintext

(iii) *Cloud* (proxy): the cloud is assumed to be the same object as the remote proxy server. Because the cloud is a semitrusted server, it comes with a danger of data leakage. Therefore, the sender must apply data encryption to safely store data in the cloud. In addition, during data sharing, the plaintext or the user's private key should not be exposed to the cloud. Finally, users with legitimate rights should be able to access and use data in the cloud at any time

(iv) *Key generation center* (KGC): the KGC is an object that generates and issues a user key. Although the KGC is involved in generating each user's private key, in this study, to solve the key escrow problem, the KGC does not generate the user's full private key but generates a partial key and delivers it to each user. In addition, all users must use the public parameters created by the KGC to perform data encryption, decryption, and reencryption.

3.2. Security Requirements. This study consists of seven security requirements. The details are as follows:

(i) *Confidentiality*: the data that are kept in the proxy and the data delivered through the proxy shall not be unknown other than the authorized user. To do this, the data must be encrypted using the encryption key and the user who does not have a legitimacy decryption key should not be able to decrypt the contents

(ii) *Integrity*: data uploaded and shared by the sender must not be changed without permission in the

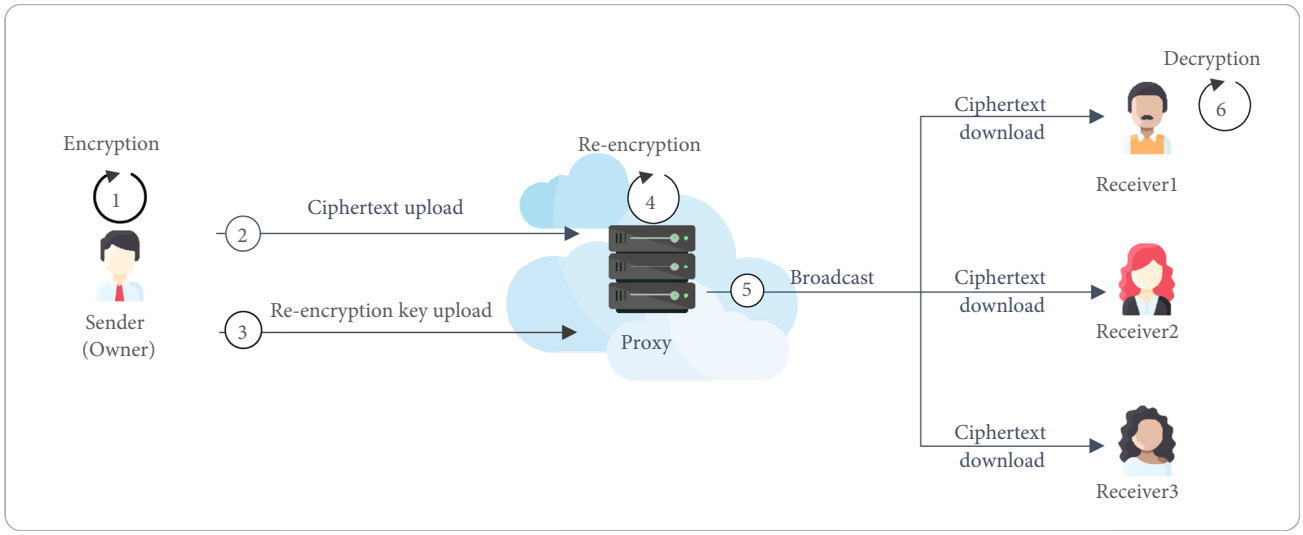


Figure 2: Broadcast proxy reencryption.

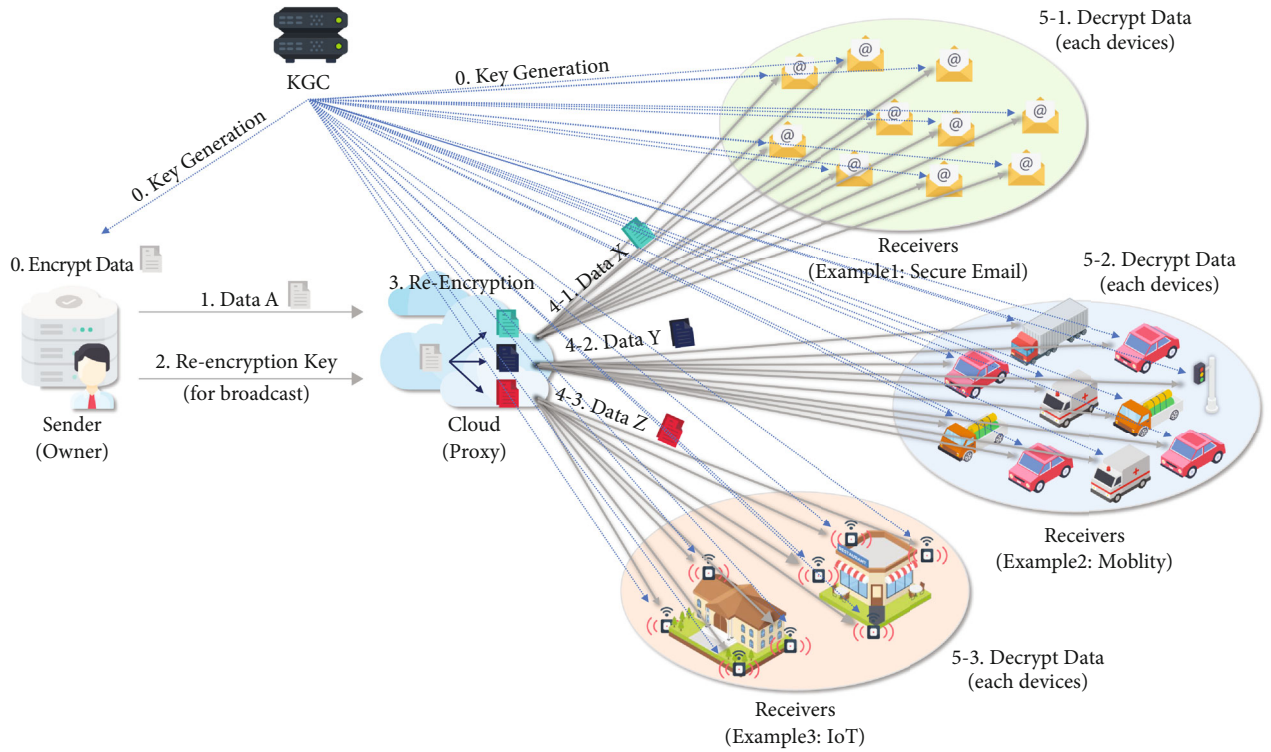


Figure 3: Form of the proposed system model.

process of being delivered to the cloud and the receiver and stored in the cloud. If at all the contents are changed, the sender or receiver who shares the data must be made aware of the change

(iii) *Key escrow problem*: all users who want to use the cloud must communicate with the KGC to generate a private key and public key pair. In this process, the KGC generates a user's full private key and the KGC may arise the user's authority. This problem is called a key escrow problem, and a method for solving this problem is required

(iv) *Partial key verifiability*: to solve the previously described key escrow problem, a key generation method in the form of a partial key can be used. In this case, each user must be able to verify whether the partial key generated and issued by the KGC to each user is legitimately generated by the correct KGC

(v) *Receiver anonymity*: the reencrypted ciphertext in cloud storage can be decrypted by a number of designated receivers. For this purpose, the reencryption key and reencrypted ciphertext include information generated by the public key of each receiver. However, privacy issues arise when such information allows a particular recipient or third party to identify another receiver

(vi) *Decryption fairness*: each legitimate receiver designated by the sender can decrypt the reencrypted ciphertext. However, in this process, a specific receiver should not be discriminated against or disadvantaged in the decryption process by a specific receiver or a third party

3.3. Algorithms. A total of 10 algorithms were used in the scheme proposed in this study. The purpose and details of each algorithm are as follows.

(i) Setup $(\lambda) \longrightarrow (\mathrm{msk}, \mathrm{mpk})$: this algorithm is performed by the KGC, which generates KGC's master secret key msk and master public key mpk for each user to use the cloud and publishes the mpk

(ii) Set-secret-value $(\mathrm{mpk}) \longrightarrow (T_i, \mathrm{ID}_i)$: this algorithm is performed by the user, wherein user i generates T_i using randomly selected t_i and mpk and sends it to the KGC along with ID_i

(iii) Partial-key-extract $(T_i, \mathrm{ID}_i, \mathrm{msk}, \mathrm{mpk}) \longrightarrow (R_i, k_i)$: this algorithm is performed by the KGC, which generates partial keys (R_i, k_i) of user i using T_i and ID_i transmitted by user i and its own msk and mpk and delivers it to user i

(iv) Set-private-key $(t_i, R_i, k_i, \mathrm{mpk}) \longrightarrow \mathrm{sk_i}$: this algorithm is performed by the user, wherein user i generates his/her own private key sk_i using the partial keys (R_i, k_i) received from the KGC. The generated private key sk_i was kept secure

(v) Set-public-key $(t_i, R_i, \mathrm{mpk}) \longrightarrow \mathrm{pk}_i$: this algorithm is performed by the user, wherein user i generates his/her public key pk_i using the partial key $(R_i, k_i$) received from the KGC and the secret value t_i generated by the user. The generated public key p k_i is made public so that anyone can use it

(vi) Enc $(\mathrm{pk}_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}}, m, \mathrm{mpk}) \longrightarrow \mathrm{CT}$: this algorithm is performed by the sender, wherein sender $\mathbb{S}$ encrypts his/her data $m \in M$ using public key $\mathrm{pk}_{\mathbb{S}}$ to obtain ciphertext CT and uploads it to the cloud

(vii) Re-key-gen $(\mathrm{sk}_{\mathbb{S}}, \mathrm{pk}_{\mathbb{R}}, \mathrm{ID}_{\mathbb{S}}, \mathrm{ID}_{\mathbb{R}}, \mathrm{mpk}) \longrightarrow \mathrm{rk}_{\mathbb{S}\longrightarrow\mathbb{R}}$: this algorithm is performed by the sender, wherein sender $\mathbb{S}$ specifies a receiver set $\mathbb{R} = (\curlywedge_1, \curlywedge_2, \cdots, \curlywedge_n)$ of receivers $\curlywedge_j (1 \le j \le n)$ to share their data with, generates a reencryption key for $\mathbb{R}$, and delivers it to the cloud

(viii) Re-enc $(\mathrm{CT}, \mathrm{rk}_{\mathbb{S}\longrightarrow\mathbb{R}}, \mathrm{mpk}) \longrightarrow \mathrm{CT}_R$: this algorithm is performed by the cloud, wherein the cloud reencrypts ciphertext CT of sender $\mathbb{S}$ using reencryption key $\mathrm{rk}_{\mathbb{S}\longrightarrow\mathbb{R}}$ of sender $\mathbb{S}$ to obtain reencrypted ciphertext CT_R

(ix) Dec-1 $(\mathrm{CT}, \mathrm{sk}_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}}, \mathrm{mpk}) \longrightarrow m$: this algorithm is performed by the sender, wherein sender $\mathbb{S}$ downloads his/her ciphertext CT stored in the cloud and then uses his/her private key $\mathrm{sk}_{\mathbb{S}}$ to decrypt it to obtain plaintext m

(x) Dec-2 $(\mathrm{CT}_R, \mathrm{sk}_j, \mathrm{ID}_j, \mathrm{mpk}) \longrightarrow m$: this algorithm is performed by the receiver, wherein receiver $\curlywedge_j$ downloads ciphertext CT_R stored in the cloud and then uses his/her private key sk_j to decrypt it to obtain plaintext m

4. Proposed CL Broadcast Proxy Reencryption

This section describes the scheme proposed in this study. For this, a technical overview, system parameters, and algorithm construction are described.

4.1. Technical Overview. The basic model of BRE, shown in Figure 4, can be broadly divided into four phases: *a setup phase*, *key generation phase*, *data storage phase*, and *data broadcast phase*. More details about these phases are presented in Sections 4.2 and 4.3.

4.2. System Parameters. The following are the system parameters used in this proposed scheme.

(i) $*$: participants (KGC, sender $\mathbb{S}$, receiver set $\mathbb{R}$, receiver $\curlywedge_j$, and user i)

(ii) p, q: λ-bit prime integer

(iii) E: elliptic curve

(iv) F_q: finite field for q

(v) λ: security parameter

(vi) l_1, l_2: length of message space (determined by the λ)

(vii) P: random generator in G_q $(P \in G_q)$

(viii) G: additive group on elliptic curve E

(ix) G_q: subgroup of G with prime order q

(x) ID_*: identity of participant $*$ $(\mathrm{ID}_* \in \{0,1\}^*)$

(xi) msk: KGC's system master secret key

(xii) mpk: KGC's system master public key

(xiii) sk_i: user i's full private key

(xiv) pk_i: user i's full public key

(xv) $\mathrm{rk}_{\mathbb{S}\longrightarrow\mathbb{R}}$: reencryption key (sender $\mathbb{S}$ delegates to receiver set $\mathbb{R}$)

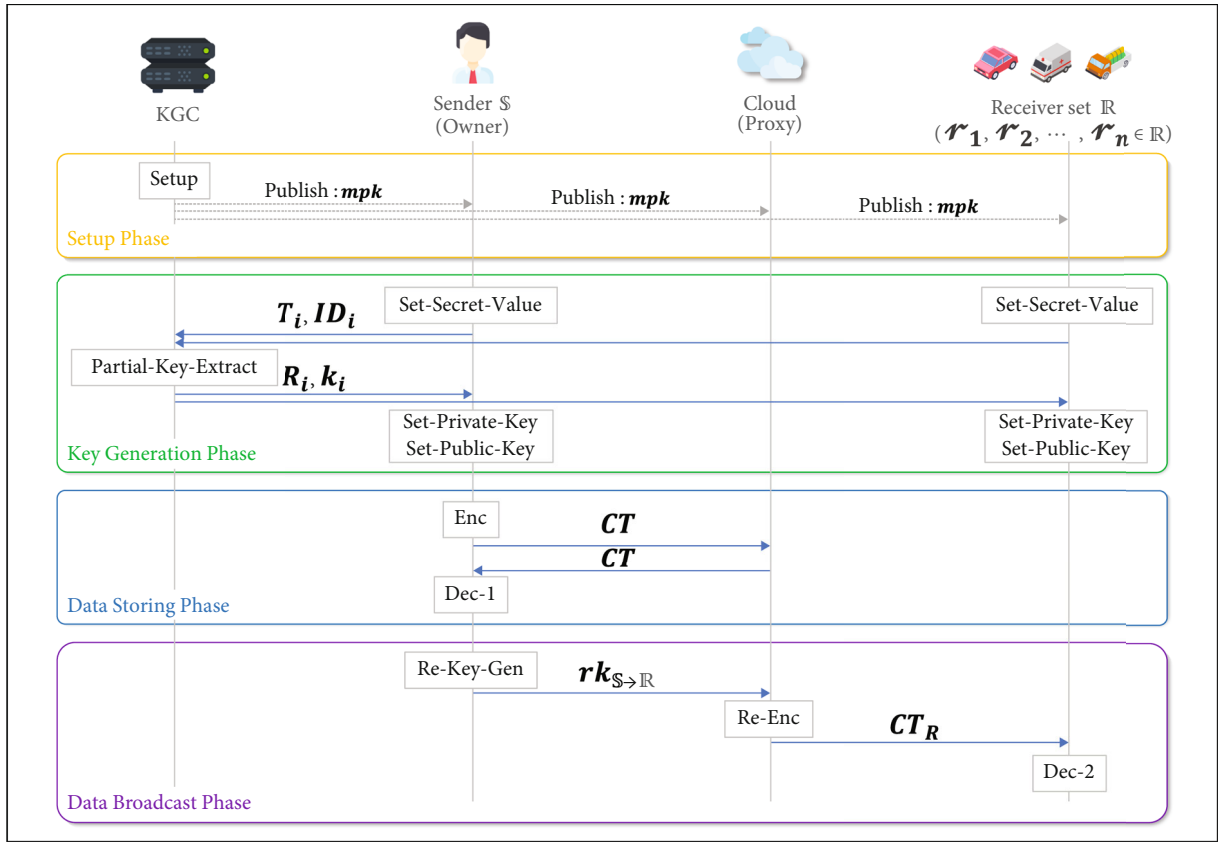


Figure 4: Overview of the proposed scheme.

(xvi) M: message space

(xvii) m: plaintext (message) $(m \in M)$

(xviii) CT: ciphertext

(xix) CT_R: reencrypted ciphertext

(xx) H_1 : one-way hash function $Z_q^* \longrightarrow Z_q^*$

(xxi) H_2 : one-way hash function $\{0,1\}^{l_1+l_2} \longrightarrow Z_q^*$

(xxii) H_3 : one-way hash function $G_q \times \{0,1\}^* \longrightarrow Z_q^*$

(xxiii) H_4 : one-way hash function $Z_q^* \times Z_q^* \longrightarrow \{0,1\}^{l_1+l_2}$

(xxiv) H_5 : one-way hash function $Z_q^* \times \{0,1\}^* \times \{0,1\}^* \longrightarrow Z_q^*$

(xxv) H_6 : one-way hash function $Z_q^* \longrightarrow \{0,1\}^{l_2}$

(xxvi) H_7 : one-way hash function $G_q \times G_q \longrightarrow Z_q^*$

4.3. Construction. The overall structure of this proposed scheme is shown in Figure 4. This scheme is mainly composed of four phases, each of which is composed of the *setup phase*, *key generation phase*, *data storage phase*, and *data broadcast phase*. A detailed description of each phase proceeds in each phase.

4.3.1. Setup Phase. This phase includes the *setup* algorithm. This phase is performed by the KGC in advance so that each user can use the cloud. Here, a master public key that can be commonly used by each user and a master secret key known only to the KGC are generated.

(i) Setup $(\lambda) \longrightarrow$ (msk, mpk): this algorithm is an algorithm performed by the KGC. With the security parameter λ as input, the KGC performs the following process

(1) Choose two λ-bit prime integers p, q and an elliptic curve E defined on F_p. Let G be the additive group on elliptic curve E and G_q be the subgroup of G with prime order q

(2) Select randomly a generator $P \in G_q$

(3) Randomly choose $d \in Z_q^*$ as the msk and calculate $P_{\mathrm{pub}} = d \bullet P$ which is part of mpk

(4) Select five secure one-way hash functions are follows:

$$
\begin{aligned}
&H_1 : Z_q^* \longrightarrow Z_q^*,\\
&H_2 : \{0,1\}^{l_1+l_2} \longrightarrow Z_q^*,\\
&H_3 : G_q \times \{0,1\}^* \longrightarrow Z_q^*,\\
&H_4 : Z_q^* \times Z_q^* \longrightarrow \{0,1\}^{l_1+l_2},\\
&H_5 : Z_q^* \times \{0,1\}^* \times \{0,1\}^* \longrightarrow Z_q^*,\\
&H_6 : Z_q^* \longrightarrow \{0,1\}^{l_2},\\
&H_7 : G_q \times G_q \times \{0,1\}^* \longrightarrow Z_q^*,
\end{aligned}
\tag{1}
$$

where l_1 and l_2 mean the length of the bit string and is determined by the security parameter λ

(5) Publish the system's maser public key $\text{mpk} = \{p, q, l_1, l_2, E, G, G_q, P, P_{\text{pub}}, H_1, H_2, H_3, H_4, H_5, H_6, H_7\}$ and message space $M = \{0,1\}^{l_1}$

4.3.2. Key Generation Phase. This phase includes *set-secret-value*, *partial-key-extract*, *set-private-key*, and *set-public-key* algorithms. In this phase, each user generates their own private key and public key pair so that they can use the cloud. In this phase, each user communicates with the KGC to receive a partial key and uses the partial key to generate their own public key and private key pair as shown in Figure 5.

(i) *Set-secret-value*: this algorithm is an algorithm performed by user i. A user i randomly selects $t_i \in Z_q^*$ and keeps it secure. User i computes $T_i = t_i \bullet P$ as the public key, and user i sends (T_i, ID_i) to the KGC

(ii) *Partial-key-extract*: this algorithm is an algorithm performed by the KGC. According to the identity ID_i of user i, the KGC performs the following steps:

(1) Randomly select $r_i \in Z_q^*$ and compute $R_i = r_i \bullet P$

(2) Calculate a part of the partial private key k_i as follows:

$$
k_i \longleftarrow r_i + dH_3(R_i, T_i, \text{ID}_i) + H_3(dT_i, \text{ID}_i) \,(\text{mod } q) \tag{2}
$$

(3) After that, partial key (R_i, k_i) is delivered to user i through the public channel

(iii) *Set-private-key*: this algorithm is an algorithm performed by user i. After receiving partial key (R_i, k_i) from the KGC, user i verifies these like equations (3) and (4). If verification passes, user i is compute private key $\text{sk}_i = (s_i, t_i)$ as the following steps:

(1) Verify whether the following equation holds:

$$
k_i \bullet P \stackrel{?}{=} R_i + H_7(R_i, T_i, \text{ID}_i) P_{\text{Pub}} + H_3(t_i P_{\text{Pub}}, \text{ID}_i) P \tag{3}
$$

(2) If not, return $\perp$; otherwise, user i computes s_i as follows:

$$
s_i \longleftarrow k_i - H_3(t_i P_{\text{Pub}}, \text{ID}_i) \tag{4}
$$

(3) After that, user i keeps secret $\text{sk}_i = (s_i, t_i)$ as his/her the full private key

(iv) *Set-public-key*: this algorithm is an algorithm performed by user i. User i keeps $\text{pk}_i = (R_i, T_i)$ as the full public key

4.3.3. Data Storing Phase. This phase includes the Enc and Dec-1 algorithms. This phase represents the process of the sender encrypting his/her data with his/her public key and storing it in the cloud. In addition, the sender downloads his/her own data stored in the cloud and a decryption process is also included using the private key to obtain the data source again as shown in Figure 6.

(i) *Enc*: this algorithm is an algorithm performed by the sender $\mathbb{S}$. Sender $\mathbb{S}$ encrypts message m with ciphertext CT by entering his/her public key $\text{pk}_{\mathbb{S}} = (R_{\mathbb{S}}, T_{\mathbb{S}})$ and message $m \in M$. Then, upload the ciphertext CT to the cloud

(1) Compute w, z, and Z using the given message $m \in M$ and $\text{pk}_{\mathbb{S}} = (R_{\mathbb{S}}, T_{\mathbb{S}})$

$$
\begin{aligned}
w &\longleftarrow H_7(R_{\mathbb{S}}, T_{\mathbb{S}}, \text{ID}_{\mathbb{S}}),\\
z &\longleftarrow H_2(m \| w),\\
Z &\longleftarrow zP
\end{aligned}
\tag{5}
$$

(2) Then, sender $\mathbb{S}$ calculates $U_{\mathbb{S}}$ using z and $\text{pk}_{\mathbb{S}} = (R_{\mathbb{S}}, T_{\mathbb{S}})$

$$
U_{\mathbb{S}} \longleftarrow z \bullet \left(R_{\mathbb{S}} + H_7(R_{\mathbb{S}}, T_{\mathbb{S}}, \text{ID}_{\mathbb{S}}) P_{\text{pub}} + T_{\mathbb{S}}\right) \tag{6}
$$

(3) Sender $\mathbb{S}$ calculates α, θ, and C as follows:

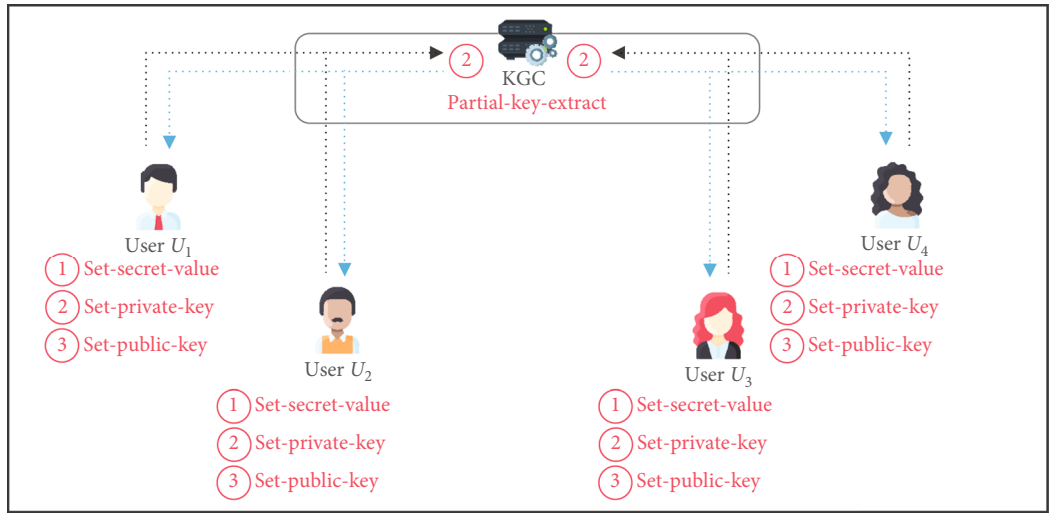


Figure 5: Key generation phase.

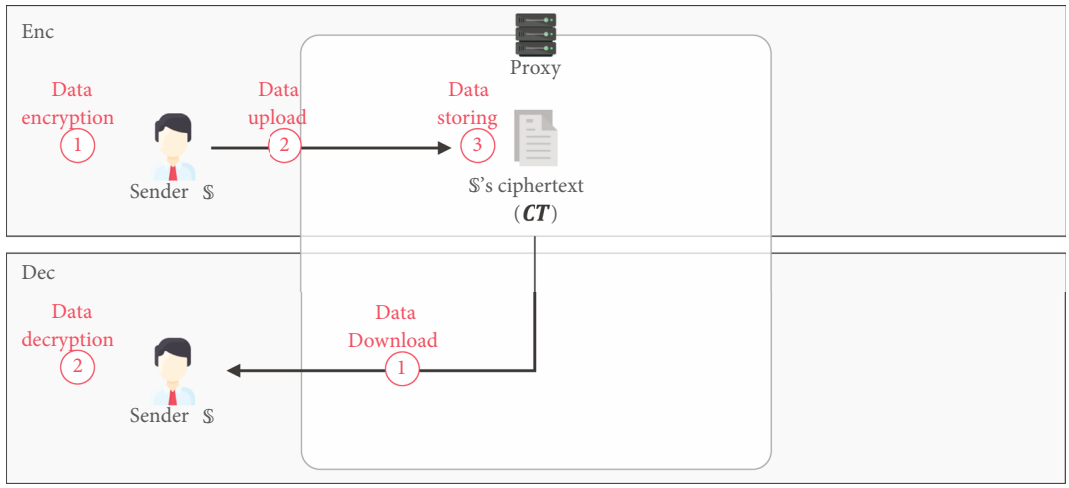


Figure 6: Data storing phase.

$$\begin{aligned} \alpha &\longleftarrow H_1(s_{\mathbb{S}} \cdot t_{\mathbb{S}}), \\ \theta &\longleftarrow H_1(U_{\mathbb{S}} \cdot \alpha), \\ C &\longleftarrow H_4(Z, \theta) \oplus (m \| w) \end{aligned} \tag{7}$$

(4) Generate ciphertext $\mathrm{CT} \longleftarrow (C_1, C_2) = (Z, C)$. Then, the generated CT is uploaded and stored to the cloud

(ii) *Dec-1*: this algorithm is an algorithm performed by the sender $\mathbb{S}$. The sender $\mathbb{S}$ can download the ciphertext $\mathrm{CT} = (C_1, C_2) = (Z, C)$ from cloud. The sender $\mathbb{S}$ who has downloaded the ciphertext CT can obtain the plaintext m by decrypting the ciphertext CT with his/her private key $\mathrm{sk}_{\mathbb{S}} = (s_{\mathbb{S}}, t_{\mathbb{S}})$

(4) Sender $\mathbb{S}$ calculates $U'_{\mathbb{S}}$ using its private key $\mathrm{sk}_{\mathbb{S}} = (s_{\mathbb{S}}, t_{\mathbb{S}})$ and the given ciphertext $\mathrm{CT} = (C_1, C_2, C_3)$

$$U'_{\mathbb{S}} \longleftarrow (s_{\mathbb{S}} + t_{\mathbb{S}}) \bullet C_1 \tag{8}$$

(5) Calculate α and θ' by inputting $\mathrm{sk}_{\mathbb{S}}$ and $U'_{\mathbb{S}}$

$$\alpha \longleftarrow H_1(s_{\mathbb{S}} \cdot t_{\mathbb{S}}), \tag{9}$$

$$\theta' \longleftarrow H_1\left(U'_{\mathbb{S}} \cdot \alpha\right) \tag{10}$$

(6) Calculate m by inputting C_1, C_2, θ'

$$(m\|w) \longleftarrow C_2 \oplus H_4\left(C_1, \theta'\right), \tag{11}$$

$$\begin{aligned} \because C_2 \oplus H_4\left(C_1, \theta'\right) &= H_4(Z, \theta) \oplus (m\|w) \oplus H_4\left(C_1, \theta'\right) \\ &= H_4(Z, \theta) \oplus (m\|w) \oplus H_4\left(Z, \theta'\right) = (m\|w), \end{aligned} \tag{12}$$

where $C_1 = Z$

(7) Verify whether the following equation holds. If not, return $\perp$; otherwise, sender $\mathbb{S}$ keeps plaintext m

$$C_1 \stackrel{?}{=} H_2(m\|H_7(R_{\mathbb{S}}, T_{\mathbb{S}}, ID_{\mathbb{S}}))P, \tag{13}$$

$$\because C_1 = H_2(m\|H_7(R_{\mathbb{S}}, T_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}}))P = H_2(m\|w)P = zP = Z, \tag{14}$$

where $Z = zP$ and $z = H_2(m\|w)$

4.3.4. Data Broadcast Phase. This phase includes *re-key-gen*, *re-enc*, and *dec-2* algorithms. In this phase, the sender generates a reencryption key for a set of recipients and passes it to the proxy. After receiving the reencryption key, the proxy reencrypts the encrypted data and broadcasts it to the recipients. The receiver who has received the broadcast ciphertext can obtain the message by decrypting the ciphertext with their private key as shown in Figure 7.

(i) *Re-key-gen*: this algorithm is executed by sender $\mathbb{S}$ to delegate a ciphertext to set of recipients $\mathbb{R} = (\curlyvee_1, \curlyvee_2, \cdots, \curlyvee_n)$ of selected receiver $\curlyvee_j$ with identity ID_j $(1 \le j \le n)$. The following steps will be performed in this algorithm

(1) Compute U_j, where $j = 1, 2, \cdots, n$.

$$U_j \longleftarrow z \bullet \left(R_j + H_7\left(R_j, T_j, \mathrm{ID}_j\right) P_{\mathrm{pub}} + T_j\right) \tag{15}$$

(2) Compute a polynomial $f(x)$ with degree n using $\beta \in Z_q^*$ as follows:

$$f(x) = \prod_{i=0}^{n} \left(x - U_j\right) + \beta \,(\mathrm{mod}\ q) = x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0, \tag{16}$$

where, $a_i \in Z_p^*$ $(i = 0, 1, \cdots, n-1)$

(3) Compute x using $\mathrm{sk}_{\mathbb{S}}$, α, and β^{-1} as follows:

$$x \longleftarrow (s_{\mathbb{S}} + t_{\mathbb{S}}) \cdot \alpha \cdot \beta^{-1} \tag{17}$$

(4) Sender $\mathbb{S}$ generates reencryption key $\mathrm{rk}_{\mathbb{S} \longrightarrow \mathbb{R}} = (rk_1, rk_2) = (x, \{a_0, a_1, \cdots, a_{n-1}\})$ and sends $\mathrm{rk}_{\mathbb{S} \longrightarrow \mathbb{R}}$ to cloud

(ii) *Re-Enc*: this algorithm is executed by cloud. This algorithm reencrypts ciphertext CT to ciphertext C T_R using reencryption key $rk_{\mathbb{S} \longrightarrow \mathbb{R}}$. The following steps will be performed in this algorithm

(1) Compute CT_R using ciphertext CT and reencryption key $\mathrm{rk}_{\mathbb{S} \longrightarrow \mathbb{R}}$

$$C_1' \longleftarrow C_1, \tag{18}$$

$$C_2' \longleftarrow C_2, \tag{19}$$

$$C_3' \longleftarrow \mathrm{rk}_1 \cdot C_1, \tag{20}$$

$$C_4' \longleftarrow \mathrm{rk}_2 \tag{21}$$

(2) Output $\mathrm{CT}_R = (C_1', C_2', C_3', C_4')$ and send CT_R to receivers $\mathbb{R}$

(iii) *Dec-2*: this algorithm is executed by the selected receiver $\curlyvee_j$ to extract the plaintext from the received ciphertext $\mathrm{CT}_R = (C_1', C_2', C_3', C_4')$. Receiver $\curlyvee_j$ performs following steps:

(1) Compute U_j

$$U_j' \longleftarrow \left(s_j + t_j\right) \bullet C_1' \tag{22}$$

(2) Generate polynomial $f(x)$ and compute β'

$$f(x) = x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0, \tag{23}$$

$$\beta' = f\left(U_j'\right) \tag{24}$$

(3) Compute θ' as input C_3' and β'

$$\theta' = H_1\left(C_3' \cdot \beta'\right), \tag{25}$$

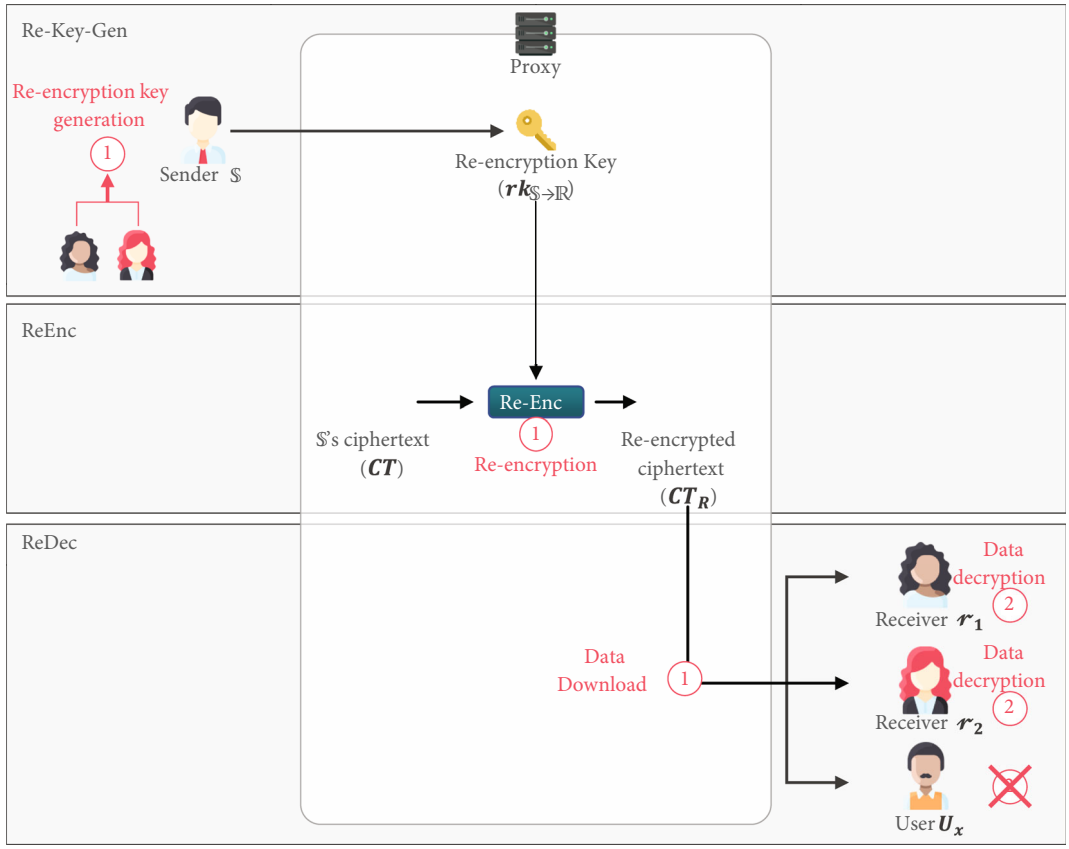


Figure 7: Data broadcast phase.

$$\because C_3' \cdot \beta' = \mathrm{rk}_1 \cdot C_1 \cdot \beta' = x \cdot C_1 \cdot \beta' = (s_{\mathbb{S}} + t_{\mathbb{S}}) \cdot \alpha \cdot \beta^{-1} \cdot C_1 \cdot \beta' = (U_{\mathbb{S}}) \cdot \alpha \tag{26}$$

(4) Compute m as input C_1', C_2', θ'

$$m \longleftarrow C_2' \oplus H_4\left(C_1', \theta'\right), \tag{27}$$

$$\because C_2' \oplus H_4\left(C_1', \theta'\right) = H_4(Z, \theta) \oplus m \oplus H_4\left(C_1', \theta'\right) = H_4(Z, \theta) \oplus m \oplus H_4\left(Z, \theta'\right) = m, \tag{28}$$

where $C_1' = C_1 = Z$

(5) Verify message m. If not, return $\perp$; otherwise, receiver i outputs the plaintext m

$$C_1' \stackrel{?}{=} H_2(m)P, \tag{29}$$

$$\because C_1' = H_2(m)P = zP = Z, \tag{30}$$

where $Z = zP$ and $z = H_2(m)$

4.4. Correctness. In this section, we will prove the correctness of the scheme proposed in Section 4. First, Theorem 1 describes in detail the execution process of the *set-private-key* algorithm, which is a process in which the user verifies whether the partial key received from the KGC is a correct value. Second, Theorem 2 describes in detail the execution process of the *Dec-1* algorithm, which is an algorithm for the sender to decrypt his/her data. Finally, Theorem 2 describes in detail the execution process of the *Dec-2* algorithm, which is an algorithm for the receiver to decrypt the reencrypted data.

Theorem 1. *User i can verify whether the partial key (R_i, k_i) received from the KGC is a value generated from the (T_i, ID_i) created by him/her and the mpk of the correct KGC. This process corresponds to equations (2)–(4).*

Proof. Assuming that one of the users is $\mathcal{U}_1$, $\mathcal{U}_1$ can perform the following process using (R_1, k_1) received from the KGC and its own value (T_1, ID_1) and KGC's master public key mpk.□ □

$\mathcal{U}_1$ can verify whether the received partial key (R_i, k_i) is correct by using the (T_i, ID_i) and mpk. This process corresponds to equation (3).

$$
\begin{aligned}
k_i \bullet P &\stackrel{?}{=} R_1 + H_7(R_1, T_1, \mathrm{ID}_i)P_{\mathrm{Pub}} + H_1(t_1 P_{\mathrm{Pub}}, \mathrm{ID}_1)P, \\
\because k_1 \bullet P &= r_1 \bullet P + H_7(R_1, T_1, \mathrm{ID}_1) \cdot d \cdot P + H_3(t_1 P_{\mathrm{Pub}}, \mathrm{ID}_1)P \\
&= (r_1 + H_7(R_1, T_1, \mathrm{ID}_1) \cdot d + H_3(t_1 \cdot d \cdot P, \mathrm{ID}_1))P \\
&= (r_1 + d \cdot H_7(R_1, T_1, \mathrm{ID}_1) + H_3(T_1 \cdot d_1, \mathrm{ID}_1))P = (k_1)P,
\end{aligned} \tag{31}
$$

where $k_i = r_i + dH_7(R_i, T_i, \mathrm{ID}_i) + H_3(dT_i, \mathrm{ID}_i)$.

Theorem 2. *The sender $\mathbb{S}$ can perform decryption using the ciphertext CT received from the cloud and his/her private key and obtain the plaintext m. This process corresponds to equations (8)–(13).*

Proof. Assuming that one of the senders is $\mathbb{S}$, $\mathbb{S}$ can perform the following process using $\mathrm{CT} = (C_1, C_2, C_3, C_4)$ received from the sender and its own private key $\mathrm{sk}_{\mathbb{S}} = (s_{\mathbb{S}}, t_{\mathbb{S}})$.□ □

$\mathbb{S}$ creates $U'_{\mathbb{S}}$ as follows using his/her private key $\mathrm{sk}_{\mathbb{S}} = (s_{\mathbb{S}}, t_{\mathbb{S}})$. This process corresponds to equations (8) and (9).

$$
\begin{aligned}
U'_{\mathbb{S}} &\longleftarrow (s_{\mathbb{S}} + t_{\mathbb{S}}) \bullet Z, \\
U_{\mathbb{S}} &= z \bullet \left(R_{\mathbb{S}} + H_7(R_{\mathbb{S}}, T_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}})P_{\mathrm{pub}} + T_{\mathbb{S}}\right) \\
&= z \bullet (r_{\mathbb{S}} P + H_7(R_{\mathbb{S}}, T_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}})dP + t_{\mathbb{S}} P) \\
&= z \bullet (r_{\mathbb{S}} + H_7(R_{\mathbb{S}}, T_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}})d + t_{\mathbb{S}})P \\
&= (r_{\mathbb{S}} + dH_7(R_{\mathbb{S}}, T_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}}) + t_{\mathbb{S}})Z \\
&= (r_{\mathbb{S}} + dH_7(R_{\mathbb{S}}, T_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}}) + H_3(dT_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}}) \\
&\quad - H_3(dT_{\mathbb{S}}, \mathrm{ID}_{\mathbb{S}}) + t_{\mathbb{S}})Z = (k_{\mathbb{S}} - H_7(t_{\mathbb{S}} P_{\mathrm{Pub}}, \mathrm{ID}_{\mathbb{S}}) \\
&\quad + t_{\mathbb{S}})Z = (s_{\mathbb{S}} + t_{\mathbb{S}})Z = (s_{\mathbb{S}} + t_{\mathbb{S}})C_1 = U'_{\mathbb{S}}.
\end{aligned} \tag{32}
$$

$\mathbb{S}$ obtains θ' using the generated equations (9) and (10) as follows:

$$
\begin{aligned}
\alpha &\longleftarrow H_1(s_{\mathbb{S}} \cdot t_{\mathbb{S}}), \\
\theta' &\longleftarrow H_1\left(U'_{\mathbb{S}} \cdot \alpha\right).
\end{aligned} \tag{33}
$$

$\mathbb{S}$ can obtain m as follows using C_1, C_2 and the acquired θ' in equation (11).

$$
\begin{aligned}
(m \| w) &\longleftarrow C_2 \oplus H_4\left(C_1, \theta'\right), \\
\because C_2 \oplus H_4\left(C_1, \theta'\right) &= C \oplus H_4\left(Z, \theta'\right) \\
&= H_4(Z, \theta) \oplus (m \| w) \oplus H_4\left(Z, \theta'\right) \\
&= (m \| w),
\end{aligned} \tag{34}
$$

where $\mathrm{CT} = (C_1, C_2) = (Z, C)$ and $C = H_4(Z, \theta) \oplus (m \| w)$.

Theorem 3. *The receivers $\mathbb{R}$ can perform decryption using the reencrypted ciphertext CT_R received from the cloud and his/her private key and obtain the plaintext m. This process corresponds to equations (18)–(29).*

Proof. Assuming that one of the receivers is $\mathscr{r}_1$, $\mathscr{r}_1$ can perform the following process using $\mathrm{CT}_R = (C'_1, C'_2, C'_3, C'_4)$ received from the sender and its own private key $\mathrm{sk}_{\mathscr{r}_1} = (s_{\mathscr{r}_1}, t_{\mathscr{r}_1})$.□ □

$\mathscr{r}_1$ creates $U_{\mathscr{r}_1}$ as follows using his/her private key $\mathrm{sk}_{\mathscr{R}} = (s_{\mathscr{R}}, t_{\mathscr{R}})$. This process corresponds to equations (18) and (22).

$$
\begin{aligned}
U'_{\mathscr{r}_1} &\longleftarrow \left(s_{\mathscr{r}_1} + t_{\mathscr{r}_1}\right) \bullet Z, \\
U_{\mathscr{r}_1} &= z \bullet \left(R_{\mathscr{r}_1} + H_7\left(R_{\mathscr{r}_1}, T_{\mathscr{r}_1}, \mathrm{ID}_{\mathscr{r}_1}\right)P_{\mathrm{pub}} + T_{\mathscr{r}_1}\right) \\
&= z \bullet \left(r_{\mathscr{r}_1} P + H_7\left(R_{\mathscr{r}_1}, T_{\mathscr{r}_1}, \mathrm{ID}_{\mathscr{r}_1}\right)dP + t_{\mathscr{r}_1} P\right) \\
&= z \bullet \left(r_{\mathscr{r}_1} + H_7\left(R_{\mathscr{r}_1}, T_{\mathscr{r}_1}, \mathrm{ID}_{\mathscr{r}_1}\right)d + t_{\mathscr{r}_1}\right)P \\
&= \left(r_{\mathscr{r}_1} + dH_7\left(R_{\mathscr{r}_1}, T_{\mathscr{r}_1}, \mathrm{ID}_{\mathscr{r}_1}\right) + t_{\mathscr{r}_1}\right)Z \\
&= \left(r_{\mathscr{r}_1} + dH_7\left(R_{\mathscr{r}_1}, T_{\mathscr{r}_1}, I\mathscr{r}_1\right)\right. \\
&\quad \left. + H_3\left(dT_{\mathscr{r}_1}, \mathrm{ID}_{\mathscr{r}_1}\right) - H_3\left(dT_{\mathscr{r}_1}, \mathrm{ID}_{\mathscr{r}_1}\right) + t_{\mathscr{r}_1}\right)Z \\
&= \left(k_{\mathscr{r}_1} - H_7\left(t_{\mathscr{r}_1} P_{\mathrm{Pub}}, \mathrm{ID}_{\mathscr{r}_1}\right) + t_{\mathscr{r}_1}\right)Z = \left(s_{\mathscr{r}_1} + t_{\mathscr{r}_1}\right)Z \\
&= \left(s_{\mathscr{r}_1} + t_{\mathscr{r}_1}\right)C_1 = U'_{\mathscr{r}_1}.
\end{aligned} \tag{35}
$$

$\mathscr{r}_1$ obtains θ' using the generated equations (23)–(25) as follows:

$$
\begin{aligned}
\beta' \longleftarrow f\left(U_{\mathscr{r}_1}\right) &= \prod_{i=0}^{n}\left(U_{\mathscr{r}_1} - U_i\right) + \beta \,(\mathrm{mod}\ q) \\
&= \left(U_{\mathscr{r}_1} - U_{\mathscr{r}_1}\right) \bullet \left(U_{\mathscr{r}_1} - U_{\mathscr{r}_2}\right) \cdots \left(U_{\mathscr{r}_1} - U_{\mathscr{r}_i}\right) \\
&\quad + \beta \,(\mathrm{mod}\ q) = 0 \bullet \left(U_{\mathscr{r}_1} - U_{\mathscr{r}_2}\right) \cdots \left(U_{\mathscr{r}_1} - U_{\mathscr{r}_i}\right) \\
&\quad + \beta \,(\mathrm{mod}\ q).
\end{aligned} \tag{36}
$$

$\mathscr{r}_1$ can obtain m using C'_1, C'_2 and the acquired θ' in equation (27).

$$
\begin{aligned}
(m \| w) &\longleftarrow C'_2 \oplus H_4\left(C'_1, \theta'\right), \\
\because C'_2 \oplus H_4\left(C'_1, \theta'\right) &= C \oplus H_4\left(Z, \theta'\right) \\
&= H_4(Z, \theta) \oplus (m \| w) \oplus H_4\left(Z, \theta'\right) \\
&= (m \| w),
\end{aligned} \tag{37}
$$

where $\mathrm{CT}_R = (C'_1, C'_2, C'_3, C'_4) = (Z, C, x \cdot Z, \mathrm{rk}_2)$ and $C = H_4(Z, \theta) \oplus (m \| w)$.

Table 1: Comparison of the security requirements.

	Bilinear pairing	Key escrow problem	Receiver anonymity	Re-key-generation
Wang and Yang [41]	Used	Insecure	Offer	KGC/BC
Maiti and Misra [37]	Used	Insecure	Offer	Sender
Sun et al. [38]	Used	Insecure	Offer	Sender
Yin et al. [39]	Used	Insecure	Offer	Sender
Chunpeng et al. [40]	Used	Insecure	Offer	Sender
Proposed scheme	Not used	Secure	Offer	Sender

5. Analysis of the Proposed CL-BPRE Scheme

In this section, we analyze the efficiency of the proposed scheme and explain how to achieve data security. In addition, the advantages of the proposed scheme are explained through a comparison with previous studies.

5.1. Analysis of Security Requirements. We analyze whether the proposed scheme is successful in achieving the security requirements presented in Section 3.2. There are a total of 7 security requirements, each of which is *confidentiality*, *integrity*, *key escrow problem*, *partial key verifiability*, *receiver anonymity*, and *decryption fairness* as shown in Table 1.

(i) *Confidentiality*: in the proposed scheme, an ECC-based encryption operation is performed to provide data confidentiality. In this process, the message itself is not encrypted with the public key of each recipient but a session key is created to encrypt the message. Therefore, to decrypt a message, a session key must be obtained, and to obtain a session key, only a legitimate recipient must carry out the computation

$$f(x) = \prod_{i=0}^{n}(x-\mu_i) + \beta \,(\mathrm{mod}\ q) = x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0. \tag{38}$$

Here, μ_i is $\mu_i \longleftarrow H_3(U_i, \mathrm{ID}_i, w)$ and U_i is $U_i \longleftarrow z\bullet(R_i + H_7(R_i, T_i, \mathrm{ID}_i)P_{\mathrm{Pub}} + T_i)$. Therefore, each recipient must have its own private key to generate μ_i and the user who generates μ_i can obtain the θ through the following process:

$$\begin{aligned}\beta' \longleftarrow f(\mu_i) &= \prod_{i=0}^{n}(\mu_i - \mu_i) + \beta \,(\mathrm{mod}\ q)\\ &= (\mu_1 - \mu_1)\bullet(\mu_1 - \mu_2)\cdots(\mu_1 - \mu_n) + \beta \,(\mathrm{mod}\ q)\\ &= 0\bullet(\mu_1 - \mu_2)\cdots(\mu_1 - \mu_n) + \beta \,(\mathrm{mod}\ q).\end{aligned} \tag{39}$$

μ_i can obtain the θ through the following process:

$$m \longleftarrow C \oplus H_4(Z, \theta). \tag{40}$$

If an attacker attempts to create U_i' with only the public key of the recipient i, U_i' cannot be generated normally, as follows:

$$U_i' \longleftarrow \overset{?}{z}\bullet(R_i + H_7(R_i, T_i, \mathrm{ID}_i)P_{\mathrm{Pub}} + T_i). \tag{41}$$

According to the above formula, because the attacker does not know z, it is impossible to forge U_i'.

(ii) *Integrity*: recipients who have decrypted the data can verify the integrity of the data using the values contained in the integrity ciphertext and the parameters of the public KGC, as in Theorem 3 of Section 4.4. The proof of this is as follows:

$$\begin{aligned}C_1' &\overset{?}{=} H_2(m\|w)P,\\ \because C_1' &= H_2(m\|w)P = zP = Z,\end{aligned} \tag{42}$$

where $Z = zP$ and $z = H_2(m\|w)$.

The receiver that decrypts ciphertext CT_R can obtain message m and verification value w. Here, $H_2(m\|w)$ is equal to z, and thus, the integrity of the message can be verified by comparing whether $H_2(m)P$ is equal to $C_1 = Z$.

(iii) *Key escrow problem*: the proposed scheme uses a certificateless PKC method that has been demonstrated to be successful in solving the key escrow problem. To solve the key escrow problem, the KGC must not know the user's complete private key. In the existing IBC, in the private-key-gen algorithm, the KGC generates a user's complete private key and delivers it to each user. The key escrow problem is by dividing the key generation process into four algorithms: *set-secret value*, *partial-key-extract*, *set-private-key*, and *set-public-key*

First, the *set-secret-value* algorithm generates T_i using secret value t_i, randomly selected by the user, and the master public key value P of the KGC. At this time, t_i is safely stored only by the user and T_i and ID_i are delivered to the KGC through an open channel.

In the *partial-key-extract* algorithm, using the T_i and ID_i received by the KGC from the user, partial keys R_i and k_i are generated through the following process and delivered to the user.

$$\begin{aligned}R_i &= r_i\bullet P,\\ k_i &\longleftarrow r_i + dH_7(R_i, T_i, \mathrm{ID}_i) + H_3(dT_i, \mathrm{ID}_i)(\mathrm{mod}\ P).\end{aligned} \tag{43}$$

In the *set-private-key* algorithm, the private key is calculated using the R_i and k_i received by the user from the KGC.

At this time, the user does not use (R_i, k_i) as the private key, but uses t_i, which only the user knows, and s_i, which is generated based on the k_i received from the KGC, as the private key.

$$s_i \longleftarrow k_i - H_3(t_i P_{\text{Pub}}, \text{ID}_i). \tag{44}$$

Finally, in the *set-public-key* algorithm, T_i generated by the user and R_i generated by the KGC are used as public keys.

As a result, the KGC must know t_i to obtain the user's private key $\text{sk}_i = (s_i, t_i)$. However, because t_i is a secret value stored safely by the user, the key escrow problem by the KGC does not occur.

The KGC only knows $\text{pk}_i = (T_i, R_i)$ and k_i, and the KGC cannot know $\text{sk}_i = (s_i, t_i)$.

(iv) *Partial key verifiability*: the proposed scheme is designed to satisfy several security requirements. In this process, there was an increase in the amount of computation

$$\begin{aligned} k_i \bullet P &\stackrel{?}{=} R_i + H_7(R_i, T_i, \text{ID}_i) P_{\text{Pub}} + H_3(t_i P_{\text{Pub}}, \text{ID}_i) P, \\ \because k_i \bullet P &= r_i \bullet P + H_7(R_i, T_i, \text{ID}_i) \cdot d \cdot P + H_3(t_i P_{\text{Pub}}, \text{ID}_i) P \\ &= (r_i + H_7(R_i, T_i, \text{ID}_i) \cdot d + H_3(t_i \cdot d \cdot P, \text{ID}_i)) P \\ &= (r_i + d \cdot H_7(R_i, T_i, \text{ID}_i) + H_3(T_i \cdot d_i, \text{ID}_i)) P = (k_i) P, \end{aligned} \tag{45}$$

where, $k_i = r_i + dH_7(R_i, T_i, \text{ID}_i) + H_3(dT_i, \text{ID}_i)$,

$$\begin{aligned} R_i &= r_i P, \\ T_i &= t_i P, \\ P_{\text{pub}} &= dP. \end{aligned} \tag{46}$$

Through the above calculation, user i can know that the partial key that it has received is based on secret value r_i generated by user i and that it is generated by a legitimate KGC

(v) *Receiver anonymity*: in the proposed scheme, a Lagrange interpolation polynomial is applied to provide the recipient's anonymity. In this method, the information of the user included in the polynomial cannot be obtained because the recipient is only confirmed by a polynomial. The formula for this polynomial is as follows:

$$\begin{aligned} f(x) &= \prod_{i=0}^{n} (x - \mu_i) + \beta \,(\text{mod } q) \\ &= (x - \mu_1) \bullet (x - \mu_2) \bullet \cdots \bullet (x - \mu_n) + \beta \,(\text{mod } q) \\ &= x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0, \end{aligned}$$

$$U_j' = (s_j + t_j) \bullet C_1' \tag{47}$$

To identify a specific recipient in the above polynomial, it is possible to generate μ_i' of the specific receiver. However, as in the confidentiality item above, an attacker cannot forge U_i'.

$$U_i' \longleftarrow \stackrel{?}{z} \bullet (R_i + H_7(R_i, T_i, \text{ID}_i) P_{\text{Pub}} + T_i). \tag{48}$$

As a result, the attacker cannot identify the recipient.

(vi) *Decryption fairness*: in the decoding process of the recipient data included in the recipient list, the decoding should not be disadvantageous because of the intervention of a third party or the KGC. To this end, in the proposed scheme, it is not possible to change the list of recipients by configuring a polynomial or adding an amount of computation by designating only specific recipients. This takes advantage of the property of the following polynomial, and in order for an attacker to make a specific recipient disadvantageous, he/she must be able to completely forge $f(x)$, a polynomial that targets all receivers.

$$\begin{aligned} f(x) &= \prod_{i=0}^{n} (x - \mu_i) + \beta \,(\text{mod } q) \\ &= (x - \mu_1) \bullet (x - \mu_2) \bullet \cdots \bullet (x - \mu_n) \\ &\quad + \beta (\text{mod } q) = x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0. \end{aligned} \tag{49}$$

5.2. Comparison of Schemes

(i) *Security requirements*: the proposed scheme of this study was designed to satisfy various requirements that the existing schemes do not provide. Wang and Yang [41] and Maiti and Misra [37], and Sun et al. [38] proposed IBC-based BPRE. Since these two schemes operate in the IBC method, the KGC generates and issues the user's private key. Since these two schemes operate in the IBC method, the KGC generates and issues the user's private key. Sur et al. generated the user's private key through the $\text{keygen}_{\text{IBE}}$ algorithm as follows:

$$\text{sk}_{\text{ID}} = (d_0, d_1, d_2) = \left(g^{\alpha_2} \left(g_1^{\text{ID}} g \right)^u, g^u, g^{u/\alpha} \right), \tag{50}$$

where master secret key $\text{mk} = g_2^{\alpha}$ and random value $\alpha, u, x \in Z_p^{*}$.

According to the above formula, each user's private key can be generated only by the KGC that owns the master secret key and a complete private key is generated, which may cause a key escrow problem. Maiti et al. also generated the user's private key in the KG algorithm. In this process, the KGC generates a complete private key through the

Table 2: Comparison of the computation efficiency.

	Enc	Re-key-gen	Re-enc	Dec-2
Wang and Yang [41]	$(2)T_M + (4)T_e + (1)T_P$	$(10+3n)T_M + (1)T_e$	$(6)T_e$	$(7)T_M + (7)T_e + (5)T_P$
Maiti and Misra [37]	$(4)T_M + (3)T_e$	$(3+n^2+n)T_M + (3+n)T_e$	$(1)T_M + (1)T_P$	$(1)T_M + (2)T_P$
Sun et al. [38]	$(2+\mathrm{n})T_M + (5)T_e + (1)T_P$	$(3+n)T_M + (6)T_e + (1)T_p$	$(1+\mathrm{n})T_M + (2)T_P$	$(4+\mathrm{n})T_M + (2)T_e$
Yin et al. [39]	$(4+2n)T_M + (4)T_e$	$(5+n)T_M + (6)T_e$	$(4+3n)T_M + (2)T_e + (2)T_p$	$(7)T_M + (1)T_e + (3)T_P$
Chunpeng et al. [40]	$(2)T_M + (3)T_e$	$(5+n)T_M + (5+\mathrm{n})T_e + (1)T_p$	$(1)T_M + (2)T_\mathrm{p}$	$(6)T_M + (2)T_e + (2)T_p$
Proposed scheme	$(2)T_{\mathrm{EM}} + (2)T_{\mathrm{EA}}$	$(1+n)T_M + (2n)T_{\mathrm{EM}} + (2n)T_{\mathrm{EA}}$	$(1)T_{\mathrm{EM}}$	$(2)T_{\mathrm{EM}}$

T_M: computation time of modular multiplication operation; T_{EM}: computation time of ECC multiplication operation; T_{EA}: computation time of ECC point add operation; T_e: computation time of exponent operation; T_p: computation time of bilinear pairing operation.

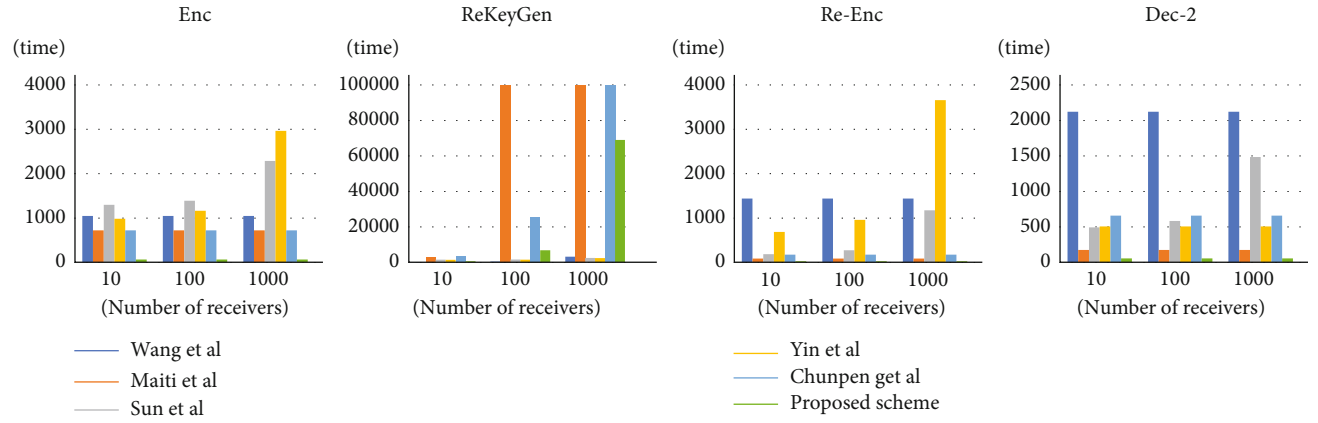


Figure 8: Computation time of schemes.

following operation, which may cause a key escrow problem.

$$\mathrm{sk}_{\mathrm{ID}_A} = g^{1/(\alpha+H_1(\mathrm{ID}_A))}, \tag{51}$$

where the system secret key is α.

Sun et al. also generated the user's private key in the key-gen algorithm as follows.

$$\mathrm{sk}_i = g_i^{\gamma}, \tag{52}$$

where system secret key msk = γ.

This also causes Sun et al. key escrow problems. In addition, both Wang et al. and Maiti et al., Sun et al., Yin et al., and Chunpeng et al. use a pairing operation, which takes a lot of computation time. Yin et al. and Chunpeng et al. all proposed certificateless BPRE. However, since all three methods use a pairing operation, a lot of computation time is required. Additionally, Yin et al. pose a threat of privacy invasion because the anonymity of the recipient is not guaranteed.

$$\mathrm{rk} = \left(\mathrm{rk}_1, \mathrm{rk}_2, \mathrm{rk}_3, \mathrm{rk}_4, \mathrm{rk}_5, (\mathrm{rk}_{6,i})_{i\in\{1,2,\cdots,k+1\}}\right), \tag{53}$$

where $\mathrm{rk}_{6,i} = \mu_i^s$, for

$$i \in \{1, 2, \cdots, k+1\}. \tag{54}$$

Therefore, the proposed scheme uses the CL method and does not use the pairing operation, so that the BPRE can be performed in less time. In addition, it is possible to use BPRE more safely and efficiently by solving the problem of key escrow and recipient anonymity.

(ii) *Computational efficiency*: the proposed scheme of this study was designed with a lower number of calculations compared to the existing schemes as shown in Table 2. Since the pairing operation is not basically used, BPRE can be performed with less computation time compared to the existing methods. In addition, it has a lower number of operations by simplifying encryption and decryption operations. However, the number of operations for generating the reencryption key has increased. Therefore, some computational efficiency may be lowered in an environment where the list of recipients is constantly changing. However, since the amount of data encryption and decryption operations is low, the burden on the user terminal can be reduced. The computation time chart is shown in Figure 8.

6. Conclusion

Data sharing using the cloud is related to data confidentiality and key management issues. First, the cloud is a semitrusted

environment and data can be exposed at any time by an insider or external attack. Therefore, in order to solve this problem, the application of encryption is essential. However, in order to share encrypted data with other users, it is essential to distribute a key that can decrypt the data. However, in a data storage and sharing environment using the cloud, it is very difficult to distribute the key because the key cannot be delivered face to face. To solve this problem, proxy reencryption has been proposed that allows data being stored encrypted in the cloud to be shared with other users. Proxy reencryption is a technology that reencrypts data that has been encrypted once to data that other users can decrypt without having to decrypt the data and share the private key. However, since the existing proxy reencryption can reencrypt by specifying only one recipient at a time, if the number of recipients increases, the number of reencryption also increases and the number of times to generate a reencryption key for reencryption also increases. Therefore, broadcast proxy reencryption has been proposed to solve this problem. Broadcast proxy reencryption is a combination of broadcast encryption technology and proxy reencryption technology. The broadcast encryption method is effective when distributing the same data to multiple recipients at the same time because multiple recipients can be specified with only one encryption. By combining these features with proxy reencryption, broadcast encryption can be used when data encrypted once is shared with multiple users at the same time. Therefore, it can be applied to various environments such as update servers that distribute data to many recipients at the same time, secure email, and IoT. However, the receiver anonymity, key escrow problem, decryption fairness, partial key verification problem, etc. that appear in the broadcast encryption method also appear in the broadcast proxy reencryption. Therefore, you can safely use broadcast proxy reencryption only after solving these problems. To this end, in this study, in the process of designating a plurality of receivers, the receiver cannot be identified using a polynomial and the receiver is additionally modulated by modulating the polynomial to change the receiver or a specific receiver is designed so that it does not have a disadvantage in decoding. In addition, by not using the pairing operation in this process, the calculation time is reduced and the amount of calculation is simplified, so that data can be broadcast even with a lower operation. In the key generation process, the key escrow problem caused by KGC was solved by using the certificateless method instead of the existing IBC type. As a result, the proposed scheme solves the security threats of the existing schemes and at the same time reduces the amount of computation and the computation time, so that it is possible to provide a more secure and efficient broadcast proxy reencryption.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare no conflict of interest.

Authors' Contributions

Won-Bin Kim and Su-Hyun Kim contributed equally to this work.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2019R1A2C1085718) and the Soonchunhyang University Research Fund and the BK21 FOUR (Fostering Outstanding Universities for Research) (No. :5199990914048).

References

[1] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 38, no. 5, pp. 968–979, 2020.

[2] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering (TNSE)*, vol. 7, no. 2, pp. 766–775, 2020.

[3] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *The 39th IEEE International Conference on Distributed Computing Systems (ICDCS 2019)*, Dallas, TX, USA, 2019.

[4] S. Ji, J. He, A. S. Uluagac, R. Beyah, and Y. Li, "Cell-based snapshot and continuous data collection in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 9, no. 4, pp. 1–29, 2013.

[5] S. Ji and Z. Cai, "Distributed data collection in large-scale asynchronous wireless sensor networks under the generalized physical interference model," *IEEE/ACM Transactions on Networking*, vol. 21, no. 4, pp. 1270–1283, 2013.

[6] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 127–144, Konstanz, Germany, 1998.

[7] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," *ACM Transactions on Information and System Security*, vol. 9, no. 1, pp. 1–30, 2006.

[8] D. Robert, H. J. Weng, S. Liu, and K. Chen, "Chosen-ciphertext secure proxy re-encryption without pairings," in *International Conference on Cryptology and Network Security*, pp. 1–17, Hong Kong, China, 2008.

[9] B. Libert and D. Vergnaud, "Unidirectional chosen-ciphertext secure proxy re-encryption," in *International Workshop on Public Key Cryptography*, pp. 360–379, Barcelona, Spain, 2008.

[10] K. Varad and C. Pandu Rangan, "RSA-TBOS signcryption with proxy re-encryption," in *Proceedings of the 8th ACM workshop on Digital rights management*, Alexandria Virginia USA, 2008.

[11] S. Jun and Z. Cao, "CCA-secure proxy re-encryption without pairings," in *International Workshop on Public Key Cryptography*, pp. 357–376, Irvine, CA, USA, 2009.

[12] G. Ateniese, K. Benson, and S. Hohenberger, "Key-private proxy re-encryption," in *Cryptographers' Track at the RSA Conference*, pp. 279–294, San Francisco, CA, USA, 2009.

[13] J. Shao, P. Liu, G. Wei, and Y. Ling, "Anonymous proxy re-encryption," *Security and Communication Networks*, vol. 5, no. 5, 449 pages, 2012.

[14] M. Green and G. Ateniese, "Identity-based proxy re-encryption," in *International Conference on Applied Cryptography and Network Security*, Zhuhai, China, 2007.

[15] H. Wang and Z. Cao, "More efficient CCA-secure unidirectional proxy re-encryption schemes without random oracles," *Security and Communication Networks*, vol. 6, no. 2, 181 pages, 2013.

[16] S. S. M. Chow, J. Weng, Y. Yang, and R. H. Deng, "Efficient unidirectional proxy re-encryption," in *International Conference on Cryptology in Africa*, pp. 316–332, Stellenbosch, South Africa, 2010.

[17] G. Hanaoka, Y. Kawai, N. Kunihiro, T. Matsuda, and J. Weng, "Generic construction of chosen ciphertext secure proxy re-encryption," in *Cryptographers' Track at the RSA Conference*, pp. 349–364, San Francisco, CA, USA, 2012.

[18] R. Canetti and S. Hohenberger, "Chosen-ciphertext secure proxy re-encryption," in *CCS '07: Proceedings of the 14th ACM conference on Computer and communications security*, pp. 185–194, Alexandria Virginia USA, 2007.

[19] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Annual International Cryptology Conference*, pp. 213–229, Santa Barbara, CA, USA, 2001.

[20] T. Matsuo, "Proxy re-encryption systems for identity-based encryption," in *International Conference on Pairing-Based Cryptography*, pp. 247–267, Tokyo, Japan, 2007.

[21] C. K. Chu and W. G. Tzeng, "Identity-based proxy re-encryption without random oracles," in *International Conference on Information Security*, pp. 189–202, Valparaíso, Chile, 2007.

[22] C. Sur, C. Jung, Y. Park, and K. Rhee, "Chosen-ciphertext secure certificateless proxy re-encryption," in *IFIP International Conference on Communications and Multimedia Security*, pp. 214–232, Linz, Austria, 2010.

[23] L. Xu, X. Wu, and X. Zhang, "CL-PRE: a certificateless proxy re-encryption scheme for secure data sharing with public cloud," in *Proceedings of the 7th ACM symposium on information, computer and communications security*, pp. 87-88, Seoul Korea, 2012.

[24] K. Yang, J. Xu, and Z. Zhang, "Certificateless proxy re-encryption without pairings," in *International Conference on Information Security and Cryptology*, pp. 67–88, Seoul, Korea (Republic of), 2013.

[25] A. Srinivasan and C. P. Rangan, "Certificateless proxy re-encryption without pairing: revisited," in *Proceedings of the 3rd International Workshop on Security in Cloud Computing*, pp. 41–52, Singapore Republic of Singapore, 2015.

[26] S. Berkovits, "How to broadcast a secret," in *Workshop on the Theory and Application of of Cryptographic Techniques*, pp. 535–541, Brighton, United Kingdom, 1991.

[27] D. Naor, M. Naor, and J. Lotspiech, "Revocation and tracing schemes for stateless receivers," in *Annual International Cryptology Conference*, pp. 41–62, Santa Barbara, CA, USA, 2001.

[28] D. Halevy and A. Shamir, "The LSD broadcast encryption scheme," in *Annual International Cryptology Conference*, pp. 47–60, Santa Barbara, CA, USA, 2002.

[29] Y. Dodis and N. Fazio, "Public key broadcast encryption for stateless receivers," in *ACM Workshop on Digital Rights Management*, pp. 61–80, Washington, DC, USA, 2002.

[30] C. Delerablée, P. Paillier, and D. Pointcheval, "Fully collusion secure dynamic broadcast encryption with constant-size ciphertexts or decryption keys," in *International Conference on Pairing-Based Cryptography*, pp. 39–59, Tokyo, Japan, 2007.

[31] C. Delerablée, "Identity-based broadcast encryption with constant size ciphertexts and private keys," in *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 200–215, Kuching, Malaysia, 2007.

[32] C. Gentry and B. Waters, "Adaptive security in broadcast encryption systems (with short ciphertexts)," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 171–188, Cologne, Germany, 2009.

[33] J. Hur, C. Park, and S. O. Hwang, "Privacy-preserving identity-based broadcast encryption," *Information Fusion*, vol. 13, no. 4, pp. 296–303, 2012.

[34] Z. Zhou, D. Huang, and Z. Wang, "Efficient privacy-preserving ciphertext-policy attribute based-encryption and broadcast encryption," *IEEE Transactions on Computers*, vol. 64, no. 1, pp. 126–138, 2015.

[35] W. Kim, D. Seo, D. Kim, and I. Lee, "Data distribution for multiple receivers in a connected car environment using 5G communication," *Security and Communication Networks*, vol. 2021, Article ID 5599996, 14 pages, 2021.

[36] C. K. Chu, J. Weng, S. S. M. Chow, J. Zhou, and R. H. Deng, "Conditional proxy broadcast re-encryption," *Lecture Notes in Computer Science*, vol. 5594, pp. 327–342, 2009.

[37] S. Maiti and S. Misra, "P2B: privacy preserving identity-based broadcast proxy re-encryption," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5610–5617, 2020.

[38] M. Sun, C. Ge, L. Fang, and J. Wang, "A proxy broadcast re-encryption for cloud data sharing," *Multimedia Tools and Applications*, vol. 77, no. 9, pp. 10455–10469, 2018.

[39] S. Yin, H. Li, and L. Teng, "A novel proxy re-encryption scheme based on identity property and stateless broadcast encryption under cloud environment," *International Journal of Network Security*, vol. 21, no. 5, pp. 797–803, 2019.

[40] G. Chunpeng, Z. Liu, J. Xia, and F. Liming, "Revocable identity-based broadcast proxy re-encryption for data sharing in clouds," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, 2019.

[41] X. An Wang and X. Yang, "Identity based broadcast encryption based on one to many identity based proxy re-encryption," in *2009 2nd IEEE International Conference on Computer Science and Information Technology*, Beijing, China, 2009.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 7225552, 12 pages
https://doi.org/10.1155/2021/7225552

Research Article

Service Partition Method Based on Particle Swarm Fuzzy Clustering

Hong Xia,[1,2,3] **Qingyi Dong**,[1] **Hui Gao**,[1] **Yanping Chen**,[1,2,3] **and ZhongMin Wang**[1,2,3]

[1]*School of Computer Science and Technology, Xi'an University of Posts and Telecommunications, Xi'an, 710121 Shaanxi, China*
[2]*Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processing, Xi'an, 710121 Shaanxi, China*
[3]*Xi'an Key Laboratory of Big Data and Intelligent Computing, Xi'an, 710121 Shaanxi, China*

Correspondence should be addressed to Yanping Chen; chenyp@xupt.edu.cn

Received 3 September 2021; Revised 27 October 2021; Accepted 17 November 2021; Published 8 December 2021

Academic Editor: Yueshen Xu

It is difficult to accurately classify a service into specific service clusters for the multirelationships between services. To solve this problem, this paper proposes a service partition method based on particle swarm fuzzy clustering, which can effectively consider multirelationships between services by using a fuzzy clustering algorithm. Firstly, the algorithm for automatically determining the number of clusters is to determine the number of service clusters based on the density of the service core point. Secondly, the fuzzy *c*-means combined with particle swarm optimization algorithm to find the optimal cluster center of the service. Finally, the fuzzy clustering algorithm uses the improved Gram-cosine similarity to obtain the final results. Extensive experiments on real web service data show that our method is better than mainstream clustering algorithms in accuracy.

1. Introduction

With the development of service-oriented architecture technology, web service has become a vital software resource on the Internet. The number, scale, and types of services have grown rapidly, and services with similar functions have also increased. In the current situation, the difficulty of managing services and assisting in service discovery is time-consuming. Therefore, how to manage web services more conveniently and quickly and accurately find the service that meets the needs of users in a large number of services is a big challenge [1].

The service clustering method can effectively help manage services and assist in service discovery. Web service clustering method has become a key method for service discovery, service recommendation, and service management, which can help web service search engines search services and reduce their search space [2]. Service clustering is aimed at dividing multiple services into different clusters based on similarity. In the same cluster, each service is as similar as possible, while in different clusters, each service is as different as possible. Service clustering can better classify services, compress search space, shorten search time, help quickly manage services, and provide users with accurate and efficient services.

Most related research shows that the service clustering method is based on a topic model, which can improve the efficiency of search services. Many scholars have studied the service clustering method based on the topic model. Paper [3] first applies BTM to learn the latent topics of web service description corpus and then uses the *k*-means algorithm to cluster web services. Considering the web service's description on text is short and lacks enough adequate information. Paper [4] proposes a web service clustering method based on Word2vec and Latent Dirichlet Allocation (LDA) topic model. Word2vec expands the content of web service description documents. Then, use the topic model to model the extended description document.

Most topic models cause web service clustering with low accuracy because most topic models cannot build a well model with short text. Paper [5] proposes a web service clustering with multifunctionality based on LDA and fuzzy *C* -means algorithm. LDA topic model is used to model description documents of web service, and fuzzy *c*-means

algorithm clusters web services into different functional classes. Paper [6] proposes a semantic web service discovery based on fuzzy clustering optimization. As the preprocessing part, the improved fuzzy *c*-means clustering algorithm cluster services into a different class. In this preprocessing process, the improved fuzzy clustering algorithm considers four functional parameters of service input, output, premise, and effect of service as the clustering parameters.

In summary, the existing web service cluster method only focuses on the cluster of individual services and does not consider the connection between services. In fact, there is a mutual invocation relationship between services, and they are not independent individuals. If the interconnection between services is not considered, the accuracy of service clustering will be affected. In addition, most of the current clustering algorithms select the cluster position randomly. However, in a real web environment, this usually leads to poor accuracy of the clustering algorithm. Therefore, it is more difficult to dig out the interconnections between services. Now, most service clustering technology mainly uses LDA model and *k*-means algorithms to work in the same field. Generally, the existing work has the following two shortcomings. Firstly, the semantic relationship between words is not fully considered, leading to unsatisfactory service discovery results. Secondly, the interconnection between services is not fully considered, resulting in low service clustering accuracy.

We propose a service partition method based on particle swarm fuzzy clustering (NFC-NSPO). Firstly, this method first preprocesses web service description and fully considers the semantic relationship between words. Then, use the automatic clustering algorithm to determine the number of service clusters. Secondly, fuzzy clustering algorithm combines particle swarm optimization, in order to avoid fuzzy clustering algorithm random selected of cluster location and random selection of cluster location caused poor accuracy of fuzzy clustering algorithm. Finally, the fuzzy clustering algorithm is based on Gram-improved cosine to measure the similarity of services. The function based on Gram-improved cosine similarity is used to control the sliding window to compare the service description one by one.

2. Background and Related Work

2.1. Web Service Clustering. In the past, the number diversity of web services has increased rapidly and still keeps emerging [7]. Many researchers pay much attention to service-oriented tasks [8, 9], and service computing developed so fast. Web service clustering is one of the most classical and important tasks in service computing [10, 11].

Service clustering is an integral approach to manage services and assist service discovery [12]. Service clustering is an essential part of service matching, service recommendation, service composition, and service discovery. Service clustering decomposes so many services into a set of smaller clusters to help service engineers manage the system effectively. Service engineers match or recommend a set of services in different clusters according to customer demand.

Web service clustering work can be classified into two types: semantic web services and nonsemantic web services [13]. For the semantic web services, combine keywords extracted from the web service document languages (WSDL). This method describes semantic levels through users' queries and searches by keywords. Clustering based on semantic web services is relatively mature.

Bo et al. present a service clustering based on the functional semantics requirements (SCFSR). It extracts functional information in the service requirement documents by natural language processing, then calculated the similarity between functional information matrix. Finally, apply *k* -means to cluster these services [1]. In paper [14] since web service description documents are short, they use Word2vec to expand the content of description document and then use the LDA topic model to find web services.

Sheeba et al. propose mathematical web service semantic description and registration by ontology. They use an ontology tree to catalog the mathematical web service characteristics, about functional as well nonfunctional [15]. Nguyen and Kuo present a web service discovering through ontology matching semantic relationships. The ontology is built to represent the relationship between semantics with keywords matching, and keywords matching method can find best suitable service for user request [16].

Hsu and Chiu propose a semantic Latent Dirichlet Allocation, which obtains synonym table and then acquisition domain feature word set by Word2vec model. It clusters same domain services, and based on this, builds a framework domain semantic-aided web service clustering [17]. Paper [18] proposes an improved multirelational topic model for web service clustering. Since web service description documents contain limited words, the existing LDA model is hard for short text documents. They care about web service multirelational network, so build a model called MR-LDA. This model consider relationship and annotation relationships and then apply a clustering algorithms to get final results.

For the nonsemantic web service clustering. Service clustering methods do not consider the semantic relationship among services; they pay more attention to the service clustering method. In paper [19], they propose a cluster feature-based latent factor model for Qos prediction. Divide users and services into different groups based on historical records. And that is the same group; users and services have the same latent feature. Furthermore, they design an integrated latent factor model to cluster. In [20], a *k*-means clustering method based on principal component analysis was proposed to predict web services. Solve the problem of low quality accuracy of service prediction caused by the sparse web service matrix.

In short, the particle swarm fuzzy clustering proposed in this paper adopts the fuzzy clustering algorithm based on Gram-improved cosine similarity to consider the connection between services in more detail. At the same time, combining with the particle swarm algorithm can find the optimal service cluster center. It also avoids the fuzzy *c*-means algorithm to randomly select the cluster center, thereby improving the accuracy of service clustering.

2.2. Particle Swarm Optimization Algorithm. The particle swarm optimization algorithm (PSO) simulates the predation behavior of a flock of birds. A flock of birds is searching for food at random, and there is only one piece of food in this area. All the birds do not know where the food is. But they know how far they are from the food. The best way to find food is to search for birds around the area close to the food. The particle swarm optimization algorithm is a kind of bionic evolutionary algorithm [21].

The mathematical model of PSO [21] supposes that in a search space D, the position of the particle is $X_i = \{x_{i1}, x_{i2}, \cdots, x_{\text{in}}\}$, the particle velocity is $V_i = \{v_{i1}, v_{i2}, \cdots, v_{\text{in}}\}$, and p_{best} represents the personal best position of current particle, and g_{best} represents the global best position of current particle, the particle update velocity, and position according to the equation:

$$\begin{aligned} v_{id}^{m+1} &= v_{id}^{m} + c_1 r_1 (p_{id}^{m} - v_{id}^{m}) + c_1 r_1 \left(p_{gd}^{m} - p_{id}^{m}\right), \\ v_{id}^{m+1} &= v_{id}^{m} + v_{id}^{m+1}. \end{aligned} \tag{1}$$

v_{id} represents the velocity of particles, x_{id} represents the position of the particle, m is number of iterations, p_i represents the personal best position of current particle, p_g represents the global best position of current particle, r_1, r_2 represents the acceleration factor which is a random value between 0 and 1, and c_1, c_2 represents the influence degree of personal best and global best position on particle moving direction.

2.3. Fuzzy C-Means Clustering Algorithm. In many problems, the result is only two possibilities, 0 or 1. For example, a student is either a boy or a girl. But this cannot describe the attributes of many things, such as the degree of hot or cold weather. There is no clear definition of what temperature is hot and what is cold. The reason is that in many cases, the boundaries between multiple categories are not absolutely clear. It is needed to use vague words to judge. Fuzzy logic extends the general concept of taking only 1 or 0 (belonging to/not belonging) to taking real numbers between 0 and 1, "Degree of Membership Function." The "Degree of Membership Function" is used to describe the relationship between elements and sets, and the degree of membership is used to express the probability of a sample belonging to a certain class.

Fuzzy c-means clustering algorithm (FCM) is a partition-based clustering algorithm. FCM combines the essence of fuzzy theory. Compared with the hard clustering of k-means, FCM provides more flexible clustering results. Because in most cases, the objects in the dataset cannot be divided into clearly separated clusters, assigning an object to a specific cluster is a bit blunt, and errors may also occur. Therefore, a weight is assigned to each object and each cluster, indicating the degree to which the object belongs to the cluster. Of course, probability-based methods can also give such weights, but sometimes, it is difficult for us to determine a suitable statistical model. Therefore, it is a better choice to use FCM with natural and nonprobabilistic characteristics.

FCM work for service clustering base on the similarity between services in dataset and service clusters c through the iteration of the objective function; the final service clustering result is obtained. The objective function is as follows:

$$\begin{aligned} \text{Min } J_m(U, V, X) &= \sum_{i=1}^{c} \sum_{j=1}^{n} \left(u_{ij}\right)^m d_{ij}^2, \\ \text{s.t.} \sum_{i=1}^{c} u_{ij} &= 1, 1 < j < n, \\ 0 < u_{ij} &< 1, 1 < i < c, 1 < j < n. \end{aligned} \tag{2}$$

$X = \{x_1, x_2, \cdots, x_n\}$ represents service dataset, n is the number of service, $V = \{v_1, v_2, \cdots, v_n\}$ represents c cluster center of service clustering, u_{ij} represents degree of membership of the i service sample belong to cluster c, and d_{ij} represents distance between service sample i and service cluster j. In this paper, d_{ij} applies improved cosine similarity based on Gram.

3. Method

We introduce the presented framework in Section 3.1 and details from Section 3.2 to Section 3.6.

3.1. Framework. The flowchart of the service partitioning method based on particle swarm fuzzy clustering is proposed in this chapter. The method is divided into the preprocessing part of web service description and the service partition method NFC-NSPO based on particle swarm fuzzy clustering (as shown in Figure 1). The left part is the preprocessing part of the web service description. First, crawl the web service description from the programmable website and write it into excel, then extract keywords from excel and filter stop words from it, and finally, restore the word to the stem and use TF-IDF to calculate the frequency of each word. Among them, the preprocessing part is an important part of the service partition method based on particle swarm fuzzy clustering. The right part is the main introduction of NFC-NSPO, a service partition method based on particle swarm fuzzy clustering, which is divided into the following steps. The first step is to identify the number of service clusters and use it as the number of particles in the particle swarm algorithm, where each particle is designed into two parts. The first part is the control variable used to identify the number of the service cluster. The second part of the function is the service distribution of the cluster. The second step is to initialize the speed and position of the particles and calculate the fitness value of each particle. The fitness function is a linear combination of the overall compactness evaluation function and the fuzzy separation function. The third step is to update the speed and position of each particle and repeat the process; if the output condition is met, the output is performed; if the condition is not met, return to the third step. The output result is based on

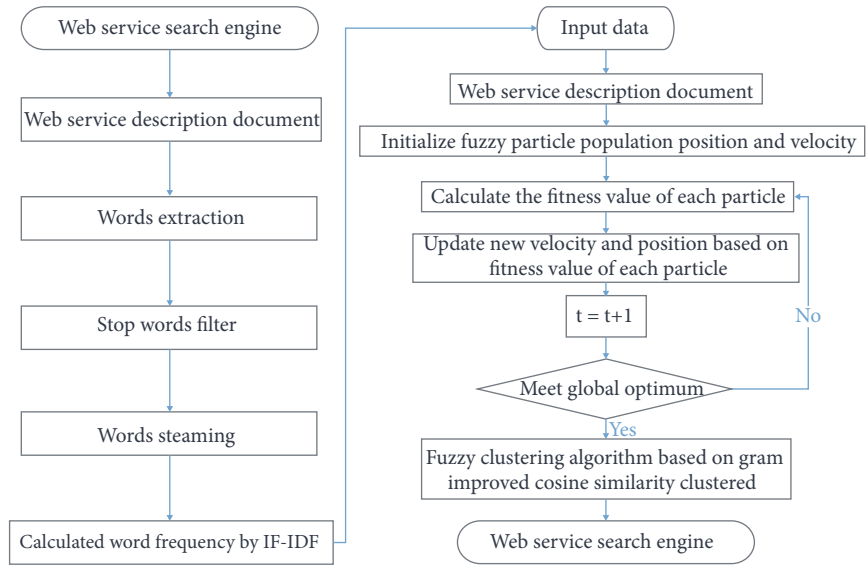


Figure 1: Flow chart of service partition method based on particle swarm fuzzy clusterin.

Gram-improved cosine similarity fuzzy clustering algorithm. Clustering obtains the final clustering result.

3.2. Identify the Number of Service Clusters (K). In the service clustering algorithm, the number of clusters plays a vital role in the accuracy of service clustering. Most existing clustering algorithms used empirical rules that is $k \leq \sqrt{n}$ to determine the number of clusters, k represents the number of service clusters, and n is the number of service samples. There are some drawbacks about empirical rules to identify k the number of service clusters, for large datasets. The number of service clusters k will be very large, which will increase the time complexity of service clustering. For small datasets, k, the number of service clusters will be greater than or equal to the number of samples.

In the study of how to identity the number of service clusters [22], most of the ideas about identifying the number of service clusters is based on the local density of service sample points. The center of the service cluster is surrounded by other services, so the local density center of the service cluster is larger than noncenter density. For example, [23] proposes a clustering algorithm using relative KNN kernel density called RECOME. Firstly, this algorithm is to determine the core object, also known as the cluster center. Secondly, sort according to the local density of the core object. Finally, the point with the highest density is selected as the first cluster center, so that the adjacent noncore object data points form a cluster, and the other data points repeat this process became clusters.

Since RECOME algorithm [23] is only suitable for numerical data, this paper solves the problem of how to determine the number of service clustering clusters. The service description data WSDL belongs to the text, so the formula for calculating the density of core numerical data should be modified to the density formula for calculating the core points of service.

Let X is a set of n service data objects with m attributes. Each service x_i can be represented as a set with m classification attributes as features. Therefore, a certain service x_i can be expressed as $x_i = \{x_{i1}, x_{i2}, \cdots, x_{im}\}$, and the core point density of the service x_i is expressed as $\mathrm{Dens}(x_i)$, so density of each objects can be defined as follows [24]:

$$\mathrm{Dens}(x_i) = \frac{\sum_{l=1}^{m} \mathrm{Dens}_l(x_i)}{m},$$
$$\mathrm{Dens}_l(x_i) = \frac{\left|\left\{x_j \in X \mid x_{il} = x_{jl}\right\}\right|}{n}. \tag{3}$$

For each attribute $l \in m$, $\mathrm{Dens}(x_i) = 1/n$, if:

$$\left|\left|\left\{x_j \in X \mid x_{il} = x_{jl}\right\}\right| = 1 x_{il} = x_{jl}\right| = 1. \tag{4}$$

Otherwise, $\mathrm{Dens}(x_i) = 1$ if:

$$\left|\left|\left\{x_j \in X \mid x_{il} = x_{jl}\right\}\right| = n x_{il} = x_{jl}\right| = n. \tag{5}$$

Therefore, the density of a categorical object is limited at $1/n \leq \mathrm{Dens}(x_i) \leq 1$. However, $\mathrm{Dens}(x_i) = 1$ is very rare because it means that the two services are completely similar.

Services adjacent to the service core point are defined as services in noncore areas. dc is a cutoff distance, which represents the distance between the service core point and the service noncore point. To calculate the similarity between the two services using Gram-improved cosine similarity, choose the number of nearest neighbor services about 1% to 2% of the total number of services by Rodriguez and Laio [22]. Specifically, first calculate the density $\mathrm{Dens}(x_i)$ of service x_i and calculate the neighboring service $Neib_{xi}$ of service x_i, and sort $\mathrm{Dens}(x_i)$ in descending order. Second, determine the cutting distance dc. Finally, by identifying the core object and its neighbors several times to form atom clusters, get the number of clusters in each service dataset. The algorithm for identifying the number of services in this paper adopts the literature [16].

Table 1: Representation of particles.

	C	1	2	3	⋯	N
1	c_1	w_1	w_2	w_3	⋯	w_n
…	…	…	…	…	…	…
k	c_k	w_{k1}	w_{k2}	w_{k3}	⋯	w_{kn}

Table 2: The particle representation of $k_{\max} = 4$, $n = 6$.

	C	1	2	3	4	5	6
1	0.6	0.9	0.8	0.8	0.4	0.8	0.5
2	0.1	0	0	0	0	0	0
3	0.8	0.7	0.7	0.3	0.6	0.2	0.5
4	0.4	0	0	0	0	0	0

3.3. Particle Swarm Representation in Web Service. As shown from Table 1, the particle swarm represents a web service clustering fuzzy matrix whose size is $k \times (1 + n)$, where k represents the maximum number of clusters $k_{\max}$ and where n is the number of web services. Cluster control variable in the particle is used to identify the number of clusters that should be defined in web service, which is from 0 to 1. In this case, if the cluster control variable is larger than 0.5 or equal to 0.5, it means the fuzzy membership function will assign web service objects to cluster based on the control variable. On the other hand, if the cluster control variable is less than 0.5, there does not exist a cluster in the variable, and the fuzzy membership values are 0. The web service clusters' assignment is a fuzzy membership matrix $W = (w_{ij})$, $j = 1, 2, \cdots k_{\max}$, when $k_{\max} = 4$, $n = 6$, the specific way of expressing particles as fuzzy matrix is shown in Table 2.

3.4. Fitness Function. The fitness function is the linear combination of the compactness function and the fuzzy separation function, clustering compactness (π), and fuzzy separation (sep) to evaluate the clusters. If clustering compactness (π) is smaller, it represents the clusters is tighter. If fuzzy separation (sep) is larger, it means the distance between clusters is larger, and the gap in different clusters is larger. This function is calculated as follows:

$$\mathrm{Fit}(x_i) = \pi + \mathrm{sep}, \tag{6}$$

$$\pi = \sum_{j=1}^{k} \frac{\sum_{i=1}^{n} w_{ij}^{\alpha} d(x_i, z_j)}{\sum_{i=1}^{n} w_{ij}^{\alpha}}. \tag{7}$$

$$\mathrm{sep} = \sum_{j=1}^{k} \sum_{l=1, l \neq j}^{k} w_{ij}^{\alpha} d(z_j, z_l), \tag{8}$$

where $W = (w_{ij})$ is the fuzzy membership matrix, $k = k_{\max}$, $Z = \{z_1, z_2, \cdots, z_k\}$ is the set of web service cluster centers, α is the weight, $d(x_i, z_j)$ presents the distance between web services i with clusters j, and $d(z_i, z_j)$ is the distance from cluster j and l.

3.5. Particle Swarm Algorithm Procedure. Kennedy et al. proposed a particle swarm algorithm by observing the trajectory of birds looking for food and conducting research [25]. In the particle swarm algorithm, individuals are called particles. The algorithm includes the following steps:

Step1. Initializing particle swarm

N particle population, each particle consists of two parts, control variables and cluster assignment. Control variables identify how many clusters are active. Next, set the velocity and position of the initial particle. The initialization process of particles: first, randomly generate control variables $C = (c_1, c_2, \cdots, c_k)$ from 0 to 1 for all particles, denoted as $C(p)$, $k = k_{\max}$. After calculating the number of active clusters in each particle, when $c_j \geq 0.5$, according to the initialization process in [26], a fuzzy membership matrix with active clusters $h(p)$ is generated; otherwise, no partitioning is performed. Finally, use $h(p)$ to get the $W(p)$ allocated by the cluster. The initialization process of particle velocity: first, randomly generate control variables $V_C = (v_1, v_2, \cdots, v_k)$ for all particles, denoted as $V_c(p)$, $k = k_{\max}$. Finally, use $h(p)$ to get the cluster distribution speed Vw. Note that during the initialization process, the number of cluster activities $h(p)$ must ensure that $k_{\min} \leq h(p) \leq k_{\max}$.

Step 2. Use formula (8) to calculate the fitness function value of each particle and record the number of iterations

Step 3. For each particle, compared with the fitness value, $\mathrm{Fit}(x_i)$ is compared with p_{best}(individual extremum), if $\mathrm{Fit}(x_i) > p_{\mathrm{best}}$, $\mathrm{Fit}(x_i)$ value replaced p_{best}; otherwise, p_{best} keeps before value

Step 4. For each particle, the fitness value $\mathrm{Fit}(x_i)$ is compared with g_{best} (local extremum), if $\mathrm{Fit}(x_i) > g_{\mathrm{best}}$, $\mathrm{Fit}(x_i)$ value replaced g_{best}; otherwise, g_{best} keeps before value

Step 5. Update new position and velocity for all particle; the new position and velocity are updated as follows:

$$V_c^{t+1}(p) = w \times V_c^t(p) + c_1 r_1 \left(p_{\mathrm{best}_c}(p) - C^t(p)\right) + c_2 r_2 \left(g_{\mathrm{best}_c} - C^t(p)\right),$$

$$C^{t+1}(p) = C^t(p) + V_c^{t+1}(p). \tag{9}$$

w is the inertia weight, $C^t(p)$, $V_c^t(p)$, respectively, represent particles p position and speed, c_1,c_2 is positive acceleration constant, representing local and global learning ability of particles, and r_1,r_2 are random numbers in intervals. In the update process, the value of the control variable will be greater than 1 or less than 0 as the velocity value in the particle changes. If $C(p) > 1$, then adjust to 1. If $C(p) > 1$, adjusted to 0. The update position of control variables may lead to changes in the number of active clusters. Therefore, the number of active clusters will also be updated (as shown in formula (13)):

$$h^{t+1}(p) = \mathrm{count}\left(C^{t+1}(p) \middle| c_j > 0.5\right). \tag{10}$$

In addition, in order to increase the flexibility of membership function, this paper uses the degree of hesitation of

IFS Sugenos [27] to add to the speed of cluster allocation. Like the traditional particle swarm algorithm, the algorithm for updating the velocity and position of the particles adopts literature [16].

Step 6. If the condition is satisfied, exit; otherwise, return step 2

3.6. Improved Cosine Algorithm Based on Gram. Gram algorithm [28] considers that the topic of the service is closely related to the words in the description of the service. So, the probability of the words in the service description is used to describe the topic of the service, and the probability of the topic data is its Gram value. If the Gram value is higher, the services are more similar. The Gram algorithm process is as follows:

(1) Dataset preprocessing includes deleting special characters and filtration stop words

(2) Establish corpus, count the number of each word, and record it as N

(3) Each service topic is closely related to the word. Markov probability formula is used to calculate the probability of a topic word as shown in

$$P(S) = P(c_1)P(c_2 \mid c_1)P(c_3 \mid c_2c_1) \cdots P(c_{n-1} \mid c_{n-2} \cdots c_1). \tag{11}$$

$P(S)$ is the probability of the entire data, and c represents the relative topic words.

Since the probability of a certain word in the service is only related to the previous word. Formula (12) can be simplified to formula (13), reducing unnecessary operations.

$$P(S) \approx P(c_1)P(c_2 \mid c_1)P(c_3 \mid c_2) \cdots P(c_n \mid c_{n-1}), \tag{12}$$

$$P(S) \approx \frac{N(c_1c_2)}{N(c_2)} \times \frac{N(c_2c_3)}{N(c_2)} \times \frac{N(c_3c_4)}{N(c_3)} \cdots \frac{N(c_{n-1}c_n)}{N(c_{n-1})}. \tag{13}$$

Gram algorithm uses a sliding window to assist the measure of service similarity. When the Gram value is small, the service window is expanded to accelerate the measure of service similarity. When the Gram value in the sliding window is large, the window is narrowed to improve the accuracy of service similarity calculation and improve the accuracy of a service clustering algorithm. Gram dynamic sliding window size calculation is shown in formula (14).

$$W_{i+1} = \begin{cases} W_i + n, S_{i+1} < S_i, \\ W_i, S_{i+1} = S_i, \\ W_i - n, S_{i+1} > S_i, \end{cases} \tag{14}$$

where n is the dynamic change of the window. W_i represents the size of the window of i the service data, which is updated by the variance of the service data i. S_i represents the variance of the Gram value of the service data in the window.

The cosine similarity value is obtained from the word frequency vector. As shown in formula (15),

$$\cos(\theta) = \frac{\sum_i^n (a_i + b_i)}{\sqrt{\sum_i^n (a_i)^2 \times \sum_i^n (b_i)^2}}. \tag{15}$$

n is the number of service samples, and the vector a_i, b_i represented as two services.

4. Experiment

Firstly, the details of preprocessing and evaluation metrics were introduced. Secondly, compared with other algorithms in terms of entropy, accuracy, recall, and F value.

4.1. Preprocessing

4.1.1. Remove Stop Words. According to our observation of WSDL documents. We found some words do not have practical meaning as stop words ("be," "the," "a," and "an"). We will filter the stop word in order to filter the noise of the data.

4.1.2. Stemming. WSDL descriptions are written in English, and the same words will be different for different people and tenses. For example, "change" and "changed" have the same meaning, but the computer will consider them to have different meanings. Therefore, we process such words to improve the accuracy of NFC-NSPO through python NLTK (Natural Language Toolkit).

4.1.3. TF-IDF. The TF-IDF algorithm calculates feature words in a document while it offers the frequency. In this paper, we use TF-IDF to calculate the frequency of web service document words and then generate a word frequency matrix.

$$\begin{aligned} tf_{ij} &= \frac{n_{ij}}{\sum n_{ij}}, \\ idf_i &= \log\left(\frac{N}{n_i}\right). \end{aligned} \tag{16}$$

n_{ij} represents the number of j^{th} word in the i^{th} service, $\sum n_{ij}$ donates the total number of words, and idf_i measures the importance of word. N represents the number of web service descriptions, and n_i presents the number of n_{ij} in the web service description.

4.2. Evaluation Measures. This section introduces evaluation measures and comparison algorithms.

It is important to evaluate the performance of the algorithm. We choose widely employed metrics: entropy, recall, accuracy, and F value to assess the performance of NFC-NSPO in web services. The four metrics are shown as follows. We can compare the results with the class label. The accuracy rate is shown in formula (18). The recall rate is

shown in formula (17). The entropy is shown in formula (20). The value of F is as shown in formula (19).

$$\text{Recall} = \frac{\text{succ}(c_i)}{\text{succ}(c_i) + \text{missed}(c_i)}, \tag{17}$$

$$\text{Precision} = \frac{\text{succ}(c_i)}{\text{succ}(c_i) + \text{mispl}(c_i)}, \tag{18}$$

$$F = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{19}$$

$$r = -\sum_{k=1}^{c} P_{ij} \log_2 P_{ij}, \tag{20}$$

$$P_{ij} = \frac{\|C_k^t\|}{\|C_k\|}, \tag{21}$$

where c_i represents the cluster i, $\text{succ}(c_i)$ represents the number of web service put correct cluster c_i, $\text{mispl}(c_i)$ represents the number of web service false put in cluster c_i, and $\text{missed}(c_i)$ represents the number of web service should be in cluster c_i but put it in other clusters. P_{ij} represents the probability that the services belongs to the cluster.

There are several classic clustering methods, such as k-means [29] and PSO-$k$means algorithms [30]. It has been applied to web service recommendations and service combinations. These algorithms have achieved good results. k-means, k-modes, and k-prototype algorithms represent different feature algorithms. For PSO-$k$means, it is similar to our algorithm. We conduct experiments on the same dataset.

K-means: for the k-means algorithm, the first thing to pay attention to is the choice of the k value. Generally speaking, we will choose an appropriate value of k based on the prior experience of the data. k-means divides the sample set into k clusters according to Euclidean distance [29].

K-modes: K-modes is a method used by k-means on nonnumerical sets. Replace the Euclidean distance used by k-means with the Hamming distance between characters [31].

K-prototype: K-prototype is a collective form of k-means and k-modes, which is suitable for data of the numerical type and character type collection. But the web service has only a small amount of numerical data. To some extent, the k-prototype has weakened into k-mode and k-means [32].

PSO-$k$means: the literature [30] proposes a k-means algorithm based on particle swarm optimization (PSO). It makes the k-means algorithm unaffected by the initial cluster centers.

In web service clustering, for k-means and k-modes, they need to give the value of k in advance, and the clustering effect is affected by the cluster center. The particle swarm fuzzy clustering method we proposed uses an automatic clustering algorithm to determine the number of service clusters, and the fuzzy clustering algorithm combines a particle swarm optimization algorithm to determine the location of the cluster center. Although k-means and k-modes have improved, they still cannot get rid of the limitation of k. K-prototype is a combination of k-means and k-modes, and the cluster center is updated by combining K-means and K-modes. PSO-$k$means is similar to our proposed algorithm. It makes the k-means algorithm unaffected by the initial cluster centers. In order to increase the accuracy of clustering, our algorithm uses Gram-improved cosine similarity to measure the similarity between services. We conducted experiments on real web service datasets, and the experiments proved that our algorithm is better than these four algorithms.

4.3. Dataset. In this paper, the dataset of web service text data crawl from a programmable website by python. ProgrammableWeb is a public web service repository. The method of obtaining our dataset is the same as that of the literature [33, 34] using python crawler. The website is https://www.programmableweb.com/. These datasets are statistically calculated, including the number of each service statistics. The number of services is in descending order; it can be seen from Figure 2 that the number of Mapping services is up to nearly 1000. In the dataset, there are more than 200 services that are less than 300, such as search, social, eCommerce, photos, and music. The number of other services is less than 200. Search, social, eCommerce, photos, music, and other services are selected for clustering in this experiment.

The name and number of each service in the service dataset are shown in Table 3. The most significant number of services in the first column is 286 named search, and the smallest number of services is 229 called music. The most significant number of services in the second column is 95 named Government, and the smallest number of services is 52 named Movies. In the third column, the most significant number of services is 46 named Financial, and the smallest number of services is 32 called Books.

The web service sample data obtained includes service names, labels, descriptions, and categories (as shown in Table 3).

4.4. Analysis of Results. In this paper, MATLAB R2016a is used to generate the experimental results. In order to avoid the contingency of the experiment, each clustering algorithm run 10 times, and the average value of the running results is each algorithm's final clustering result. Accuracy, entropy, recall, and F value are used to evaluate each clustering algorithm. The MATLAB code and dataset are available at https://github.com/dqy1122/PSOcmeans.git.

As shown from Table 4, in terms of accuracy, the highest accuracy of NFC-NSPO algorithm is 0.896. The second is PSO-$k$means algorithm. Its accuracy is 0.845. The worst is k-prototype clustering algorithm. The accuracy is 0.743. In terms of recall rate, the highest recall rate of k-modes algorithm is 0.756. Next is the NFC-NSPO algorithm. Its value is 0.734. The worst is k-means algorithm; its value is 0.621. In terms of entropy, k-prototype clustering algorithm has the maximum entropy value of 0.781. The second is PSO-$k$means algorithm; its value is 0.772. The worst is NFC-NSPO algorithm; its value is 0.642. In terms of F value, the F value of NFC-NSPO algorithm is the highest, which is 0.806. The second is

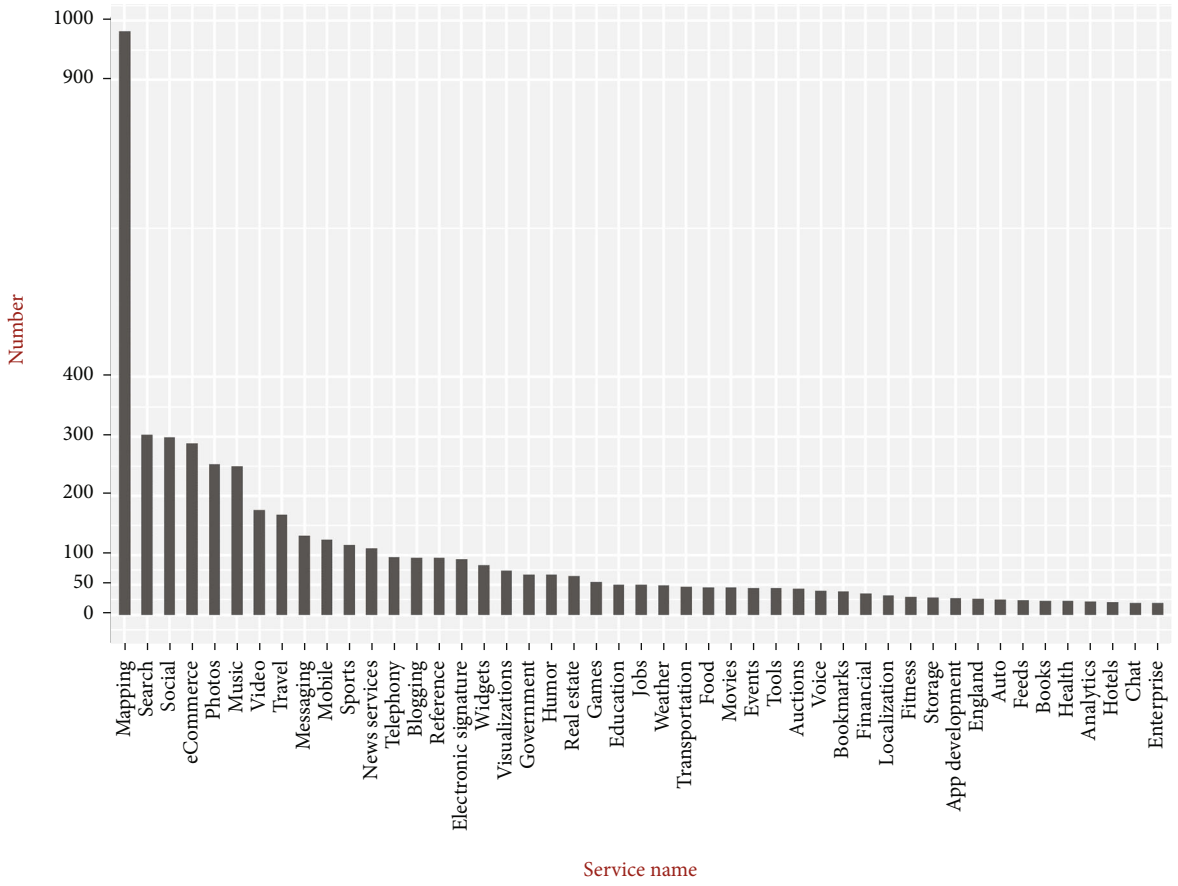


Figure 2: Statistics of service datasets.

Table 3: The number of services.

Name	Number	Name	Number	Name	Number
Search	286	Government	95	Bookmarks	32
Social	276	Humo	97	Financial	46
eCommerce	245	Real estate	93	Localization	46
Photos	236	Games	95	Fitness	45
Music	229	Movies	52	Books	32

Table 4: The accuracy of each algorithm.

Algorithm name	Accuracy	Recall	Entropy	F value
k-means	0.756	0.621	0.721	0.682
k-modes	0.825	0.756	0.762	0.789
k-prototype	0.743	0.738	0.781	0.733
PSO-$k$means	0.845	0.712	0.772	0.772
NFC-NSPO	0.896	0.734	0.642	0.806

the k-modes algorithm, whose value is 0.789. The worst is PSO-$k$means algorithm. Its value is 0.772.

The reason analysis is as follows: NFC-NSPO fuzzy clustering service partition method based on particle swarm optimization. Firstly, the improved cosine similarity calculation based on Gram is used to calculate the similarity between services. Gram algorithm uses a sliding window to assist the service similarity. When the Gram value in the window is small, the service window should be expanded to accelerate the detection of service similarity. When the Gram value in the sliding window is large, the window should be narrowed to improve the accuracy of service clustering. Secondly, using the advantages of the particle swarm optimization algorithm, the optimal global solution can be found through the movement of particles. It avoids the problem that fuzzy clustering algorithm randomly selects the clustering center and falls into the local optimum. Therefore, the clustering accuracy of NFC-NSPO is improved. In k-means, Euler distance is used to measure the similarity between different services. Since Euler distance is not suitable for calculating the similarity of text data, the calculation of service similarity is not accurate enough. At the same

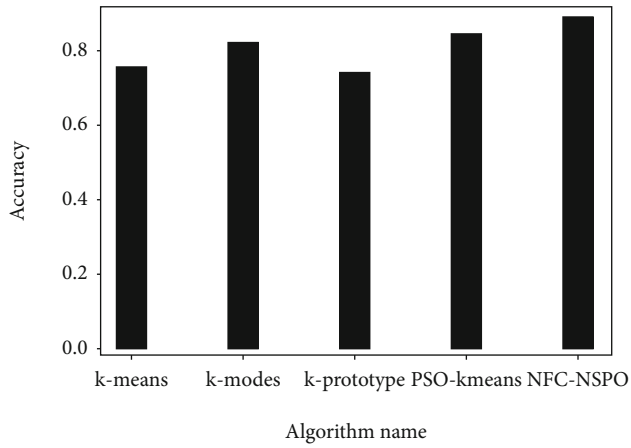


Figure 3: The accuracy of each algorithm.

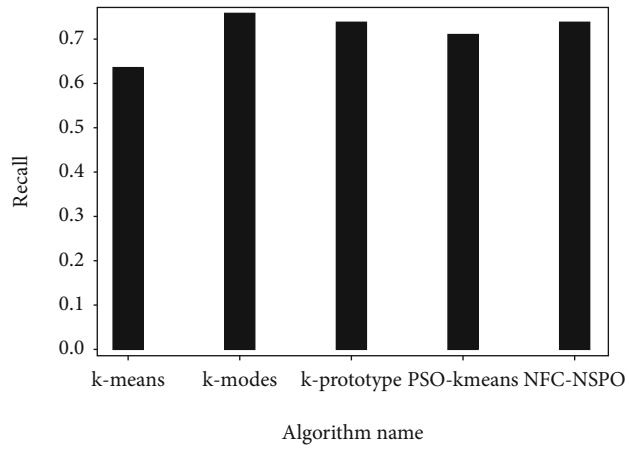


Figure 4: Recall rate of each algorithm.

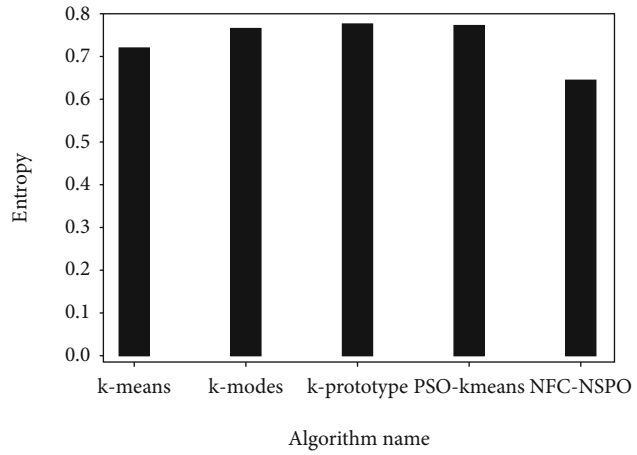


Figure 5: Entropy of each algorithm.

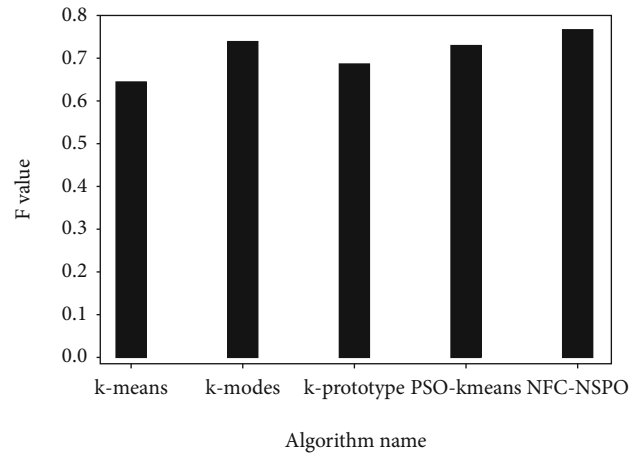


Figure 6: F value of each algorithm.

time, k-means belongs to the partition clustering algorithm. The location and number of clusters are randomly selected so it is easy to fall into the local optimum, thereby affecting the accuracy of the clustering algorithm.

As Figure 3 shown, the accuracy of NFC-NSPO peaked at 0.896. At the same time, the accuracy of the PSO-K means algorithm and k-prototype algorithm is smaller than NFC-NSPO, with 0.845 and 0.734, respectively. Some reasons cause NFC-NSPO higher accuracy. Firstly, NFC-NSPO applies improved cosine similarity based on Gram, which can better calculate the similarity between two services. Secondly, fuzzy clustering algorithm combines particle swarm algorithm, to avoid fuzzy clustering algorithm random selection of cluster location. PSO-means algorithm uses Euler distance to calculate the similarity between two services. It will lead to low service clustering accuracy, because Euler distance is not suitable for calculating service similarity.

In Figure 4, the recall rate of the k-modes algorithm reached the highest, followed by NFC-NSPO and PSO-k means. The similarity calculation of k-modes used Hamming distance to measure the similarity between services. By comparing whether each bit of the vector is the same or not. If the vectors were different, the Hamming distance increased by 1. Otherwise, the Hamming distance remained unchanged.

The entropy value represents the degree of the chaos of an object. If the entropy value is larger, it means that objects are chaotic. If the entropy value is smaller, the object is stable, and the chaos coefficient is low. Figure 5 shows the entropy value of k-prototype reached the largest, followed by PSO-$k$means, and the entropy value of NFC-NSPO algorithm fall the lowest. On the one hand, the k-prototype algorithm is an improved algorithm combining k-means and k-modes, which can deal with both numerical data and categorical data. Since web service description text has only a small amount of numerical data. To some extent, k-prototype is the similarity to k -modes. On the other hand, the k-prototype algorithm is easily affected by the position of the cluster center, which is easy to fall into local optimum. So the k-prototype algorithm is unstable.

The F value is a linear combination of accuracy and recall, which measures the performance of the algorithm in a more stable form. Figure 6 shows that F value of NFC-NSPO reached the highest, followed by k-modes, and F value of k-prototype falls the lowest.

Table 5: The accuracy of each algorithm.

Algorithm name	Accuracy	Recall	Entropy	F value
NFC-NSPO (Gram-cosine)	0.896	0.734	0.713	0.806
NFC-NSPO (Cosine)	0.842	0.710	0.747	0.770
NFC-NSPO (Euler)	0.638	0.653	0.773	0.637
NFC-NSPO (Hamming)	0.801	0.624	0.752	0.701
NFC-NSPO (Manhatten)	0.743	0.612	0.781	0.671

Figure 3 shows that the accuracy of NFC-NSPO is significantly higher than other algorithms. Figure 4 shows that the recall value of NFC-NSPO is lower than that of the k-modes algorithm. The recall rate of NFC-NSPO is still higher than other clustering algorithms, because NFC-NSPO clustering algorithm applies the improved cosine similarity based on Gram, which can better calculate the similarity between two samples. On the other hand, fuzzy clustering algorithm combined particle swarm algorithm, to avoid fuzzy clustering algorithm random selected of cluster location; the random selection of cluster location caused poor accuracy of a fuzzy clustering algorithm.

The similarity measure plays a very important role in the clustering algorithm. Even if the same clustering algorithm uses different similarity measures, the accuracy of the clustering algorithm is different. This paper improves the similarity measure, which combined the Gram algorithm and cosine measure. To verify the performance of the improvement similarity measure. We use the same algorithm and use different similarity functions to compare accuracy, recall, entropy, and F value.

In Table 5, the accuracy of NFC-NSPO (Gram-Cosine similarity) is the highest with 0.896. Followed by NFC-NSPO (Cosine similarity) with 0.842, the accuracy of NFC-NSPO (Euler) falls lowest. In terms of recall, NFC-NSPO (Gram-Cosine similarity) reached the highest with 0.734, followed by NFC-NSPO (Manhatten) with 0.612. About entropy, the entropy of NFC-NSPO (Euler) reaches the highest with 0.773, followed by NFC-NSPO (Gram-Cosine similarity) with 0.713. In terms of F value, the F value of NFC-NSPO (Gram-Cosine similarity) reaches the highest, which is 0.806. The F value of NFC-NSPO (Euler) falls the lowest, which is 0.637.

This paper gave a brief analysis of the point of higher performance of NFC-NSPO (Gram-Cosine similarity), which uses the improved cosine similarity based on Gram to better measure the similarity between two services. This method can adjust the window size between services and clusters. This method can improve the accuracy of service clustering algorithm.

In most existing algorithms for automatically determining the number of clusters, k is obtained by means of running many times, which leads to the fact that k is not an integer. For solving this problem, most scholars choose the rounded method to take the integer k value, because the optimal number of clusters k should be an integer. In this paper, the NFC-NSPO algorithm proposed can determine the number of clusters k on six datasets. The NFC-NSPO algorithm calculates the number of clusters k equal to the number of predetermined classes. On the contrast, the rounding method can only correctly provide the optimal k on five datasets. NFC-NSPO algorithm generates that the number k of clusters on eCommerce dataset is the same as the expected value.

5. Conclusion

Because of the interrelationship between services, it is challenging to assign services to a specific cluster accurately. This paper propose a service partition method based on particle swarm fuzzy clustering. It can avoid the random selection of clustering positions, which leads to poor accuracy of the fuzzy clustering algorithm. The fuzzy clustering algorithm applies based on Gram-improved cosine similarity measure the similarity service. The function based on Gram-improved cosine similarity controls the sliding window to compare the service description one by one. When the Gram value is small, the Gram window is expanded to accelerate the measure service similarity. When the Gram value is large, the window is narrowed to improve service similarity measures' accuracy and service clustering accuracy. Experimental results show that the NFC-NSPO algorithm can better evaluate the interconnection between services and improve the accuracy of service clustering compared with existing algorithms. That can reasonably consider the relationship between service and service. Combined with the particle swarm algorithm, the relationship between the two can find the optimal cluster center position.

Data Availability

The MATLAB code and dataset are available at https://github.com/dqy1122/PSOcmeans.git.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the Science and Technology Project in Shaanxi Province of China (program no. 2019ZDLGY07-08), Natural Science Basic Research Program of Shaanxi Province, China (grant no. 2020JM-582), Science and Technology of Xi'an (grant no. 2019218114 GXRC017CG018-GXYD17.9), Scientific Research Program Funded by Shaanxi Provincial Education Department (no. 21JP115), Natural Science Basic Research Program of Shaanxi (program no. 2021JQ-719), and Special Funds for Construction of Key Disciplines in Universities in Shaanxi.

References

[1] J. Bo, H. U. Song, P. Wei-Feng, W. Ye, and S. Bei-Bei, "Service clustering based on the functional semantics of requirements," *Chinese Journal of Computers*, vol. 41, no. 6, pp. 1036–1040, 2015.

[2] Z. Hao-quan and Z. Qi, "Web service discovery clustering performance analysis based on clustering LDA method," *Computer Application*, vol. 39, no. 10, pp. 27–30, 2020.

[3] D. Yang and D. He, "Web service clustering method based on word vector and biterm topic model," in *2021 IEEE 6th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*, pp. 299–304, Chengdu, China, 2021.

[4] Ö. Çoban and G. T. Özyer, "Word2vec and clustering based twitter sentiment analysis," in *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, pp. 1–5, Adana, Turkey, September, 2018.

[5] Z. Xiangping, J. Liu, Q. Xiao, M. Shi, and B. Cao, "Web services clustering with multi-functionality based on lda and fuzzy c-means algorithm," *Journal of Central South University(Science and Technology)*, vol. 49, no. 12, pp. 2986–2992, 2018.

[6] Y. M. Wang, Y. J. Zhang, B. H. Xie, L. H. Pan, and L. C. Chen, "Semantic web service discovery based on fuzzy clustering optimization," *Computer Engineering*, vol. 39, no. 7, pp. 219–223, 2013.

[7] Q. Feng, L. Chen, C. L. P. Chen, and L. Guo, "Deep fuzzy clustering—a representation learning approach," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 7, pp. 1–1433, 2020.

[8] C. Lv, W. Jiang, S. Hu, J. Wang, G. Lu, and Z. Liu, "Efficient dynamic evolution of service composition," *IEEE Transactions on Services Computing*, vol. 11, no. 4, pp. 630–643, 2018.

[9] Y. Yin, J. Xia, Y. Li, xu, W. Xu, and L. Yu, "Group-wise itinerary planning in temporary mobile social network," *IEEE Access*, vol. 7, pp. 83682–83693, 2019.

[10] H. Gao, K. K. Dluzniak, H. Xia et al., "A service clustering method based on wisdom of crowds," in *2019 IEEE International Congress on Big Data (BigDataCongress)*, pp. 98–105, Milan, Italy, July, 2019.

[11] C. Cho, K. Lee, M. Lee, and C. Lee, "Software architecture module-view recovery using cluster ensembles," *IEEE Access*, vol. 7, pp. 72872–72884, 2019.

[12] B. Cao, X. F. Liu, M. M. Rahman, B. Li, J. Liu, and M. Tang, "Integrated content and network-based service clustering and web apis recommendation for mashup development," *IEEE Transactions on Services Computing*, vol. 13, no. 1, pp. 99–113, 2020.

[13] T. Liang, Y. Chen, W. Gao, M. Chen, M. Zheng, and J. Wu, "Exploiting user tagging for web service co-clustering," *IEEE Access*, vol. 7, pp. 168981–168993, 2019.

[14] Q. Xiao, B. Cao, X. Zhang, J. Liu, and L. I. Yanxinwen, "Web services clustering based on word2vec and lda topic model," *Journal of Central South University(Science and Technology)*, vol. 49, no. 12, pp. 2979–2985, 2018.

[15] A. Sheeba, S. Padmakala, and C. A. Subasini, "Ontology based semantic description and registration of mathematical web services," in *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pp. 521–525, Coimbatore, India, February, 2019.

[16] T. P. Q. Nguyen and R. J. Kuo, "Automatic fuzzy clustering using non-dominated sorting particle swarm optimization algorithm for categorical data," *IEEE Access*, vol. 7, pp. 99721–99734, 2019.

[17] C.-I. Hsu and C. Chiu, "A hybrid latent dirichlet allocation approach for topic classification," in *2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, pp. 312–315, Gdynia, Poland, July, 2017.

[18] M. Shi, J. X. Liu, D. Zhou, B. Q. Cao, and Y. P. Wen, "Multi-relational topic model-based approach for web services clustering," *Chinese Journal of Computers*, vol. 42, no. 4, pp. 820–836, 2019.

[19] S. Chen, Y. Peng, H. Mi, C. Wang, and Z. Huang, "A cluster feature based approach for QoS prediction in web service recommendation," in *2018 IEEE Symposium on Service-Oriented System Engineering (SOSE)*, pp. 246–251, Germany, March, 2018.

[20] H. Yang, H. Yan, and C. Dong, "A k-means clustering approach for PCA-based web service QoS prediction," in *2019 IEEE International Conferences on Ubiquitous Computing Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS)*, pp. 129–132, Shenyang, China, October 2019.

[21] P. C. Fourie and A. A. Groenwold, "The particle swarm optimization algorithm in size and shape optimization," *Structural and Multidisciplinary Optimization*, vol. 23, no. 4, pp. 259–267, 2002.

[22] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[23] M. Liang, Q. Li, Y. A. Geng, J. Wang, and Z. Wei, "Remold: an efficient model-based clustering algorithm for large datasets with spark," in *2017 IEEE 23rd International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 376–383, Shenzhen, China, December, 2017.

[24] F. Cao, J. Liang, and L. Bai, "A new initialization method for categorical data clustering," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10223–10228, 2009.

[25] J. Kennedy, R. C. Eberhart, and Y. Shi, "The particle swarm," *Swarm Intelligence*, pp. 287–325, 2001.

[26] I. Heloulou, M. S. Radjef, and M. T. Kechadi, "A multi-act sequential game-based multi-objective clustering approach for categorical data," *Neurocomputing*, vol. 267, pp. 320–332, 2017.

[27] K. T. Atanassov, "Intuitionistic fuzzy sets," *Fuzzy Sets & Systems*, vol. 20, no. 1, pp. 87–96, 1986.

[28] A. Ahmad, M. Rub Talha, M. Ruhul Amin, and F. Chowdhury, "Pipilika n-gram viewer: an efficient large scale n-gram model for bengali," in *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*, pp. 1–5, Sylhet Bangladesh, September, 2018.

[29] I. Alhadid, S. Khwaldeh, M. al Rawajbeh, E. Abu-Taieh, R.'e. Masa'deh, and I. Aljarah, "An intelligent web service composition and resource-optimization method using K-means clustering and knapsack algorithms," *Mathematics*, vol. 9, no. 17, p. 2023, 2021.

[30] M. Handa, H. Xiaoyu, and M. Renqing, "Parallel PSO-kmeans algorithm implementing web log mining based on hadoop," *Computer Science*, vol. 42, no. S1, pp. 470–473, 2015.

[31] O. S. Soliman, D. A. Saleh, and S. Rashwan, "A hybrid fuzzy particle swarm and fuzzy k-modes clustering algorithm," in *8th International Conference on Informatics and Systems (INFOS)*, pp. 68–75, Giza, Egypt, July, 2012.

[32] X. Chen, "An improved clustering algorithm for mixed attributes data based on K-prototypes algorithm," in *2015 10th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA)*, pp. 396–399, Krakow, Poland, March, 2015.

[33] Z. Neng, W. Jian, and M. Yutao, "An intelligent web service composition and resource-optimization method using K-means clustering and knapsack algorithms," *IEEE Transactions on Services Computing*, vol. 13, no. 3, pp. 488–502, 2020.

[34] N. Zhang, K. He, J. Wang, and Z. Li, "WSGM-SD: an approach to RESTful service discovery based on weighted service goal model," *Clarivate Analytics Web of Science*, vol. 25, no. 2, pp. 256–263, 2016.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 2438972, 10 pages
https://doi.org/10.1155/2021/2438972

Research Article

A Utility Method for the Matching Optimization of Ride-Sharing Based on the E-CARGO Model in Internet of Vehicles

Xiaohui Li,[1,2] **Hongbin Dong**,[1] **Shuang Han**,[1] **Xiaowei Wang**,[1] **and Xiaodong Yu**[3]

[1]*College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China*
[2]*Harbin Vocational & Technical College, Harbin 150081, China*
[3]*Computer Science and Technology College, Harbin Normal University, Harbin 150080, China*

Correspondence should be addressed to Hongbin Dong; donghongbin@hrbeu.edu.cn

Received 29 September 2021; Accepted 22 October 2021; Published 12 November 2021

Academic Editor: Yingjie Wang

The Internet of Vehicles (IoV) is the extension of the Internet of Things (IoT) technology in the field of transportation systems. Ride-sharing is one of intelligent travel applications in IoV. Ride-sharing is aimed at taking passengers with similar itineraries and time arrangements to travel in the same car according to a certain matching rule. To effectively integrate transport capacity resources and reduce the number of cars on the road, ride-sharing has become a popular and economical way of travel. The matching and optimizing of drivers and passengers are the core contents of a ride-sharing application system. This paper mainly studies the dynamic real-time matching of passengers and drivers in IoV, considering the main factors such as travel cost, car capacity, and utility. The matching problem is formulated in a ride-sharing system as a Role-Based Collaboration (RBC). A new utility method for the matching optimization of ride-sharing is present. In this paper, we establish a model for simulating the assignment of ride-sharing with the help of the Environments-Classes, Agents, Roles, Groups, and Objects (E-CARGO) model. The objective function and formal definitions are proposed. The utility and time of optimal matching are obtained by using the Kuhn-Munkres algorithm on the revenue matrix. The experimental results show that the proposed formal method based on the E-CARGO model and utility theory can be applied in the ride-sharing problem. Numerical experiments show that the matching time cost increases with the increase of the number of drivers and passengers participating in the ride-sharing system. When the number of drivers and passengers is different, one-to-many matching takes the least time, and one-to-one matching takes more time. When the number of drivers and passengers is the same, the time cost of one-to-one matching increases sharply with a certain value (bigger than 800). Compared with other matching methods, the time spent by the one-to-many method is reduced by 30%. The results show that the proposed solution can be applied to the matching and pricing in a ride-sharing system.

1. Introduction

The Internet of Vehicles (IoV) is a huge and interactive network composed of vehicle locations, speed, routes, and other information. Through the Internet technology, all vehicles can transmit all kinds of information to the central processing unit. This vast vehicle information can be analysed and processed in the computer technology, so as to calculate the best route of different vehicles, timely report the road conditions, and arrange the signal lamp cycle. Ride-sharing is one of the representatives of intelligent travel applications in IoV.

With the development of economy, the number of private cars is increasing in cities. Although it is convenient for travel with the increase of urban car, it also brings a series of problems, such as traffic congestion, excessive consumption of oil resources, and environmental pollution. Traffic congestion in cities is a universal phenomenon. In order to solve traffic congestion and environmental pollution problems, the matching and optimizing problems have been researched extensively in a ride-sharing system. Technologies such as IoV and GPS real-time positioning provide support and guarantee for ride-sharing application systems. Data sharing is privacy-preserved towards passengers and

Table 1: The ride-sharing relationships between drivers and passengers.

	One passenger	Many passengers
One driver	One-to-one matching between drivers and passengers	One-to-many matching between drivers and passengers according to passengers' routes and time
Many drivers	One passenger transferring to different drivers' car according to his travel routes	One-to-one or one-to-many matching for many times, respectively, according to the routes and time of passengers and drivers

drivers in IoT [1, 2]. The economic aspects of shared mobility systems have attracted more and more applications and attention [3]. Ride-sharing travel services have been proved to be an effective way, which could make full use of traffic resources, alleviate the shortage of parking spaces, reduce environmental pollution, and optimize social benefits [4]. In addition, ride-sharing travel services allow passengers and drivers to share the travel cost, which is also considerable. With the continuous progress of mobile Internet technologies and the popularity of Internet mobile devices, ride-sharing travel, which is simple, safe, flexible, efficient, and economical, will be more popular in the future.

Ride-sharing travel refers to a travel mode. The passengers with the same or similar travel paths are assigned to travel together in the same vehicle according to a matching mode. The passengers share the travel cost together in a certain driving interval. A ride-sharing application system mainly considers to complete the optimal pricing strategy and vehicle scheduling strategy in combination with geographical locations and time information [5]. In a ride-sharing application system, it matches the passenger sending the ride request with the nearby driver whose car has vacant seats. The passenger sends the ride request including the departure place, time, and destination for traveling. Then, the ride-sharing application system returns the approximate fee to the passenger according to a certain pricing rule. After accepting and confirming the fee, the passenger can be match to a nearby driver. In case of one driver and many passengers, when the driver's car still has vacant seats, some passengers with similar paths will be assigned to travel together according to a certain matching strategy. After the completion of the ride-sharing travel, passengers will pay a certain fee. If it is a single passenger mode, only a single passenger can be match to a single driver at the ride-sharing time. As the driver can only match one passenger's request each time, the next matching will be carried out only after the passenger is delivered to the destination.

A driver may only pick up one passenger or may be willing to pick up two or more passengers when there are vacant seats. Similarly, each passenger may choose to take a driver's car alone or share it with other passengers. An effective incentive mechanism [6] can attract more and more passengers and drivers to participate in a ride-sharing system. A cost-sharing mechanism [7, 8] for ride-sharing can solve the problem of vehicles with no passengers in a ride-sharing system. Furthermore, effective and incentive mechanisms can improve the service level of sharing systems [9]. The existing riding-share relationships between the drivers and passengers are shown in Table 1. No matter which travel mode was selected, the matching method of passengers and drivers is the core of a riding-sharing system.

From the perspective of leaders, utility theory is a theory used by leaders in decision-making. From the perspective of consumers, utility theory refers to the satisfaction of consumers from consuming certain goods, which is the core of consumer behavior theory.

In riding-share travel, the passengers' satisfaction can be expressed by their utility. The utility function is used to quantify the satisfaction [10]. The actual utility function describes the quantitative relationship between the utility obtained by passengers and their travel choices in ride-sharing. In addition, the utility function can measure the satisfaction of passengers in different travel choices.

Many problems have been formalized and solved in collaboration [11–13]. Role-Based Collaboration (RBC) was initially proposed to support natural collaboration through computer-based systems. The E-CARGO model is proposed by Canadian scholar Professor Zhu, which denotes Environments—Classes, Agents, Roles, Groups, and Objects. We concentrate on the online dynamic matching problem between passengers and drivers. A utility method for the matching problem of ride-sharing was proposed. In this paper, the method combines RBC and the E-CARGO model [14–17] to model the matching of the ride-sharing problem. Agents, roles, groups, and the relationship between them are defined. The ride-sharing optimization problem is formalized as a group role assignment (GRA) problem. The formal definitions and objective function are present. Under the constraints of vacant seats and acceptable flexible time, the optimization matching scheme is obtained by maximizing the revenue of a ride-sharing system. On the distance profit matrix, Kuhn-Munkres algorithm [18–20] is used to obtain the optimal matching matrix and time based on GRA.

GRA constructs a group by assigning roles to its members or agents to achieve its highest performance. In the proposed model, a driver is regarded as a role, and a passenger is regarded as an agent. In the system, current agents and roles are our focus. The problem is to find a role assignment matrix that makes the group workable, where each agent is assigned at most one role. In this situation, GRA seeks to find a matching matrix with the maximum revenue of the revenue matrix for a ride-sharing system.

According to the matching strategies and core factors affecting passengers' choice of travel in a ride-sharing application system, the specific contributions of this paper are shown as follows.

(1) We propose a new utility method for the ride-sharing matching based on the E-CARGO model

with utility and seat number constraints. We formulate the matching problem in ride-sharing application system as a RBC problem. To better understand the GRA process, we develop a matching model, in which drivers can be match to more than one passenger in a flexible time

(2) This is the first paper that considers the utility between passengers and drivers in terms of matching and pricing based on the E-CARGO model. This paper contributes a novel way to study the matching problem in ride-sharing from the perspective of utility. When the number of drivers and passengers is the same, the utility of one-to-one matching between drivers and passengers is approximately equal to one-to-many matching. When the number of drivers and passengers is different, the utility of one-to-one matching is the least

(3) The experiments present that the utility has an impact on the revenue in a ride-sharing system. The advantages of the proposed method in terms of economic efficiency, matching incentives are demonstrated in the numerical experiments. The results show that our solutions are practical in terms of matching optimization and predicting behaviors

The rest of the paper is organized as follows. The related works are introduced in Section 2. The E-CARGO model, travel utility, model formulation, and algorithm analysis are introduced in Section 3. It formally specifies the ride-sharing problem with the E-CARGO model and utility theory. Experiments and analyses are illustrated in Section 4. This paper concludes and points our further research in Section 5.

2. Related Works

Travel cost can be reduced through cost-sharing in a ride-sharing system. Travelers who do not own private cars make travel more convenient by ride-sharing. Ride-sharing services may be provided by private car owners or public service providers. Gong et al. [21] propose the technique which has been introduced to realize the privacy-preserving distributed service recommendation. If the ride-sharing application system is private, the provider can get commission or advertising revenue. If it is a public system, there may be a social goal, such as reducing pollution and traffic congestion. However, no matter what the nature of the ride-sharing system is, it is necessary to complete the matching of drivers and passengers. Generally speaking, there are some specific objectives to be considered in the matching, such as minimizing the vehicle mileage or the travel time or maximizing the number of participants or revenue in a ride-sharing system. High success matching rate will stimulate more participants to participate in ride-sharing. In this paper, the modeling and matching methods in ride-sharing are proposed.

The application system of ride-sharing travel generally requires passengers to provide departure time. In the area of obtaining participants' time preference, Long et al. [22] present a ride-sharing problem with travel time uncertainty and propose a matching model considering travel time uncertainty. Due to various uncertain factors, the passengers' time preference cannot always be satisfied. Lopez et al. [23] propose a new formal attack model. Their research shows that people are more concerned about the guarantee of travel time. In addition to the time factor, other factors will also affect the success of matching between passengers and drivers. The data uploading in ride-sharing systems suffers novel challenges on privacy preservation [24]. For example, people may prefer to share a car with familiar people. Women feel more secure when taking a female driver's car. In short, the more restrictions on potential passengers when choosing coriding partners, the more difficult it will be to find a successful match for a passenger [25]. Unlike the aforementioned preference, this paper proposed a utility method to measure participants' preference, which is more universal.

If a driver has enough flexible time, he may be willing to provide services for multiple passengers. The driver can pick up passengers one by one or take more than one passenger at the same time. With assigning many passengers to a driver, the ride-sharing application system will provide a feasible optimal route to minimize the travel cost. Javier et al. [26] propose and construct a mathematical model for real-time high-capacity ride-sharing. The model can be extended to a large number of passengers and trips and dynamically generate optimal routes according to online demand and vehicle locations. Calvo and Fabio [27] propose an integrated system for the organization of a ride-sharing service, in which there is an optimization module solving heuristically the specific routing problem. This paper combines the E-CARGO model with utility theory to establish travel matching model, improve the utilization of available seat capacity, maximize platform revenue, and rationalize resource matching.

When people choose to travel together, they consider not only the time factor but also hope to share the travel cost and reduce the cost through ride-sharing. Therefore, many scholars have studied the cost sharing methods in ride-sharing. Wang et al. [28] have studied various cost sharing strategies. They provided the necessary conditions for cost sharing strategies to maintain participation or reduce cars. In literature [29], considering the difference of travel distance, a cost sharing method in proportion to distance is proposed. The cost is apportioned in proportion to the distance. Kleiner et al. [30] proposed a method to determine the driver's compensation based on the auction mechanism. In this method of determining remuneration, it is necessary to take into account the evaluation of drivers. For drivers, the cost they are willing to pay is between the cost of driving a private car alone and the cost of taking a taxi. The higher the compensation and commission for the driver is, the longer the acceptable length of the detour in the driver's mind.

There are also studies focusing on pricing policy in ride-sharing systems. Sun et al. [31] point out that there are fewer restrictions on the access to the ride-sharing market and less strict supervision on pricing. They present the impact of labor supply elasticity on market labor supply from the

perspective of supply and demand in economics. Cachon et al. [32] study the surge pricing strategy. The provider is paid a fixed commission at a dynamic price. It is concluded that all stakeholders can benefit from the platform with self-scheduling ability under the surge pricing strategy. These papers focus on pricing policy but ignore the utility of passengers in a ride-sharing system. When the more utility of passengers is obtained, they will be willing to pay more for a certain route of their travel.

In terms of matching for dynamic ride-sharing systems, Wang et al. [33] introduce some mathematical methods to establish stable matches. They consider the stability of ride-sharing matches and propose optimization approaches. Peng et al. [34] propose a stable matching model for the ride-sharing. The objective function of the model is minimizing travel cost of the passengers. They also illustrate the main factors affecting the successful matching rate. Zkan [35] researches the interplay between pricing and matching decisions of a ride-sharing firm. He proves that the optimization matching strategies effect on the performance of a ride-sharing application system. A utility method for the matching optimization of ride-sharing based on the E-CARGO model is proposed, which describes a real scenario in a ride-sharing problem and considers the overall system's utility.

3. Methodology

3.1. E-CARGO Model. The E-CARGO model is proposed by Canadian scholar Professor Zhu based on the research and discussion of role collaboration theory [36–38]. The E-CARGO model has been applied in assignment problems and recommendation systems. It is the research and extension of the basic theory of role distribution. The experiment shows that this method is efficient and reliable. The E-CARGO model is suitable for modeling social systems and economic systems with the formal analysis. In this paper, the engineering theory method based on role collaboration and the E-CARGO model is abstracted, and the ride-sharing application is modeled.

In the E-CARGO model, a system $\sum$can be described as a nine-tuple $\sum := \langle C, O, A, M, R, E, G, S_0, H\rangle$, where C is a set of classes, O is a set of objects, A is a set of agents, M is a set of messages, R is a set of roles, E is a set of environments, G is a set of groups, S_0 is the initial state of the system, and H is a set of users. In such a system, A and H and E and G are tightly coupled sets. A user and his agent can play a role together. Each group works in the same environment, and the environment has a normative effect on the groups. In the E-CARGO model, each agent plays only one role at a time.

Many problems in reality can be formalized and specified with the E-CARGO model [39–41]. Firstly, the problem is decomposed into subproblems, and the roles and agents are determined. Secondly, the relationships and constraints between roles and agents are described by constraints. Thirdly, agents are assigned to roles and groups. The assignment results are evaluated through the evaluation criteria. Then evaluation and assignments are performed circularly to meet evaluation criteria indicators. Finally, the final assignment scheme is determined. In the following ride-sharing model, we focus on agents (A) and roles (R). The following formal definitions can be taken as a part of the E-CARGO model.

3.2. Travel Utility. In the utility theory, utility refers to the satisfaction that customers get in the process of exchanging goods. Utility function is often used to model different consumption behaviors and measure social welfare or satisfaction of a consumer [42]. The utility function is used to identify different customers' behavior [43]. Passengers usually consider travel experience and actual conditions to select travel modes. In the utility theory, passengers always choose the trip modes that can maximize their personal utility.

The drivers and passengers in a ride-sharing system can be regarded as participating agents. When the utility obtained by each participating agent is strictly greater than that obtained without participating, and no other protocol provides greater utility for all agents, it can be regarded as that all agents spontaneously participate in the system game.

Passengers prefer to travel with acquaintances or similar age people. Due to objective factors affecting travel mode such as travel cost, convenience, and time, there is great randomness to a certain extent. In the utility function, we mainly consider the following core factors:

(i) Expenses incurred in a driver to passenger matching with a profit matrix

(ii) Time cost

(iii) Cost of privacy protection

(iv) Cost due to lack of reputation

We assume that the relationship between utility and core factors and attributes is linear. The utility function can be established with RBC as follows:

$$U = \alpha Q[i, j] \times T[i, j] + \beta C_t - \gamma C_p - \delta C_r, \quad (1)$$

where α,β,γ, and δ are the weight coefficients of the core factors. When the drivers' revenue increases, the drivers' utility also increases. When passengers' payment increases, the passengers' utility decreases. There is a negative correlation between the passengers' travel utility and the travel cost. The weight coefficient value α in the driver utility function is bigger than that in the passenger utility function. There is a positive correlation between the drivers' travel utility and revenue.

We assume that all pick-up arrangements are completed within an acceptable time range. Therefore, there is little difference in the utility weight coefficient value β of time cost in the driver and passenger utility functions. Passengers pay more attention to privacy and reputation. Therefore, the weight coefficients γ and δ of privacy cost and reputation cost are bigger than those of drivers under the same cost.

3.3. Model Formulation. In a ride-sharing system, it is necessary to consider the acceptable time range, profit income,

and vacant seats for vehicle scheduling matching and transfer modes. In order to save travel cost and time, passengers may transfer.

Definition 1. Role R. In a ride-sharing system, a driver role is defined as $R := < n, I, Ac, Ap, No, Q_1 >$, where n is the identification of the driver role; $I := < M_{\text{in}}, M_{\text{out}} >$ denotes the message set processed by the driver roles of the ride-sharing system, where M_{in} expresses the incoming messages and M_{out} expresses the outgoing messages; Ac expresses a set of agents who are currently playing this role, that is, a set of passengers who are matched to the drivers; Ap expresses a set of agents who are potential to play this role, that is, a set of passengers that may be matched; No expresses a set of objects that can be obtained by the agent playing this role, mainly including traveling passengers on the way together; and $Q1$ expresses the minimum evaluation value required for the agent to play this role, i.e., travel revenue. In the ride-sharing system, role R represents a car driver who will be matched to passengers.

Definition 2. Agent A. In a ride-sharing system, a passenger agent is defined as $A := < n, Rc, Rp, Ng, Qr >$, where n is the identification of the passenger agent; Rc expresses a role that the agent is playing, that is, the driver of a car in which the passenger agent is riding; Rp expresses a set of roles that this agent may play in the future, i.e., the driver who will be matched due to transfers in the travel; Ng indicates the group number that this agent belongs to; and Qr expresses the evaluation value of this agent for each role in the set of roles. In the ride-sharing system, agent A expresses a passenger who sends out the message about the departure time, place, and destination.

Definition 3. L represents the passenger vector matched to the drivers, i.e., the number of passengers that each driver can be matched to within a certain time range.

Definition 4. The profit matrix Q of driver picking up passengers is a mxn matrix, where $Q[i, j]$ represents the revenue of the driver role $j(0 \le j < n)$ picking up the passenger agent i $(0 \le i < m)$.

Definition 5. The matching matrix T is an mxn matrix, where $T[i, j]$ represents whether the agent i is matched to the role j. $T[i, j] = 1$ indicates that the agent i is matched to the role j. $T[i, j] = 0$ indicates not. $T[2, 2] = 1$ represents the third passenger agent is matched to the third driver role.

Definition 6. When the number of passengers carried by the driver role R in the same time is less than or equal to the maximum passenger capacity of the car, the driver role R is operable, i.e., $\sum_{i=0}^{m} T[i, j] \le L[j]$.

Definition 7. Passengers may transfer many times from the departure to the destination. $L^a[i]$ $(0 \le i < m)$ indicates the maximum transfer times of passenger agent i. $L^a[i] = 0$ indicates that the passenger agent is not matched to a driver, and the sharing request fails.

Definition 8. $L[j] \in N(0 \le j < n)$ is a driver role range vector, where N is a natural number. In a ride-sharing travel system, it represents the maximum number of passengers carried by a car. For example, the maximum number of passengers carried by a 5-seat car at the same time is 4, i.e., $L[j] = 4$.

Definition 9. After determining the matching matrix T, the total revenue r is the sum of all drivers' revenue, i.e., $r = \sum_{j=0}^{n-1} \sum_{i=0}^{m-1} Q[i, j] \times T[i, j]$. The solution process of total income is to multiply the revenue matrix Q by the matching matrix T.

In the many drivers to many passenger mode, one driver can pick up more than one passenger. It is assumed that there is an intersection between ride time and distance. It is in a flexible and acceptable time range. Let m represent the number of passenger agents and n represent the number of driver roles. There are N seats in cars. Each passenger can transfer many times. The main objective is to maximize the total revenue r of all drivers picking up and transporting passengers. Let U_d represent drivers' utility and U_p represent passengers' utility. To encourage more drivers and passengers to participate in the ride-sharing system, the gap between driver utility and passenger utility should be within a reasonable range. A utility constraint is added to the following model.

$$\max r = \sum_{j=0}^{n-1} \sum_{i=0}^{m-1} Q[i, j] \times T[i, j], \tag{2}$$

$$\text{s.t.} T[i, j] \in \{0, 1\} (0 \le i < m, 0 \le j < n), \tag{3}$$

$$\sum_{i=0}^{m-1} T[i, j] = L[j] \le \text{N-1} (0 \le j < n), \tag{4}$$

$$\sum_{j=0}^{n-1} T[i, j] \le L^a[i] (0 \le i < m), \tag{5}$$

$$\frac{\left| U_d - U_p \right|}{U_d} \le \varepsilon, \tag{6}$$

where expression (3) is a 0-1 constraint, and it represents whether the passenger i is matched to the driver role j; (4) guarantees passengers are allocated according to the number of vacant seats; (5) indicates the maximum transfer times of the passenger i; and (6) indicates the utility constraint between drivers and passengers.

3.3.1. Case Analysis. It is assumed that 10 passengers send ride requests at the same time within a certain acceptable distance. There are 4 drivers available for dispatching. The cars all have 5 seats. Passengers do not transfer. According

Algorithm 1: Solve OMPR.
Inputs:
the number of driver roles n
the number of passenger agents m
the number of passengers carried by the n-dimensional role range vector L
the passenger transfer vector L_a
An m x n revenue matrix Q for the drivers to pick up and drop off passengers
Outputs: optimal matching matrix T and time t
begin
(1) While (time < passenger acceptance time && vacant seats < loading)
(2) {
(3) Call the Kuhn_Munkres algorithm to get the matching matrix
(4) }
(4) If (transfer times < L_a) {
(5) Return optimal matching matrix T, time t}
(6) End if
end

Algorithm 1: Kuhn_Munkres solves the optimal matching problem of ride-sharing (OMPR).

to the distance between drivers and each passenger, the revenue matrix R is as follows:

$$\begin{bmatrix} 1 & 1 & 3 & 6 \\ 4 & 3 & 1 & 5 \\ 2 & 1 & 10 & 5 \\ 1 & 3 & 6 & 5 \\ 8 & 2 & 1 & 7 \\ 10 & 2 & 1 & 5 \\ 9 & 4 & 6 & 1 \\ 7 & 1 & 10 & 2 \\ 8 & 4 & 1 & 3 \\ 1 & 8 & 3 & 9 \end{bmatrix}. \quad (7)$$

The matching matrix T may be as follows: (a)

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad (8)$$

Table 2: Experimental platform configuration.

Configuration name	Configuration description
CPU	AMD Ryzen 5 PRO 3500 U w/Radeon Vega Mobile Gfx 2.10 GHz
MM	8 GB
OS	Microsoft Windows 10 Home
Eclipse	Version: Eclipse Java Oxygen (4.7.3)
JDK	1.8.0_181

(b)

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (9)$$

According to the matching matrix (a), the total revenue is 75, and $L\,(\mathrm{a}) = [4, 2, 4, 0]$. With the matching matrix (b), the total revenue is 81, and $L\,(\mathrm{b}) = [4, 0, 3, 3]$. In the matching scheme (a), the 4th driver is not matched to a passenger, and the 2nd driver is not matched to a passenger in the matching scheme (b).

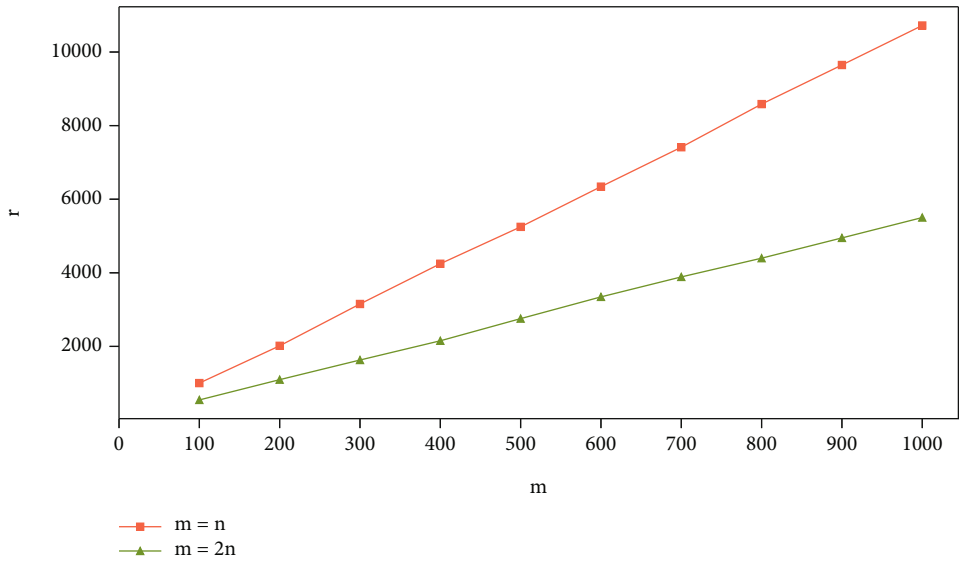


Figure 1: The total revenue with $m = n$ and $m = 2n$.

3.4. Algorithm Analysis. The time complexity of K-M algorithm [18, 19] is $O(m^3)$, which is better than the exhaustive search method. The above problem can be transformed into integer programming to solve the maximum matching matrix T. The algorithm is described as follows:

4. Simulation Experiments and Result Analysis

The experimental platform is shown in Table 2. To check the applicability and performance of the proposed model, the experiments are conducted. The values of the revenue matrix Q are produced randomly between 0 and 10. We let $m = \{100, 200, \cdots, 900, 100\}$, $n = m$, $m = 2n$, $L[j] \in [0, 1, 2]$, $\varepsilon = 0.2$.

According to the revenue matrix, simulate the scenarios with the same and different number of drivers and passengers as follows.

4.1. Nonutility Scenario. The utility of drivers and passengers is not considered. For each scale, the matching times in Kuhn-Munkres algorithm are collected. The total revenue r is shown with $m = n$ and $m = 2n$ in Figure 1. Tables 3 and 4 show the matching times with $m = n$ and $m = 2n$.

4.2. Utility Scenario. The utility of drivers and passengers is considered. In utility scenario, we let $\varepsilon = 0.2$. The utility of one-to-one and one-to-many matching when $m = n$ and $m = 2n$ is shown in Figure 2. The time of one-to-one and one-to-many matching when $m = n$ and $m = 2n$ is shown in Figure 3.

The experiments indicate that the greater the gap between the number of passengers and drivers, the less the total revenue and utility. When the number of drivers and passengers is the same, the utility of one-to-one matching between drivers and passengers is approximately equal to one-to-many matching. When the number of drivers and passengers is different, the utility of one-to-one matching is the least.

Table 3: The times (ms) for the assignment algorithm with $m = n$.

m	n	Largest	Smallest	Average
100	100	58.8402	34.7006	42.33467
200	200	211.0163	135.737	163.0133
300	300	595.0694	470.5191	518.6717
400	400	1694.838	1189.367	1451.6
500	500	5313.623	2639.78	3282.617
600	600	8314.075	5864.875	6565.413
700	700	12101.59	6828.488	9727.422
800	800	18487.95	12509.08	13758.43
900	900	25012.89	21341.05	23315.23
1000	1000	41122.11	34377.18	37197.4

Table 4: The times (ms) for the assignment algorithm with $m = 2n$.

m	n	Largest	Smallest	Average
100	50	35.7345	24.7931	28.50592
200	100	146.3957	102.3475	119.7406
300	150	473.8116	389.648	435.9143
400	200	1375.672	1157.329	1262.229
500	250	2927.459	2546.168	2786.129
600	300	6049.835	5204.061	5663.721
700	350	11628.82	9658.468	10809.14
800	400	21250.19	17858.75	19202.92
900	450	32658.36	28327.54	30718.17
1000	500	49614.14	42969.37	47163.84

In terms of the time spent on matching, Figure 3 shows that the time cost increases with the increase of the number of drivers and passengers participating in the ride-sharing

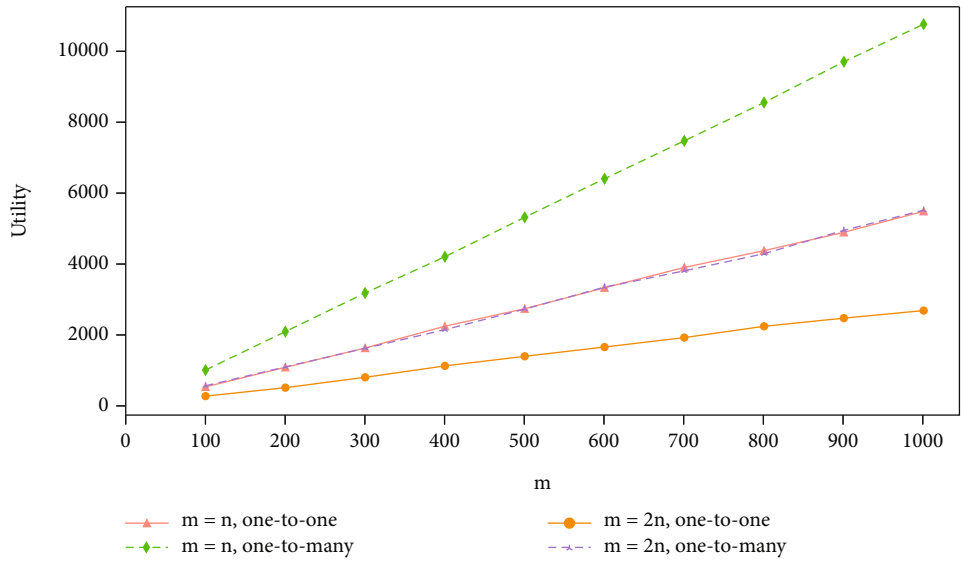


Figure 2: The utility of one-to-one and one-to-many matching with $m = n$ and $m = 2n$.

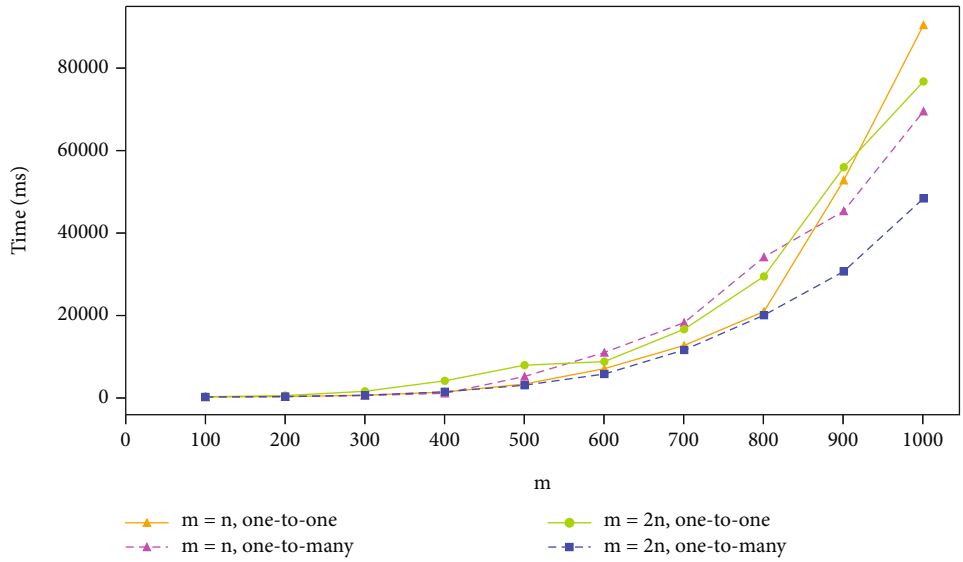


Figure 3: The utility of one-to-one and one-to-many matching with $m = n$ and $m = 2n$.

system. When m is bigger than a certain value ($m > 600$), time cost increases sharply in both matching methods. When the number of drivers and passengers is different, one-to-many matching takes the least time, and one-to-one matching takes more time. When the number of drivers and passengers is the same, the time cost of one-to-one matching increases sharply with $m > 800$. The results show that our solutions are practical.

5. Conclusions

The matching optimization is a general problem in a ride-sharing system. In this paper, under the constraints of vacant seats and passenger transferring, the matching problem in the ride-sharing system is formally modelled based on the E-CARGO model. First, the E-CARGO model and travel utility are introduced. Next, the proposed method is formalized. Then, algorithm analysis is provided. Finally, the Kuhn-Munkres algorithm is used to solve the optimal matching matrix. The analysis of time performance shows that the proposed method is practical.

In the future work, we will further study methods to solve matching optimization problems. To improve the matching rate and time performance, we will improve the algorithm to meet the real-time requirements of dynamic ride sharing matching. We plan to study the relationship between pricing and matching based on the E-CARGO model and utility theory. Besides, to attract more and more passengers and drivers to participate in the ride-sharing system, utility incentive mechanism is also the direction and content of our research in the future.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

We acknowledge support from the Educational Science Planning of Heilongjiang Province under Grant No. ZJB1421113 and No. GJB1421251. This work is supported by the National Key R&D Program of China under Grant No. 2020YFB1710200 and the National Natural Science Foundation of China under Grant No. 61872105 and No. 62072136.

References

[1] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[2] T. Liu, Y. Wang, Y. Li, X. Tong, L. Qi, and N. Jiang, "Privacy protection based on stream cipher for spatiotemporal data in IoT," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 7928–7940, 2020.

[3] L. Pan, Q. Cai, and Z. Fang, "Rebalancing dockless bike sharing systems," in *In Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, 2019.

[4] J. Cramer and A. B. Krueger, "Disruptive change in the taxi business: the case of uber," *The American Economic Review*, vol. 106, no. 5, pp. 177–182, 2016.

[5] M. Chen, W. Shen, P. Tang, and S. Zuo, "Dispatching through pricing: modeling ride-sharing and designing dynamic prices," in *Proc of Twenty-Eighth International Joint Conference on Artificial Intelligence IJCAI-19*, Macao, NJ: IEEE, 2019.

[6] Y. Wang, Y. Gao, Y. Li, and X. Tong, "A worker-selection incentive mechanism for optimizing platform-centric mobile crowdsourcing systems," *Computer Networks*, vol. 171, pp. 107144–107144, 2020.

[7] Z. Bian and X. Liu, "Mechanism design for first-mile ridesharing based on personalized requirements part I: Theoretical analysis in generalized scenarios," *Transportation Research Part B Methodological*, vol. 120, pp. 147–171, 2019.

[8] S. HU, M. M. Dessouky, and N. A. Uhan, "Cost-sharing mechanism design for ride-sharing," *Transportation Research Part B: Methodological*, vol. 150, no. 2, pp. 410–434, 2021.

[9] J. Wang and Y. Wang, "A two-stage incentive mechanism for rebalancing free-floating bike sharing systems: considering user preference," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 82, pp. 54–69, 2021.

[10] S. Majumder, A. P. Agalgaonkar, S. A. Khaparde, P. Ciufo, S. Perera, and S. V. Kulkarni, "Allocation of common-pool resources in an unmonitored open system," *IEEE Transactions on Power Systems*, vol. 34, no. 5, pp. 3912–3920, 2019.

[11] D. Liu, B. Huang, and H. Zhu, "Solving the tree-structured task allocation problem via group multirole assignment," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 41–55, 2020.

[12] Y. Wang, Z. Cai, Z.-H. Zhan, B. Zhao, X. Tong, and L. Qi, "Walrasian equilibrium-based multiobjective optimization for task allocation in mobile crowdsourcing," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 4, pp. 1033–1046, 2020.

[13] H. Ma, H. Zhu, K. Li, and W. Tang, "Collaborative optimization of service composition for data-intensive applications in a hybrid cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 5, pp. 1022–1035, 2019.

[14] H. Zhu and M. Zhou, "M-M role-transfer problems and their solutions," *IEEE Trans on Systems, Man and Cybernetics*, vol. 39, no. 2, pp. 448–459, 2009.

[15] H. Zhu, M. Hou, C. Wang, and M. C. Zhou, "An efficient outpatient scheduling approach," *IEEE Trans on Automation Science and Engineering*, vol. 9, no. 4, pp. 701–709, 2012.

[16] H. Zhu, "Role mechanisms in collaborative systems," *International Journal of Production Research*, vol. 44, no. 1, pp. 181–193, 2006.

[17] H. Zhu and M. Zhou, "Efficient role transfer based on Kuhn-Munkres algorithm," *IEEE Trans on Systems, Man and Cybernetics*, vol. 42, no. 2, pp. 491–496, 2012.

[18] J. Munkres, "Algorithms for the assignment and transportation problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 5, no. 1, pp. 32–38, 1957.

[19] H. W. Kuhn and B. Yaw, "The Hungarian method for the assignment problem," *Naval Research Logistics*, vol. 2, no. 1-2, pp. 83–97, 1955.

[20] J. Dutta and S. C. Pal, "A note on Hungarian method for solving assignment problem," *Journal of Information and Optimization Sciences*, vol. 36, no. 5, pp. 451–459, 2015.

[21] W. Gong, L. Qi, and Y. Xu, "Privacy-aware multidimensional mobile service quality prediction and recommendation in distributed fog environment," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 3075849, 8 pages, 2018.

[22] J. Long, W. Tan, W. Y. Szeto, and Y. Li, "Ride-sharing with travel time uncertainty," *ransportation Research Part B: Methodological*, vol. 118, pp. 143–171, 2018.

[23] A. Lopez, W. Jin, and A. Mohammad, "Security analysis for fixed-time traffic control systems," *Transportation Research Part B: Methodological*, vol. 139, pp. 473–495, 2020.

[24] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[25] D. J. Dailey, D. Loseff, and D. Meyers, "Seattle smart traveler: dynamic ridematching on the World Wide Web," *Transportation Research Part C–Emerging Technologies*, vol. 7, no. 1, pp. 17–32, 1999.

[26] J. Alonso-Mora, S. Samaranayake, A. Wallar, E. Frazzoli, and D. Rus, "On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 114, no. 3, pp. 462–467, 2017.

[27] R. W. Calvo and L. Fabio, "A distributed geographic information system for the daily car pooling problem," *Computers & Operations Research*, vol. 31, no. 13, pp. 2260–2278, 2004.

[28] X. Wang, H. Yang, and D. Zhu, "Driver-rider cost-sharing strategies and equilibria in a ridesharing program," *Transportation Science*, vol. 52, no. 4, pp. 868–881, 2018.

[29] N. A. H. Agatz, A. L. Erera, M. W. P. Savelsbergh, and X. Wang, "Dynamic ride-sharing: a simulation study in metro Atlanta," *Transportation Research Part B Methodological*, vol. 45, no. 9, pp. 1450–1464, 2011.

[30] A. Kleiner, B. Nebel, and V. Ziparo, "A mechanism for dynamic ride sharing based on parallel auctions," in *In: Proc. of the 22th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 266–272, Barcelona, Spain, 2011.

[31] H. Sun, H. Wang, and Z. Wan, "Model and analysis of labor supply for ride-sharing platforms in the presence of sample self-selection and endogeneity," *Transportation Research Part B: Methodological*, vol. 125, pp. 76–93, 2019.

[32] G. P. Cachon, K. Daniels, and R. Lobel, "The role of surge pricing on a service platform with self-scheduling capacity," *Social Science Research Network Electronic Journal*, vol. 19, no. 3, pp. 368–384, 2017.

[33] L. Wang and D. Suter, "Learning and matching of dynamic shape manifolds for human action recognition," *IEEE Transactions on Image Processing*, vol. 16, no. 6, pp. 1646–1661, 2007.

[34] Z. Peng, W. Shan, P. Jia, B. Yu, Y. Jiang, and B. Yao, "Stable ride-sharing matching for the commuters with payment design," *Transportation*, vol. 47, no. 1, pp. 1–21, 2020.

[35] E. Zkan, "Joint pricing and matching in ride-sharing systems - ScienceDirect," *European Journal of Operational Research*, vol. 287, no. 3, pp. 1149–1160, 2020.

[36] H. Zhu and M. Zhou, "Role-based collaboration and its kernel mechanisms," *IEEE Transactions on Systems, Man and Cybernetics. C*, vol. 36, no. 4, pp. 578–589, 2006.

[37] H. Zhu, M. Zhou Mengchu, and R. Alkins, "Group role assignment via a Kuhn–Munkres algorithm-based solution," *IEEE Transactions On System, Man and Cybernetics. A*, vol. 42, no. 3, pp. 739–750, 2012.

[38] H. Zhu, D. Liu, S. Zhang, S. Teng, and Y. Zhu, "Solving the group multirole assignment problem by improving the ILOG approach," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 12, pp. 3418–3424, 2017.

[39] H. Zhu, "Avoiding conflicts by group role assignment," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 4, pp. 535–547, 2016.

[40] H. Zhu, "Maximizing group performance while minimizing budget," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 2, pp. 633–645, 2020.

[41] H. Zhu, "Agent categorization with group role assignment with constraints and simulated annealing," *IEEE Transactions on Computational Social Systems*, vol. 99, pp. 1–12, 2020.

[42] C. Millan, "Theory of utility and consumer behaviour: a comprehensive review of concepts, properties and the most significant theorems," in *Contributions to Economics*, Pablo, 1999.

[43] A. Niromandfam, A. S. Yazdankhah, and R. Kazemzadeh, "Modeling demand response based on utility function considering wind profit maximization in the day-ahead market," *Journal of Cleaner Production*, vol. 251, p. 119317, 2019.

Hindawi
Wireless Communications and Mobile Computing
Volume 2023, Article ID 9832673, 1 page
https://doi.org/10.1155/2023/9832673

Retraction

Retracted: An Empirical Study on the Artificial Intelligence-Aided Quantitative Design of Art Images

Wireless Communications and Mobile Computing

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] W. Zhang and S. Tsai, "An Empirical Study on the Artificial Intelligence-Aided Quantitative Design of Art Images," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 8036323, 11 pages, 2021.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8036323, 11 pages
https://doi.org/10.1155/2021/8036323

Research Article

An Empirical Study on the Artificial Intelligence-Aided Quantitative Design of Art Images

Wen Zhang[1] **and Sang-Bing Tsai**[2]

[1]*College of Design, Chongqing Industry Polytechnic College, Chongqing 401120, China*
[2]*Regional Green Economy Development Research Center, School of Business, WUYI University, China*

Correspondence should be addressed to Wen Zhang; zhwen-09@163.com and Sang-Bing Tsai; sangbing@hotmail.com

Received 1 September 2021; Revised 27 September 2021; Accepted 13 October 2021; Published 28 October 2021

Academic Editor: Yingjie Wang

This paper presents an indepth analysis and research on the quantitative design of fine art images through artificial intelligence algorithms. A CycleGAN-based network model for automatic generation of sketches of fine art images is constructed to extract the edge and contour features of fine art images. The network uses 512×1024 high-resolution art images as input and Pitchman as a discriminator. To further enhance the sketch generation effect, a bilateral filtering algorithm is added to the generator model for noise reduction, and then a K-means algorithm is used for color quantization to solve the problem of cluttered lines in the generated sketches. The experimental results show that the network model can effectively realize the automatic generation of art image sketches and can retain the detailed part of the costume information well. A rendering platform is built to realize the application of art image generation algorithms and coloring algorithms. The platform integrates the functions of image preprocessing, sketch generation, and sketch coloring, demonstrates the results of the main research content of this paper, and finally increases the interest of the system through the rendering function of the art image grid, which further improves the practicality of the platform.

1. Introduction

With the development of image processing technology, various types of image processing have become a necessary part of production and life. The image processing process relies on high-quality raw images, and if the acquired images themselves are of low quality, it will affect the effect of the processing and will not be able to play the value of the images themselves. However, in the actual technical application, the image acquisition is easily disturbed by a series of factors such as weather conditions (e.g., fog and rain), the poor performance of the acquisition equipment, and insufficient lighting conditions [1]. As a result, the quality of directly acquired images is difficult to meet the usage standards and problems such as inconspicuous details, color distortion, and excessive noise occur, which affect the process of image feature extraction and target recognition and ultimately reduce the efficiency of image processing [2]. Different filters activate different content, showing different activation values, and finally visualized as different images. From the perspective of different convolutional layers, the shallow convolutional layer has a function like edge detection and can basically extract the texture information of the image. To address these problems, image enhancement technology can play an important role in improving the display of images and making them more accurate in conveying information. Image enhancement refers to a series of processing according to certain specific requirements for the captured image with poor quality, and the algorithm is used to enhance the information expressed by the image subject and suppress the disturbing information in the image to improve the image quality and provide a better data source for the later image processing [3]. Therefore, image enhancement plays a key role in the image data preprocessing stage and occupies an irreplaceable position in the digital image processing process.

In recent years, along with the gradual improvement of hardware computing power, the application scenarios of image enhancement algorithms have been further expanded,

the human measurement of image enhancement algorithms has been improved, and such algorithms have been applied to medical care, transportation, education, and agricultural production. In the field of face recognition, image enhancement provides higher quality recognition images for later recognition algorithms, which facilitates the recognition algorithms to play their best role; in the smart city application scenario, video surveillance images processed by image enhancement algorithms can provide the important information needed by observers more intuitively; in the direction of medical imaging, image enhancement algorithms can increase the accuracy of medical equipment [4]. In the direction of medical imaging, image enhancement algorithms can increase the accuracy of medical equipment to reflect the patient's condition and facilitate medical workers to make more accurate judgments about the patient's condition. For the inheritance and protection of art images, on the one hand, we need to try to rescue the existing representative art images; on the other hand, we also need to actively guide the reform of art images, based on trying to maintain their national traditions and cultural characteristics, integrate some modern colors so that they can adapt to the needs of society so that more people can accept and like art images, and provide a new opportunity for the development of art images [5]. The sketches of art images are the basis to produce art images, and the various sketches of art images provide more choices for national costumes so that you can select the art images that meet your needs. As the art image production starts from the art image sketch, then it goes through a series of complicated processing and finally the coloring. This process is very time-consuming and labor-intensive, which is why there are only a few art images in circulation, and this brings great inconvenience to the inheritance and development of art images.

With the continuous development of artificial intelligence, there are increased collections and collations about art image sketches, which are a visual expression form composed of a few lines and can be executed quickly. For example, a certain filter may capture the vertical texture existing in the image, and a certain filter in the same layer may capture a certain color area, for example, a certain filter in the first convolutional layer may capture the eye feature is activated to highlight it. Using modern technology, the art image sketches are rendered stably and efficiently, and the art image's style is combined with modern people's aesthetics to innovate and apply it to art image design, and this kind of art image with the modernized clothing with ethnic characteristics will contain the beautiful symbolic meaning of minority elements and at the same time meet the aesthetics of most people, which provides a new direction for the inheritance and development of art image culture. Most of the traditional coloring methods require manual interaction and professional equipment processing, which is very difficult and requires manual correction if the coloring effect is not good; because of the manual interaction, the coloring of each image requires human participation, which makes it difficult to apply to the mass production industry [6]. Recently, there are many coloring algorithms based on deep learning, especially automatic sketch coloring, which can be roughly divided into guided user coloring and unsupervised coloring.

2. Current Status of Research

Image style migration methods based on image iteration can generate excellent stylized images, but the problems of slow speed and high resource usage of the whole style migration process make this type of method still have limitations. The generative neural methods based on model iteration (fast neural style migration) solve the above problems to some extent by improving the inference speed and reducing the computational cost [7]. The key idea of model iteration-based methods is to optimize the network model by iteratively updating the model instead of iteratively updating the image pixels using the gradient descent method [6]. Depending on the model iteration method, the main approaches based on model iteration can be grouped into generative model-based and image-reconfiguration decoder-based approaches. Wang et al. were the first to propose an iterative optimization of the generative model for image style migration, and their related work provides a good idea for improving the efficiency of image style migration, which builds on the algorithm [8]. Gündüz et al. by using a perceptual loss function is used to train a generative model for a particular style [9]. In addition, the work of Schaller et al. conducted using a similar network architecture and experimentally showed that the use of instance normalization during the training of the generative model can significantly improve the quality of the generated images [10]. Schaller et al. proposed a multimodal convolutional neural network that considers the feature representation of the color and luminance channels to perform stylization in a multiscale hierarchical manner, effectively solving the texture scale adaptation problem and produced considerable image generation results on high-resolution images [11].

Similarly, another "pre-processing" method to eliminate moiré is to implement pixel merging. The pixel merging technique combines several pixels into a single pixel. Combining a group of 2×2 pixels into a single pixel is equivalent to enlarging the RGB array on the sensor, which allows for signal dispersion and suppresses moiré. Similar to OLPF, the reduction in the number of pixels leads to a significant loss of image information, and although moiré suppression and noise reduction are achieved, this is at the cost of image sharpness. Second, another approach from pixels is to achieve a pixel density increase [12]. An increase in pixel density means that the sensor holds more information, and when the pixel density is increased to a level much greater than the density of the scene being captured, there is no high-frequency interference, thus suppressing moiré. However, the limitation of increasing pixel density is that it is technically difficult to achieve: when the size of the imaging components remains the same, the pixel density increases, the area occupied by each pixel decreases, and the resolution increases but the sensitivity and error tolerance decreases; in addition, when the number of pixels increases, the requirements for the processor also increase greatly, and the processor has great difficulty in processing in the face of the huge amount of data read, so by increasing pixel density [13]. This

technique is too expensive to achieve moiré suppression. Soleymanian et al. proposed a method for moiré suppression in 3D stereo imaging based on tilted lens arrays [14]. By changing the tilt angle between the lens array and the display panel, the best angle to reduce the appearance of moiré is selected. However, this method is time-consuming and may not achieve the desired results; so, the elimination of moiré by "post-processing" becomes the best choice [15].

An artificial feature extraction method for image classification is to design features artificially for image classification by observing the scene and analyzing the data, depending on the image data and for a specific scene. The image size perspective, which is changed by stretching and rotating, is not sensitive to the color feature and is easy to obtain by simple computation and is often used in applications such as image classification retrieval and search. The color feature color allows for an efficient and simple description of the color in an image and can reduce the loss in the image quantization process. A nine-dimensional feature vector is finally generated by calculating the mean and standard deviation of the three-color channels and the skewness. The color histogram describes the proportion of different colors in the same image, which has the disadvantage that it is easy to ignore the color information in the context of the image. Color correlation map features not only count the probability of occurrence of different colors in a single image but also reflect the correlation between pairs of these colors in space. Image texture features are portrayed by quantifying the pixels of an image to describe the distribution of grayscale and luminance of regions of interest in the image. Cascading local binary pattern texture features by structural analysis and gradient histogram texture features by statistical analysis can effectively improve image classification.

3. Artificial Intelligence-Aided Quantitative Design of Art Images

3.1. Artificial Intelligence-Aided Design. There are various colorful colors in nature, and how to display them to electronic devices requires a mathematical modeling method to achieve this, which is called color space or color pattern within the industry. Edge detection and contour extraction are a very tricky job. The texture itself is a very weak edge distribution pattern, and poorly handled detail parts are easily masked by overly strong image lines; so, edge detection has a very important position in digital image technology, and the main role of edge detection is to distinguish the detection target from the background. The sketches are highly abstract, sparse, and other characteristics, which leads to different people when describing the same art image, and the clothing sketches produced are often different. Therefore, there are many difficulties in the current collection and arrangement of art image sketches. It also greatly influences the research related to the understanding and analysis of fine art images. There are three main categories of commonly used image edge extraction methods: classical edge extraction methods, which are based on fixed mathematical operations in the image localization, such as differentiation and fitting method. Global extraction methods are based on the principle of energy minimization and use rigorous mathematical methods to analyze the edge extraction problem, based on a one-dimensional value function to extract and remove the image edges from a global optimization perspective, e.g., relaxation method and neural network analysis. The newly developed image edge extraction methods in recent years are represented by wavelet transform and mathematical morphology, especially the wavelet transform method makes full use of the multiscale features of the image to achieve the extraction of image edge contours, which is widely used in many research topics [16]. With the further development and research of deep learning technology, the commonly used image edge extraction algorithms today are holistic nested edge detection (HED), accurate edge detector with convolutional features (RCF), and bidirectional cascade network structure (BDCN).

In multiscale feature learning, the commonly used methods are the image pyramid method and the deep neural network method, but both methods have many drawbacks, such as repeated operations in feature learning and many parameters, which can lead to long inference time. The BDCN method is designed with a lightweight network structure for edge detection based on the convolution part of VGG and achieves good results; each layer in the BDCN corresponds to a scale of features, which solves the problem that the shallow layers can only focus on local patterns because of the perceptual field, while the higher layers can notice the target level information; so, it is unreasonable to use the same label for the last layer and the middle layers for supervision. To better achieve edge feature extraction, a module more like ASPP also propose, which uses a novel bidirectional loss supervision approach to let each intermediate layer learn its appropriate scale, is shown in Figure 1.

Convolutional neural networks were first successful in the field of image recognition. Machine recognition of images is not able to recognize complex images at once, but the complete image is partitioned into several small parts; then, the features of each small part are extracted separately (recognizing each small part), then the features of several small parts are aggregated together, and machine recognition of images can be achieved. The principle of image recognition is that the input layer reads in the image that has been preprocessed image, the convolution layer initially extracts features, the pooling layer extracts the main features, and the fully connected layer aggregates the features of each part and finally generates a classifier for predictive recognition. The principle of image style migration is to input features and then output images with corresponding features. Therefore, the powerful feature extraction capability of convolutional neural networks can be used to perform research work related to image style migration. This allows the cloud to largely solve the computing requirements of applications, but in some application scenarios, computing must be performed locally.

By convolutional neural network for feature extraction of content image and style image and visualization of the convolutional layers of the network, it can be found that the activation state of each layer of neurons in the convolutional neural network corresponds to a specific kind of

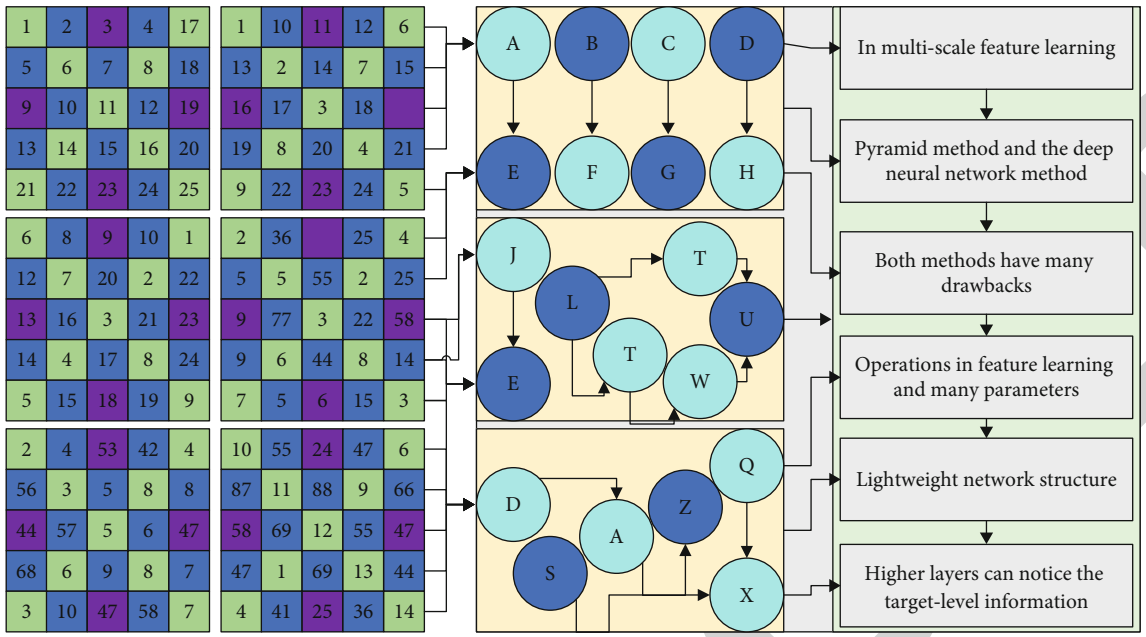


Figure 1: Artificial intelligence-assisted algorithm.

information, and different filters are activated for different contents, which exhibit different activation values and finally visualized as different images. From the perspective of different convolutional layers, the shallow convolutional layer plays a similar function to edge detection and can extract the texture information of the image [17]. The shape of the kitten's eyes can be observed; the deep convolutional layer can extract the actual content and spatial information (complex structure, object class) and can only observe the object class in the image as a kitten, and some detailed features of the kitten are completely lost, mainly due to the deepening with the number of convolutional layers. Certain convolutional kernels do not extract the features they want in the input image; so, the deeper convolutional layers extract progressively less detailed information about the image, which leads to increasingly abstract features for the visualization display. For the same convolutional layer, different filters of the convolutional layer will capture different features, e.g., a certain filter may capture the vertical texture present in the image, while a certain filter of the same layer may capture a certain color region, e.g., a certain filter of the first convolutional layer will activate the eye feature in the image and thus highlight it.

The expression of the exponential transformation function is shown below.

$$g(x,y) = af(x,y)^{\lambda}, \tag{1}$$

where a and λ are used as parameters to adjust the shape of the curve of the exponential transform function, respectively, to achieve different image enhancement effects. This nonlinear transform can be used to extend the high gray areas of the image and have a better compression effect on the low gray areas. The spatial domain filtering method essentially utilizes a template method for processing, which can remove noise or redundant details from digital images. Spatial domain filtering acts the specified neighborhood pixel values with the corresponding neighborhood subimage pixel values having the same dimension, and the resulting new pixel values replace the original pixel values.

$$g(x,y) = \sum_{s=-a}^{a}\sum_{t=-b}^{b} w(s,t)f(x-s,y-t), \tag{2}$$

where $w(s,t)$ is the filter coefficient representing the weight of the template at the pixel point, and $f(x-s,y-t)$ is the pixel value of the input image. The simplified form of the spatial domain filtering is

$$R = w_1 z_1 - w_2 z_2 - \cdots - w_m z_m. \tag{3}$$

In the above equation, w is the filter coefficient, z is the image gray value corresponding to this filter coefficient, and Mn is the total number of pixel points contained in the filter. The image pixels processed by the geometric mean filter are obtained mainly based on the inner product power of the pixels in the template window. The geometric mean filter can better filter the Gaussian noise and at the same time has a better effect on preserving the edge information of the image. However, due to the special composition of the algorithm, this filter is more sensitive to 0 pixels. When a pixel in the filter window has a gray value of 0, then the output of the corresponding geometric filter is 0, which will have a significant impact on the image denoising effect.

$$g(x,y) = \frac{\mathrm{nm}^2}{\sum_{(x,y)\in S_{xy}} (1/f(x,y))}. \tag{4}$$

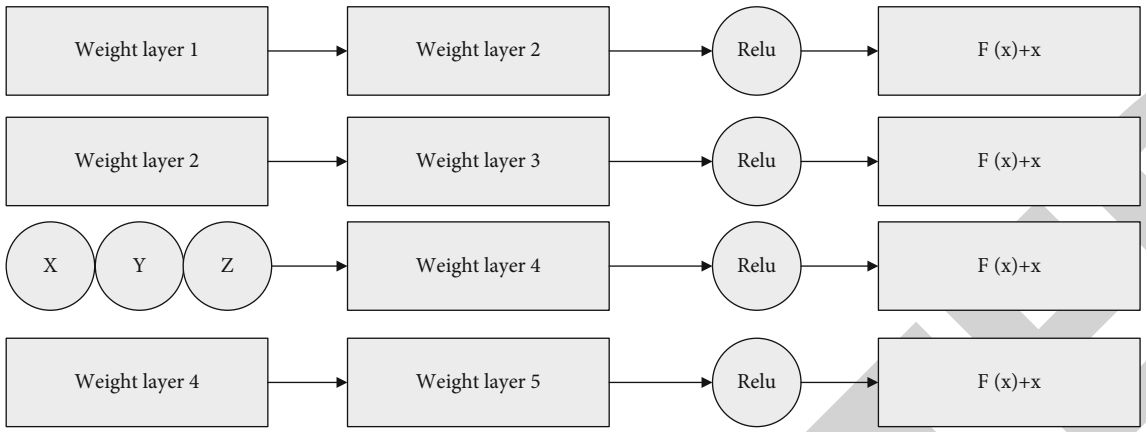


Figure 2: Resnet block.

The filter is more effective in processing salt grain noise and can be applied to the processing of Gaussian noise as well, but it is difficult to achieve the expected results in pepper noise processing.

$$g(x,y) = \frac{\sum_{(x,y)\in S_{xy}} \left(f(x,y)^{Q} / f(x,y) \right)}{\sum_{(x,y)\in S_{xy}} (1/f(x,y))}. \quad (5)$$

The first method, also known as the direct method, has the main idea of optimizing the objective function while considering all data categories. According to the traditional support vector machine binary classification model, multiple classification planes constructed for multiple categories, and then the objective function is optimized by the parameters of each classification plane, which is transformed into a quadratic programming problem that can solve multiple classification problems at once. To a certain extent, the improved method overcomes the problem of high storage and calculation cost of image style transfer model and huge computing resource consumption. It can only be used on a limited platform and cannot be transplanted to the mobile terminal at all. Although this algorithm seems to be simple and convenient, the complexity of its objective function is high, and because its computational process involves many variables, it makes the computational cost and complexity not only high but also difficult to implement the process. For linearly divisible samples, only a linear function needs to be designed as a segmentation hyperplane, and the familiar linearly divisible support vector machine can divide the data correctly and with maximum interval. In the sample space, the partitioning hyperplane can be done by equation (6).

$$W^{T}x - b = 0. \quad (6)$$

We know that GAN networks extract features by coding and decoding; however, for neural networks, as the number of layers increases, more features are extracted from each layer. Ideally, the network would need to add more layers to achieve good results, but in practice, as the number of layers increases, the neural network model goes backward, as shown in Figure 2.

Where x is the input information, the input has been extracting features in the process of convolution, and after linear changes into $F(x)$; as the network continues to deepen, to ensure that the process of extracting features will not lead to information loss due to problems such as gradient disappearance in the process of increasing the network, the input x can be added to the output, and the learned features $F(x)$ together as the next layer of calculation input; so, the final output $y = F(x) - x$, and then the residual $y = F(x) + x$; when the residual is 0, the output result $y = x$, by the way of residual at least can ensure that the overall of the network will not decline.

Cause distortion and degradation of image quality: the quality of the image directly affects the effect of the image application field. For example, in face recognition, the image quality directly affects the accuracy of the recognition result. Sketches of art images, which consist of clean white backgrounds and black lines, contain many patterns and motifs and are an important basis to produce ethnic costumes, as well as an important reference in the field of fashion design when producing works with ethnic characteristics. However, due to the diversity and richness of art image styles and the high abstraction and sparseness of sketches, the sketches produced by different people when describing the same art image are often different; so, there are many difficulties in collecting and organizing art image sketches, which also greatly affect the research related to art image understanding and analysis. How to generate high-quality art image sketches efficiently has become the focus of research. Currently, deep learning based on the generation of cartoon and caricature images has achieved good results, but there are still many problems in the application of art image sketch generation direction, such as the generation of clothing sketch texture is not clear, the generation of sketch appears messy lines, and in the process of generating clothing sketch detail information is lost.

3.2. Quantitative Design of Fine Art Images. The trend of deep learning development is that the network structure is getting deeper and deeper; thus, causing deep learning

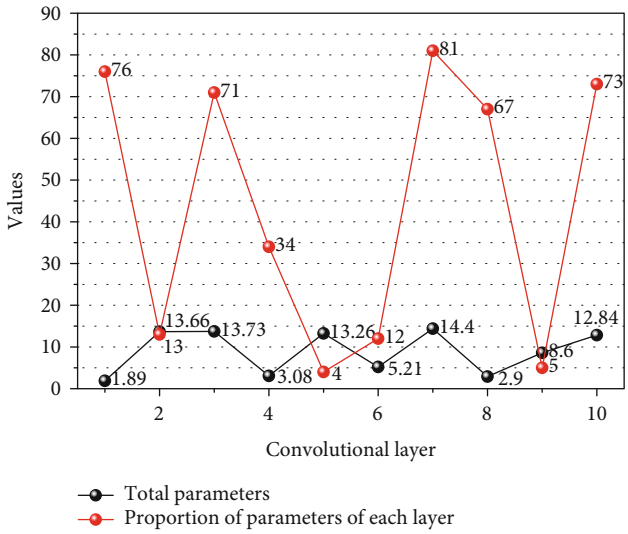


Figure 3: Number of parameters in each layer.

models to become larger and larger, and the resources used for computation are also getting larger and larger. Even though most applications are placed in the cloud and use GPUs for computation, which makes the cloud largely solve the computational needs of applications, in some application scenarios, the computation must rely on local, for example, the hottest driverless, for example, the hottest driverless applications require a timely response to the external environment, including pedestrians outside the car and traffic lights. If the quality of the collected image itself is low, it will inevitably affect the processing effect, and the value of the image itself cannot be used. However, in actual technical applications, image collection is susceptible to interference from a series of factors, such as weather conditions (such as fog and rain), poor performance of the collection equipment, and insufficient lighting conditions. If the network is delayed or fails, it will have an incalculable impact [18]. For the image style migration system studied in this paper, from the demand of real-time response, it is possible to protect the privacy of users by running the processing style model locally without transferring related images and model parameters to the server.

Although we have used the fast image style migration method to improve the performance of the network and achieved amazing migration results, the problem of the network parameters of the style model ensues. The size of a single style model trained by the fast image style migration algorithm will be about 20 M, which can only be deployed in small amounts on mobile, and as the number of style models deployed increases, it will pose a huge challenge to the power, storage capacity, and computing power of mobile devices. As the number of style models deployed increases, it will pose a huge challenge to the power, storage capacity, and computing power of mobile devices. As shown in Figure 3, by analyzing the corresponding style models trained, it is found that the parameters of the whole network model are mainly concentrated in the residual layer, the parameters of the single-layer residual layer account for 17.6% of the parameters of the whole network model structure, the parameters of the five-layer residual network account for 88% of the parameters of the whole network model structure, and the calculation method of the original residual layer cell structure (convolutional method) causes the huge parameters of the residual layer. The large parameters not only affect the speed of inference but also occupy the storage resources of the mobile terminal. See Figure 3.

To solve the above problems, this paper improves the cell structure of the original residual network by designing more efficient network computation. It is shown that the improved method can overcome the problem that the image style migration model is expensive to store and compute, consumes huge computational resources, can only be used in limited platforms, and cannot be ported too mobile at all.

Image is a kind of media to express information, is the main source of human to obtain and exchange information, compared with other information is more intuitive, concrete, and vivid, contains rich information, and is widely used in many fields. In the process of image acquisition, processing, and transmission, it is easy to cause distortion and degradation of image quality due to unscientific processing methods, noise pollution, transmission media, and imperfect hardware equipment [19]. Image enhancement refers to a series of processing for the collected images of poor quality in accordance with certain specific needs. The algorithm is used to enhance the information expressed by the main body of the image, suppress the interference information in the image, improve the image quality at the same time, and provide a better data source for later image processing. For example, in face recognition, image quality directly affects the accuracy of recognition results; in traffic monitoring, high-quality images will play a good role in the analysis of traffic accidents; therefore, the evaluation of image quality is very important. In this paper, we successfully achieve the compression goal of the style migration model, and the evaluation of the image quality of the compressed model is a very important index to measure the performance of the compressed model. The main content of this section is to use an objective and accurate image quality evaluation method to evaluate the image quality of the migrated images after compressing the model to verify the loss of the performance of the compressed model.

In the field of digital image processing research, the gradient of an image can calculate to obtain the edge information of an image. The process of solving the gradient information of an image is generally a derivative operation of the image, and the result of the operation reflects the degree and direction of grayscale change of the image pixel points so that the edge of the image can be judged more accurately. When the gradient of an image is calculated by the first-order difference method, the pixel point with a larger gradient value is in the edge region of the image, and the pixel point with a smaller gradient value is in the smooth region of the image, as shown in Figure 4.

Based on keeping its own national traditions and cultural characteristics as much as possible, it incorporates some modern colors so that it can adapt to the needs of society. Noise reduction processing is an important preprocessing

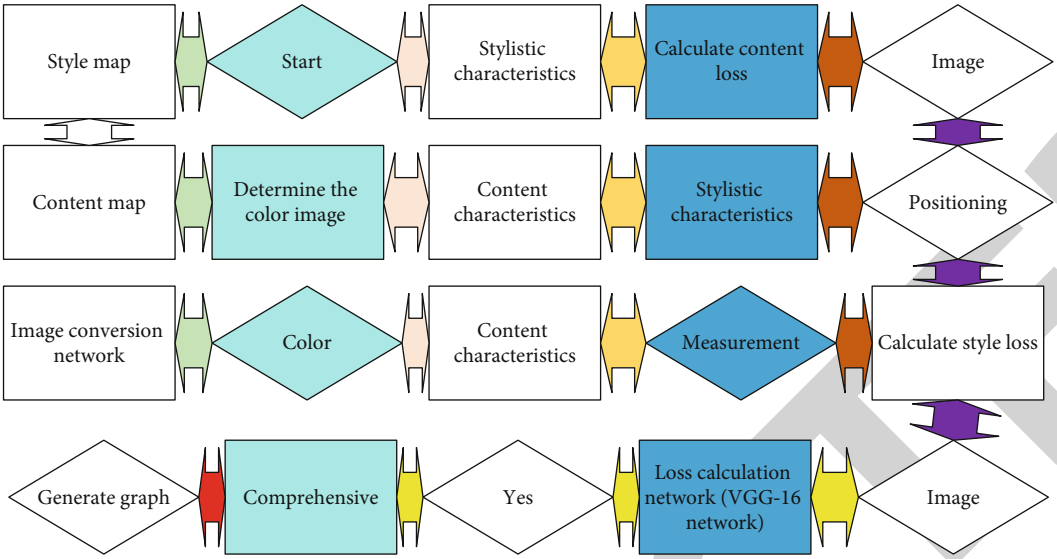


Figure 4: Image quantization design steps.

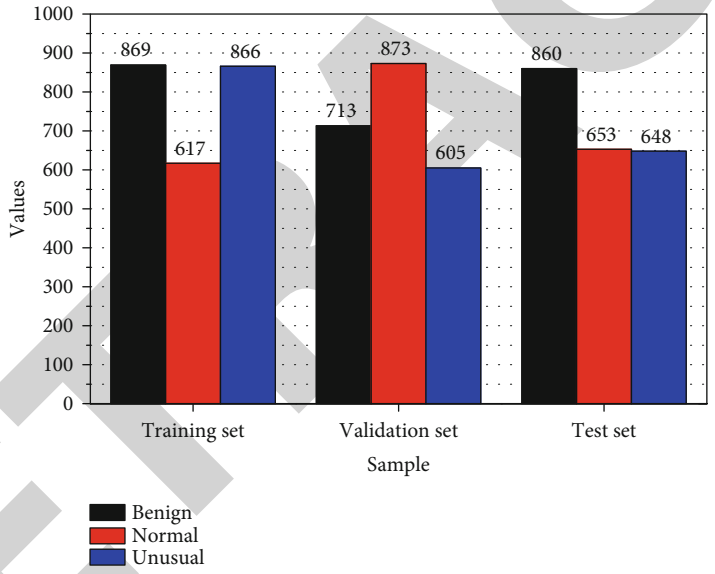


Figure 5: Experimental data.

tool in the field of computer vision. Noise interference and other experimental external environments and the imaging qualities of the camera can sometimes seriously affect the digitization and transmission of images, resulting in serious degradation of image quality. Therefore, to obtain high-quality images that can maintain the complete original image information while removing useless information, image noise reduction processing is particularly important [20]. In recent years, how noise reduction algorithms can have both advantages has become the focus of research [21–23].

In this paper, we construct a CycleGAN network-based sketch generation model for ethnic costumes. In this paper, we construct a sketch generation model based on the CycleGAN network, which can realize the conversion from color images to sketches without large-scale high-resolution datasets. The model in this paper solves the requirement of pairwise data sets for general deep learning methods and does not restrict pairwise data for the images of the input model; so, we choose high-resolution minority dress images and other images with high resolution for training, which can improve the training effect of CycleGAN network model and effectively realize the generation of minority dress sketches. See Figure 4.

4. Analysis of Results

4.1. Artificial Intelligence Algorithm Performance Results. To make the experimental configuration more realistic and challenging, we divided the region of interest data of mammography images into 60% training set, 20% validation set, and 20% test set, as shown in Figure 5, which consisted of 6971, 2323, and 2268 images as training data set, validation

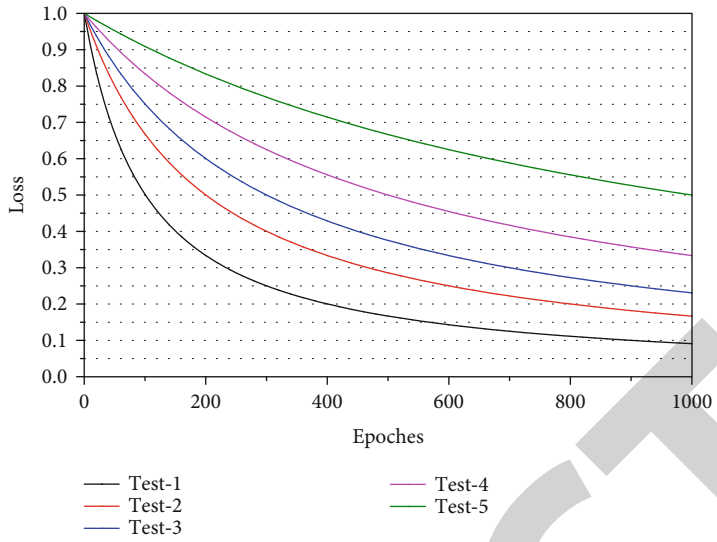


Figure 6: Algorithm loss results.

data set, and test data set, respectively. In these datasets, three categories of 1119 benign, 1266 malignant, and 9177 positives' normal regions of interest were included. A total of 677 benign, 764 malignant, and 5530 normal regions of interest randomly selected as the training dataset; 225 benign, 254 malignant, and 1844 normal regions of interest were selected as the validation dataset: the remaining 217 benign, 248 malignant, and 1803 normal regions of interest were used as the test dataset, making up a total of 11562 mammogram image data for this paper. Since the production of fine art image starts with the fine art image sketch, then it goes through a series of complicated processing and finally is the coloring. In this process, the coloring of the fine art image sketch is very time-consuming and labor-intensive. A total of 11562 mammogram images constitute the mammogram image database of this paper. The training set is used to train the model, then the validation set is used to validate the model, and the model is continuously adjusted according to the validation results of the validation set to select the best model. See Figure 5.

The horizontal and from coordinates in Figure 6 indicate the number of iterations of model training and the loss of model training, respectively, from which the change of classification loss with and without migration learning can be observed, and the classification loss of migration learning becomes smaller and smaller with the increase of the number of iterations. When the number of training is 100000, i.e., Figure 6, the classification loss converges around 0.4, while the classification loss converges around 1 and 0.6 when the number of training is 1000 and 10000, respectively, in Figure 6. If the coloring effect is not good, it needs to be corrected manually; because it is manual interaction, the coloring of each image requires human participation, which makes it difficult to apply to mass production industries. The red curve indicating the classification loss by migration learning is significantly smaller than the blue curve without migration learning, and the gap between them becomes larger as the number of training iterations increases. The difference between the loss at 1000 and 10000 iterations is not less than 0.4, showing that the performance of migration learning becomes better than that without migration learning as the number of training iterations increases. See Figure 6.

From the figure, different quantization results can be obtained after different processing of the incoming parameter values. The quantization result of the image is close to the linear transformation, and the dark area of the image gets a better enhancement effect, while the bright area of the image has a lower contrast, which is difficult to have a better visual effect. By increasing the value of parameter, a, the pixels in the middle region of the image are enhanced, and the enhancement effect is obvious in the region with higher brightness, while the detail information is missing more in the region with lower brightness. Poorly processed details are easily concealed by too strong image lines. Therefore, edge detection has a very important position in digital image technology. The main function of edge detection is to distinguish the detection target from the background. Therefore, this paper adopts the control parameter into the value to adjust the quantization effect of the reflection image, obtains the optimal solutions of the light and dark regions, distinguishes the light and dark regions of the reflection image with the help of the improved multi-threshold OTSU algorithm, and finally fuses the optimal solutions of the light and dark regions to obtain the resultant image.

In the evaluation of the experimental results, two main types of methods are used. The first type is the observation of the image enhancement result image by human eyes, which can get the subjective visual evaluation of the image enhancement effect; although this type of method has a certain subjectivity for the resulting image, one of the criteria to judge whether it can play its value is whether the necessary information can be obtained by human eyes. At the same time, image enhancement algorithms are mostly born to meet human visual observation, and subjective evaluation of images is one of the indispensable components of image

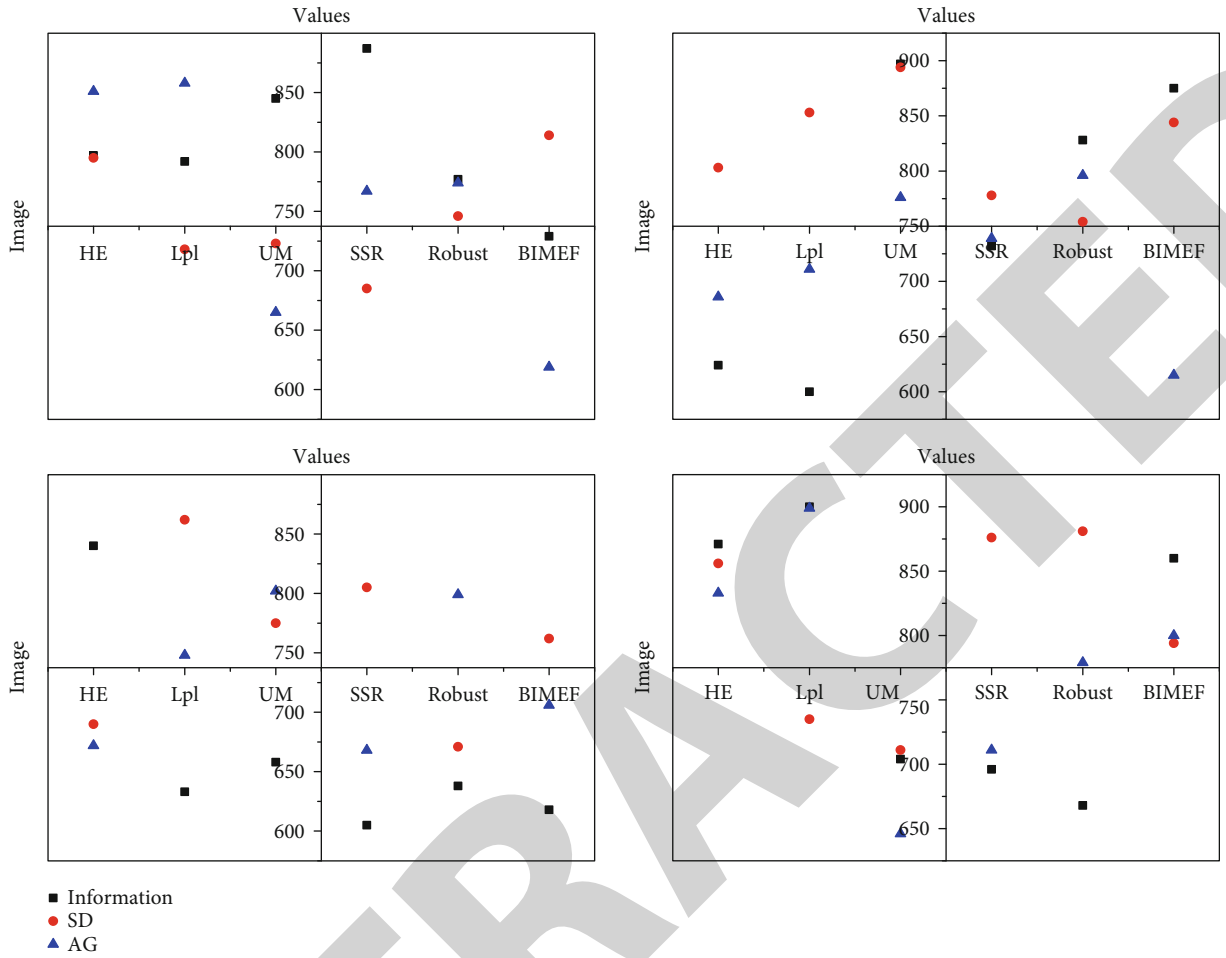


Figure 7: Comparison of image enhancement results.

evaluation criteria; the second category is to evaluate and judge the image enhancement results using objective values, i.e., calculate the relevant indexes by quantified methods, evaluate them according to the size of the index data, and complete the improvement of the algorithm according to the evaluation results.

4.2. Quantitative Design Results for Art Images. To objectively evaluate the application effect of various algorithms, several evaluation indexes were used for quantitative analysis, mainly the average gradient, information entropy, and standard deviation indexes, and the specific results are shown in Figure 7.

With the analysis of Figure 7, it can be seen that the four contrast algorithms have significant enhancement effects on haze images, among which, the enhancement effects of HE, SSR, and the algorithm in this paper are more significant for the direction of image contrast enhancement; for the direction of image edge information retention, the effects of Lpl, SSR, and the algorithm in this paper are more obvious; in terms of image contrast, HE algorithm and the algorithm in this paper perform better; in terms of image sharpness, HE, SSR, and the algorithm in this paper have the best results. In terms of image contrast, HE, SSR, and the algorithm in this paper have better performance; in terms of image sharpness, HE, SSR, and the algorithm in this paper have significant effects. In the process of image acquisition, under some special acquisition conditions, due to the limitation of the acquisition equipment, the acquired images may have low contrast, which has a certain degree of impact on human eye observation and subsequent image processing. The specific manifestation is that the overall image is grayish, the distinction between dark and bright areas of the image is not obvious, and the edge information is weakened.

Firstly, the filtering methods, whether BF, LLF, or SF, all have a large amount of moiré structure, and neither the moiré of colored stripes nor the moiré of vertical stripes is well eliminated; although the SDGF algorithm uses guided filtering, the moiré of colored stripes still has a large amount of residual, and secondly, the hue of the image is changed, and there is an over smoothing phenomenon in the recovered image. In addition, the recovered image also has the ringing effect; the PS software smoothest out the image details and edge information, resulting in an oversmoothed image, while the moiré component remains; the recovered image of the proposed algorithm not only eliminates each type of moiré but also retains the image details. Compared with other algorithms, the recovered image of the proposed algorithm is optimal in terms of visual quality.

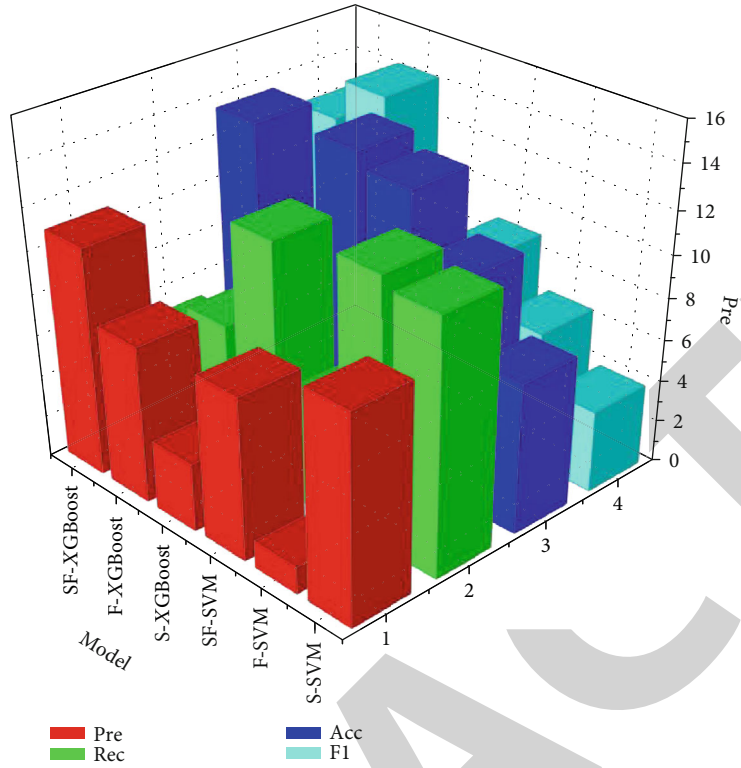


Figure 8: Comparison of tests with different feature combination strategies.

The same is true for the image similarity index SSIM. The higher the similarity, the closer the generated image is to the real image. The above results show that the output of the image generated by using Pix2Pix (generator using Unet128) is roughly close to the original image and has the effect of clear details, but its problem is that it adds a lot of extra unnecessary details, which is far from the original real image in terms of details. Since the residual blocks can better preserve the detail information of the image, the output result using Pix2Pix (Resnet_9 blocks) is more satisfactory, which not only improves the details but also makes the target image clearer, and the generated image is closer to the real image, achieving a good coloring effect, as shown in Figure 8.

It allows users to go for image edge contour extraction and sketching quickly and easily for coloring and adds some diversified functions in the design process, such as ethnic style rendering module, which allows users to realize various types of image stylization according to their needs, bringing more interesting experiences to users and making this platform realize end-to-end operation in a real sense. The convolutional layer initially extracts features, the pooling layer extracts main features, and the fully connected layer summarizes the features of each part and finally generates a classifier for predictive recognition. Various salient features of target objects in images are important representations of images; however, the diversity of target object types in images, the structure and texture of the same region of interest, and the ambiguity of target region boundaries not only increase the difficulty of special representations but also bring difficulties to image classification in identifying target categories in images of various fields. Single classification techniques are often limited by the specificity of different data and classifier preferences. In this paper, multifeature fusion is applied to overcome the defects of image classification.

5. Conclusion

In this paper, based on deep learning, GAN and its related networks are used to achieve the effective rendering of sketches. The problem of insufficient sketch dataset in the rendering task is firstly solved by using CycleGAN network, and then Pix2Pix is used to add constraints to the generator to generate reasonably colored target images, and finally, an image rendering platform is built on this basis to further verify the effectiveness of the method in this paper. By improving the image data preprocessing part, the sketch generation model can efficiently extract the edge contour information of the minority dress image, enhance the sketch generation effect, and provide data support for the subsequent part of the coloring. The experimental results demonstrate that the structural similarity of the sketch generated by using this method reaches 0.854, and the correct detection rate is 0.793, which fully proves the effectiveness of the sketch generation method. Then, by comparing the methods of generators U-net 128 and U-net 256 and relock, it is proved that the model achieves a better coloring effect. It is demonstrated that the peak signal-to-noise ratio of sketch coloring for ethnic dresses using this model reaches 24.061, and the structural similarity reaches 0.820. Compared with other