

# Emerging Trends and Challenges in Knowledge Defined Networking for 5G

Lead Guest Editor: Yan Huang

Guest Editors: Abhishek Srivastava and Jinbao Wang





---

# **Emerging Trends and Challenges in Knowledge Defined Networking for 5G**

Wireless Communications and Mobile Computing

---

## **Emerging Trends and Challenges in Knowledge Defined Networking for 5G**

Lead Guest Editor: Yan Huang

Guest Editors: Abhishek Srivastava and Jinbao  
Wang



---


Copyright © 2021 Hindawi Limited. All rights reserved.

This is a special issue published in “Wireless Communications and Mobile Computing.” All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

# Chief Editor

































Zhipeng Cai , USA

## Associate Editors

Ke Guan , China  
Jaime Lloret , Spain  
Maode Ma , Singapore

## Academic Editors

Muhammad Inam Abbasi, Malaysia  
Ghufran Ahmed , Pakistan  
Hamza Mohammed Ridha Al-Khafaji ,  
Iraq  
Abdullah Alamoodi , Malaysia  
Marica Amadeo, Italy  
Sandhya Aneja, USA  
Mohd Dilshad Ansari, India  
Eva Antonino-Daviu , Spain  
Mehmet Emin Aydin, United Kingdom  
Parameshchhari B. D. , India  
Kalapaveen Bagadi , India  
Ashish Bagwari , India  
Dr. Abdul Basit , Pakistan  
Alessandro Bazzi , Italy  
Zdenek Becvar , Czech Republic  
Nabil Benamar , Morocco  
Olivier Berder, France  
Petros S. Bithas, Greece  
Dario Bruneo , Italy  
Jun Cai, Canada  
Xuesong Cai, Denmark  
Gerardo Canfora , Italy  
Rolando Carrasco, United Kingdom  
Vicente Casares-Giner , Spain  
Brijesh Chaurasia, India  
Lin Chen , France  
Xianfu Chen , Finland  
Hui Cheng , United Kingdom  
Hsin-Hung Cho, Taiwan  
Ernestina Cianca , Italy  
Marta Cimitile , Italy  
Riccardo Colella , Italy  
Mario Collotta , Italy  
Massimo Condoluci , Sweden  
Antonino Crivello , Italy  
Antonio De Domenico , France  
Floriano De Rango , Italy

Antonio De la Oliva , Spain  
Margot Deruyck, Belgium  
Liang Dong , USA  
Praveen Kumar Donta, Austria  
Zhuojun Duan, USA  
Mohammed El-Hajjar , United Kingdom  
Oscar Esparza , Spain  
Maria Fazio , Italy  
Mauro Femminella , Italy  
Manuel Fernandez-Veiga , Spain  
Gianluigi Ferrari , Italy  
Luca Foschini , Italy  
Alexandros G. Fragkiadakis , Greece  
Ivan Ganchev , Bulgaria  
Óscar García, Spain  
Manuel García Sánchez , Spain  
L. J. García Villalba , Spain  
Miguel Garcia-Pineda , Spain  
Piedad Garrido , Spain  
Michele Girolami, Italy  
Mariusz Glabowski , Poland  
Carles Gomez , Spain  
Antonio Guerrieri , Italy  
Barbara Guidi , Italy  
Rami Hamdi, Qatar  
Tao Han, USA  
Sherief Hashima , Egypt  
Mahmoud Hassaballah , Egypt  
Yejun He , China  
Yixin He, China  
Andrej Hrovat , Slovenia  
Chunqiang Hu , China  
Xuexian Hu , China  
Zhenghua Huang , China  
Xiaohong Jiang , Japan  
Vicente Julian , Spain  
Rajesh Kaluri , India  
Dimitrios Katsaros, Greece  
Muhammad Asghar Khan, Pakistan  
Rahim Khan , Pakistan  
Ahmed Khattab, Egypt  
Hasan Ali Khattak, Pakistan  
Mario Kolberg , United Kingdom  
Meet Kumari, India  
Wen-Cheng Lai , Taiwan

Jose M. Lanza-Gutierrez, Spain  
Pavlos I. Lazaridis , United Kingdom  
Kim-Hung Le , Vietnam  
Tuan Anh Le , United Kingdom  
Xianfu Lei, China  
Jianfeng Li , China  
Xiangxue Li , China  
Yaguang Lin , China  
Zhi Lin , China  
Liu Liu , China  
Mingqian Liu , China  
Zhi Liu, Japan  
Miguel López-Benítez , United Kingdom  
Chuanwen Luo , China  
Lu Lv, China  
Basem M. ElHalawany , Egypt  
Imadeldin Mahgoub , USA  
Rajesh Manoharan , India  
Davide Mattera , Italy  
Michael McGuire , Canada  
Weizhi Meng , Denmark  
Klaus Moessner , United Kingdom  
Simone Morosi , Italy  
Amrit Mukherjee, Czech Republic  
Shahid Mumtaz , Portugal  
Giovanni Nardini , Italy  
Tuan M. Nguyen , Vietnam  
Petros Nicolitidis , Greece  
Rajendran Parthiban , Malaysia  
Giovanni Pau , Italy  
Matteo Petracca , Italy  
Marco Picone , Italy  
Daniele Pinchera , Italy  
Giuseppe Piro , Italy  
Javier Prieto , Spain  
Umair Rafique, Finland  
Maheswar Rajagopal , India  
Sujan Rajbhandari , United Kingdom  
Rajib Rana, Australia  
Luca Reggiani , Italy  
Daniel G. Reina , Spain  
Bo Rong , Canada  
Mangal Sain , Republic of Korea  
Praneet Saurabh , India

Hans Schotten, Germany  
Patrick Seeling , USA  
Muhammad Shafiq , China  
Zaffar Ahmed Shaikh , Pakistan  
Vishal Sharma , United Kingdom  
Kaize Shi , Australia  
Chakchai So-In, Thailand  
Enrique Stevens-Navarro , Mexico  
Sangeetha Subbaraj , India  
Tien-Wen Sung, Taiwan  
Suhua Tang , Japan  
Pan Tang , China  
Pierre-Martin Tardif , Canada  
Sreenath Reddy Thummaluru, India  
Tran Trung Duy , Vietnam  
Fan-Hsun Tseng, Taiwan  
S Velliangiri , India  
Quoc-Tuan Vien , United Kingdom  
Enrico M. Vitucci , Italy  
Shaohua Wan , China  
Dawei Wang, China  
Huaqun Wang , China  
Pengfei Wang , China  
Dapeng Wu , China  
Huaming Wu , China  
Ding Xu , China  
YAN YAO , China  
Jie Yang, USA  
Long Yang , China  
Qiang Ye , Canada  
Changyan Yi , China  
Ya-Ju Yu , Taiwan  
Marat V. Yuldashev , Finland  
Sherali Zeadally, USA  
Hong-Hai Zhang, USA  
Jiliang Zhang, China  
Lei Zhang, Spain  
Wence Zhang , China  
Yushu Zhang, China  
Kechen Zheng, China  
Fuhui Zhou , USA  
Meiling Zhu, United Kingdom  
Zhengyu Zhu , China





# Contents

## **DDPG-Based Energy-Efficient Flow Scheduling Algorithm in Software-Defined Data Centers**

Zan Yao , Ying Wang , Luoming Meng , Xuesong Qiu , and Peng Yu 




Research Article (10 pages), Article ID 6629852, Volume 2021 (2021)

## **A Mobility-Aware and Sociality-Associate Computation Offloading Strategy for IoT**

Yanfei Lu, Zengzi Chen , Qinghe Gao , Tao Jing , and Jin Qian 


Research Article (12 pages), Article ID 9919541, Volume 2021 (2021)

## **Privacy-Aware Online Task Offloading for Mobile-Edge Computing**

Dali Zhu, Ting Li , Haitao Liu , Jiyan Sun, Liru Geng, and Yinlong Liu 

Research Article (16 pages), Article ID 6622947, Volume 2021 (2021)

## **Hybrid Precoding Algorithm for Millimeter-Wave Massive MIMO Systems with Subconnection Structures**

Xue Zhang and Feng Zhao 

Research Article (9 pages), Article ID 5532939, Volume 2021 (2021)

## **Dual-Level Attention Based on a Heterogeneous Graph Convolution Network for Aspect-Based Sentiment Classification**

Peng Yuan , Lei Jiang , Jianxun Liu , Dong Zhou , Pei Li , and Yang Gao 


Research Article (13 pages), Article ID 6625899, Volume 2021 (2021)

## **Optimize the Communication Cost of 5G Internet of Vehicles through Coherent Beamforming Technology**

Lan Wu, Juan Xu , Lei Shi , Yi Shi, and Wenwen Zhou




Research Article (12 pages), Article ID 6668984, Volume 2021 (2021)

## **Dynamically Subarray-Connected Hybrid Precoding Scheme for Multiuser Millimeter-Wave Massive MIMO Systems**

Guangyan Liao and Feng Zhao 




Research Article (10 pages), Article ID 5528522, Volume 2021 (2021)

## **Multitask Associated Task Scheduling for Cloud Computing Based on Task Duplication and Insertion**

Lei Shi , Jing Xu, Lunfei Wang, Jie Chen, Zhifeng Jin, Tao Ouyang, Juan Xu , and Yuqi Fan 



Research Article (13 pages), Article ID 6631752, Volume 2021 (2021)

## **Min- $k$ -Cut Coalition Structure Generation on Trust-Utility Relationship Graph**

XiangLong Kong , XiangRong Tong , and YingJie Wang 

Research Article (11 pages), Article ID 8834879, Volume 2021 (2021)

## **SDRM-LDP: A Recommendation Model Based on Local Differential Privacy**

Gesu Li , Guisheng Yin, Jishen Yang , and Fukun Chen

Research Article (15 pages), Article ID 6640667, Volume 2021 (2021)

## Research Article

# DDPG-Based Energy-Efficient Flow Scheduling Algorithm in Software-Defined Data Centers

Zan Yao , Ying Wang , Luoming Meng , Xuesong Qiu , and Peng Yu 

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China

Correspondence should be addressed to Ying Wang; wangy@bupt.edu.cn

Received 1 March 2021; Accepted 11 June 2021; Published 28 June 2021

Academic Editor: Yan Huang

Copyright © 2021 Zan Yao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid development of data centers, the energy consumption brought by more and more data centers cannot be underestimated. How to intelligently manage software-defined data center networks to reduce network energy consumption and improve network performance is becoming an important research subject. In this paper, for the flows with deadline requirements, we study how to design the rate-variable flow scheduling scheme to realize energy-saving and minimize the mean completion time (MCT) of flows based on meeting the deadline requirement. The flow scheduling optimization problem can be modeled as a Markov decision process (MDP). To cope with a large solution space, we design a DDPG-EEFS algorithm to find the optimal scheduling scheme for flows. The simulation result reveals that the DDPG-EEFS algorithm only trains part of the states and gets a good energy-saving effect and network performance. When the traffic intensity is small, the transmission time performance can be improved by sacrificing a little energy efficiency.

## 1. Introduction

With the development of 5G technology [1–4], more and more data centers as important carriers of data storage and processing will be established [5–7]. Worldwide data center energy consumption has reached about 8% of global energy consumption. It mainly comes from three aspects: network, server, and refrigeration system. The network energy consumption accounts for about 20% of the total data center energy consumption [8]. With the introduction of DVFS (Dynamic Voltage and Frequency Scaling) and hardware virtualization technology [9, 10], the energy consumption efficiency of servers has been greatly improved, and more and more new heat dissipation technologies have emerged [11]. The proportion of network energy consumption in data center energy consumption will continue to increase, and it cannot be ignored.

The network energy-saving technology can be divided into three types: topology design and transformation [12], device sleeping (DS) [13–18], and adaptive link rate (ALR) [19–21]. The energy-saving topology design and transformation mainly studies how to maximize the ratio of the forward-

ing capacity provided by the topology to the total energy consumption of the network under constraint conditions, to improve the utilization rate of topology energy consumption. As for DS technology, because in the data center network with SDN technology, the “rich connection” topology has more alternative routes. Under the constraints of network connectivity and network performance, the network flows are aggregated and transmitted in the subset  $T$  of the network topology  $G$ , and the devices and links in  $T/G$  are dormant, to achieve energy saving. The ALR-based energy cost model shows that the link energy consumption is exponentially related to the actual transmission rate on the link. When the energy-saving topology is determined, the ALR-based energy cost model is used to guide further improving the energy utilization efficiency of the network link, which is through adjusting the bandwidth usage of different links at different times. There are few studies on the third one. In this paper, we will study it.

The representative ALR-based energy-saving scheme is a preemptive Most-Critical-First scheduling algorithm for the flow proposed in Ref. [20]. It selects the interval with the largest energy consumption density as the critical interval, and all



the flows in this critical interval will be preferentially scheduled. The transmitting rate of every flow is constant. If the rate is time variable, then the energy saving can be more efficient. Hence, we propose a rate-variable routing scheme based on bandwidth sharing mechanism in Ref. [21]. It sorts the flows according to EDF (earlier deadline first) policy and then calculates the routes in turn. However, the scheduling order based on EDF is too simple and the energy efficiency still has the space to be improved.

In addition to the optimization goal of energy saving, network operators must ensure and optimize the QoS (quality of service). There are some representative flows on the data center network, such as search and social networks that generate many requests and responses, which need to go through the data center to perform the tasks requested by the user. The performance that the user is concerned about is the response speed of the requests, and a tolerable deadline is generally given [22]. If the mean completion time (MCT) of the flows can be decreased based on ensuring the deadline and ensuring the network energy efficiency, then the QoS will be greatly increased [23]. Above all, this paper will assume the route has been determined and try to design a rate variable flow scheduling mechanism to minimize network energy consumption and improve the QoS performance.

Most of the current works model the traffic engineering problem as a mixed-integer linear programming problem and propose heuristic algorithms. The dual optimal problem has a large solution space, and the above heuristic algorithms are close to traversal, and the scalability is limited. Researchers began to study DRL-based flexible traffic control mechanisms to improve the performance of the data center network. The DRL-based traffic engineering algorithms [24–30] are mostly about finding the best QoS route, and they are driven by experience to deal with the overly complex and dynamic network environment. At present, there is no DRL-based flow scheduling scheme with a variable rate as far as we know.

The proposed energy-saving scheduling optimization problem can be modeled as a Markov decision process (MDP) with a state space, action space, and reward function. Although the RL algorithm can learn from the surrounding environment itself, it still needs to design the corresponding features manually for it to be able to converge. In practical applications, the number of states may be large, and in many cases, the features are difficult to be designed manually. The neural network happens to have particularly good processing for massive data. And the flow rate is designed to be constant in the current literatures as we know. We suppose the flow transmitting rate is more flexible and variable, so it is a continuous control problem to design a variable rate energy-saving flow scheduling scheme. The deep deterministic policy gradient (DDPG) algorithm is one based on the actor-critic (AC) framework proposed by Lillicap et al. [31], which is based on the DQN and the deterministic policy gradient (DPG) method, and it is an effective method to solve the continuous control problem. Hence, we adopt the DDPG method to solve it.

In summary, this paper focuses on the energy-saving flow scheduling problem based on the ALR model and

applies the DDPG algorithm to solve it. Our main contributions are two folds as follows:

- (1) When the network topology and the routes of flows are determined, to further reduce the energy consumption and improve the QoS requirement, based on the ALR energy cost model, the energy-saving QoS flow scheduling problem and the dual optimization objective of minimum energy consumption and mean completion time of flows are proposed. The dual optimal problem is a continual control problem, and it has a large solution space, which can be modeled as a Markov decision process
- (2) Based on the advantages of DDPG in solving continuous control problems, and the problem of scalability, the DDPG Energy-Efficient Flow Scheduling (DDPG-EEFS) Algorithm is proposed to obtain the optimal scheduling scheme
- (3) Based on the ALR energy cost model, a rate variable flow transmission mechanism is proposed to flexibly scheduling flow and to balance the flow transmission on the link in time and space and improve the energy-saving effectively

The rest of the paper is organized as follows: Section 2 analyzes related works. Then, the dual-objective optimization problem is presented in Section 3. We propose the DDPG-EEFS algorithm to solve the problem in Section 4. The simulation results are presented to verify the feasibility and effectiveness of the proposed approach in Section 5. Finally, the conclusion is given in Section 6.

## 2. Related Work

In this part, we review the past studies from two aspects: data center network energy-saving technology and DRL-based network traffic control algorithm.

*2.1. Data Center Network Energy-Saving Technology.* In terms of data center network energy-saving technologies, there is a lot of research, which can be divided into energy-saving topology design [12], and energy-saving routing and flow scheduling schemes [13–21], which are mainly based on the DS energy cost model or the ALR energy cost model.

The first technology is designed from the perspective of topology design and transformation to save energy. Ref. [12] defined the influence parameters of links on topological connectivity and the threshold of network connectivity decline percentage. Under the constraint of the threshold, the links are deleted from the network topology and the topology is updated according to the increasing order of the influence parameters, to achieve the goal of network energy saving. Energy-saving topology is suitable to be applied in the initial stage of network construction.

The DS-based energy-saving schemes are mainly from the view of forwarding and routing. Ref. [13] proposed a method to construct an elastic tree, which can dynamically adjust the set of active nodes and links. While reducing

energy consumption, it can deal with burst traffic and has good fault-tolerant performance. Ref. [14], Ref. [15], and Ref. [16] all proposed heuristic energy-saving routing algorithm to minimize the number of active links, under constraints. Ref. [17] proposed an online switching mechanism of multiple topologies in the data plane, which can sleep some devices and ports to achieve energy saving while meeting the dynamic demand of traffic. Li et al. [18] explored a new energy-aware flow preempting scheduling method in the time dimension and used the policy of EXR (exclusive routing), i.e., each flow preempted the route according to the priority and occupied its own route. This kind of energy-saving routing and scheduling method based on the DS model refers to the state transition of switches and ports. So, this kind of method will take some time and the DS model is not suitable for real-time flow scheduling.

A few literatures are targeted to the ALR-based energy efficiency flow scheduling. In Ref. [20], the flows in critical interval with the largest energy consumption density will be scheduling first. The energy consumption efficiency of the network is further improved by balancing the flow in the link space and time. And the flow scheduling problem mainly involves the transmission rate and transmission time. The flow bandwidth allocation is fixed, i.e., the flow rate is constant in the current flow scheduling literature as we know. In this paper, we will provide a rate variable flow scheduling mechanism to minimize network energy consumption and improve MCT.

**2.2. DRL Algorithm for Network Traffic Control.** The solution space of the above algorithms in Section 2.1 is large, so the scalability of the above algorithms is limited. With the development of DRL, the recent trend in the field of network technology is to use AI algorithm to control and operate the network traffic. Ref. [24] proposed an adaptive multimedia flow control method based on DDPG to optimize QoE performances. In a complex and dynamic network environment, each multimedia flow is allocated appropriate bandwidth in each path based on experience rather than a mathematical model. Ref. [25] involves scheduling the transmission times and bandwidths of multiple flows. The state is defined as resource allocation, and the action is the route selection. The contribution ratio of multiple resources in reducing delay is quantified, so that the performance requirements of flows are transformed into resource requirements of flows. Then, the DRL agent interacts with the network continuously and obtains a feasible path adaptively according to the network state, which means allocating the optimal network resources for the flow, to improve the network throughput, the completion time of the flow, and the load balance of the link. Most of the above literatures focus on nondata center network, and the optimization objective is network throughput, delay, and other performance requirements. And there is few DRL-based energy-saving flow control study in software-defined DCN.

To optimize the energy-saving effectiveness in DCN, based on the DS energy consumption model, we put forward a DQN-based routing algorithm in Ref. [29]. To further save energy, based on the ALR network energy consumption

model, we will use the DDPG algorithm to design a variable flow rate scheduling mechanism to achieve saving energy and improve the QoS.

### 3. Model of Network System

**3.1. Motivation.** The data center network is an undirected graph and can be modeled as  $G(V, E)$ , where  $V$  is the set of switches,  $E$  is the set of links. The set of traffic that needs to be transmitted is defined as  $J = \{j_1, j_2, j_3, \dots, j_n\}$ , where each flow is defined as  $j_i = [p_i, q_i, r_i, d_i, w_i]$ ,  $j_i \in J$ .  $p_i$  and  $q_i$  indicate the source and destination nodes separately;  $r_i$  and  $d_i$  represent the start time and the deadline of the flow, respectively; and  $w_i$  means the data size that needs to be transmitted. Taking Figure 1 as an instance, we assume that one simple network which consists of five switches and six links and currently three flows come, where  $j_1 = [1, 3, 0, 4, 8]$ ,  $j_2 = [2, 5, 0, 3, 9]$ , and  $j_3 = [3, 5, 0, 2, 4]$ . We assume each flow uses a single path transmission, and the routing has been determined in advance; thus, active switches and links are also known ahead of time. Let  $P_i$  be the sequence of links through which  $j_i$  is routed, which is calculated by the shortest path principle. Here, we set  $P_1 = [l_1, l_3]$ ,  $P_2 = [l_4]$ , and  $P_3 = [l_3, l_4]$ . Therefore, the number set of the actual active switches is  $\{1, 2, 3, 5\}$ , and the number set of the actual active links is  $\{1, 3, 4\}$ , and both of which are marked with green in Figure 1.

The main problem in flow scheduling is to provide an appropriate flow scheduling scheme to minimize the energy consumption and minimum mean completion time of flows, while guaranteeing the transmission performance of the flow deadline. The scheduling scheme mainly includes flow transmission interval and transmission rate, which have a great impact on the link energy consumption and MCT. Flows can share the same link for transmission, or they can be transmitted by exclusive link mode. The flow can be transmitted at a constant speed or at a variable speed. So, there will be a large number of possible transmission combinations. Different transmission schemes will produce different energy consumption and MCT.

Here, we adopt the ALR power consumption model to calculate the power energy. The power consumption function [19]  $f(x_e)$  is given by Formula (1) to uniformly characterize the manner in which the energy is being consumed with respect to the transmission rate  $x_e$  of each link  $e \in E$ .

$$f(x_e) = \sigma + \mu x_e^\alpha, \quad (1)$$

where  $\sigma$ ,  $\mu$ , and  $\alpha$  are constants associated with the link type. Constant  $\sigma$  represents the idle power energy for maintaining link state, and  $\alpha > 1$  that means  $f(\cdot)$  is superadditive. Here, we define the parameter  $\alpha = 2$ . Constant  $C$  is the maximum link transmission rate.

For the three flows above, the time range is set from the current time to the next five units of time, and each link has five units of bandwidth. We give three representative transmission schemes as shown in Figure 2. The first scheme adopts the transmission strategy of uniform speed and shared bandwidth, and the deadline is guaranteed, and the energy consumption is  $15\sigma + 115\mu$ , and MCT is 3 units of

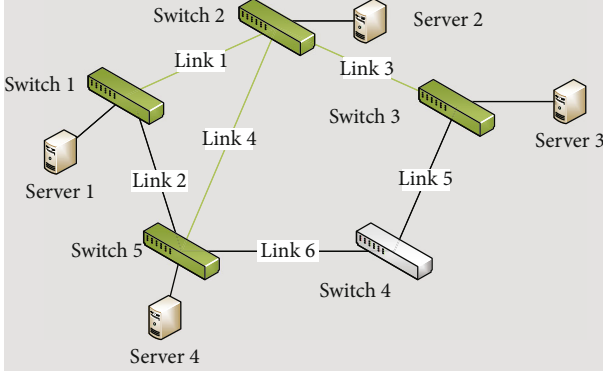


FIGURE 1: The simple network example.

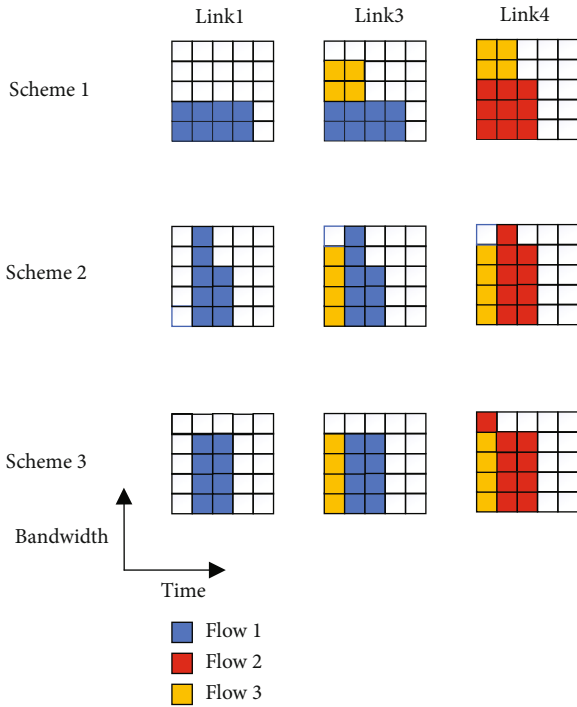


FIGURE 2: Different flow transmission schemes.

time. In the second scheme, the flow with earlier deadline is transmitted first, and the transmission strategy with the flow occupying the link bandwidth exclusively is adopted. The energy consumption is  $15\sigma + 141\mu$ , and MCT is 2.33 units of time. Scheme 3 is improved based on scheme 2, that is, the flow is transmitted as evenly as possible in time without changing the transmission interval. And its energy consumption is  $15\sigma + 137\mu$ , and MCT is 2.33 units of time. The average completion time of scheme 1 is the largest, and its energy consumption is the smallest. The average completion time of scheme 2 and scheme 3 is the smallest one, and the energy consumption of scheme 3 is smaller, because it balances the flow transmission in its transmission interval as much as possible. By adjusting the transmission rate and transmission interval, the energy consumption and the average completion time of the stream can be effectively reduced.

The goal of this paper is to find a flow transmission scheme to minimize the energy consumption and the MCT.

**3.2. Problem Formulation.** In this paper, the main optimization objective is to minimize the weighted sum of network link energy consumption and mean complete time (MCT) of flows, which is expressed by Formula (2), where the variables  $\phi'$  and  $MCT'$  are the normalized ones, which are, respectively, calculated by Formulas (3) and (4). The two constants  $\rho$  and  $(1 - \rho)$  represent the ratio between the energy consumption and MCT, respectively, and  $s$  represents one feasible solution of flow scheduling and will be included in the solution set  $s$ . Formulas (5)–(7) calculate the energy consumption  $\phi$  and MCT, respectively, where variables  $r_i'$  and  $d_i'$  represent the start time and ending time of the actual transmission of the flow  $j_i \in J$ . And constraints are expressed by Formulas (8)–(10):

$$s^* = \arg \min_s \left( \rho \phi' + (1 - \rho) MCT' \right), \quad (2)$$

$$\phi' = \frac{\phi - \min_{1 \leq j \leq n} \{\phi\}}{\max_{1 \leq j \leq n} \{\phi\} - \min_{1 \leq j \leq n} \{\phi\}}, \quad (3)$$

$$MCT'_{s_i} = \frac{MCTs_i - \min_{1 \leq j \leq n} \{MCTs_j\}}{\max_{1 \leq j \leq n} \{MCTs_j\} - \min_{1 \leq j \leq n} \{MCTs_j\}}, \quad (4)$$

$$\phi = \int_{t_0}^{t_{end}} \sum_{e \in E_a} (\sigma + \mu(x_e(t))^\alpha) dt, \quad (5)$$

$$x_e(t) = \sum_{e \in P_i} s_i(t), \quad (6)$$

$$MCT = \frac{\sum_{i=1}^n (d_i' - r_i')}{n}. \quad (7)$$

With constraints

$$\int_{r_i'}^{d_i'} s_i(t) dt = b_i, \quad (8)$$

$$0 \leq x_e(t) \leq \beta C, \quad (9)$$

$$\sum_{v \in N(u)} (f_i^{uv} - f_i^{vu}) = \begin{cases} b_i, & \text{if } u = p_i \\ -b_i, & \text{if } u = q_i \\ 0, & \text{else} \end{cases}. \quad (10)$$

Formulas (8)–(10) represent performance constraints. Formula (8) represents each flow must be completed before its latest deadline; Formula (9) represents link resource capacity constraints, i.e., the bandwidth used by network traffic cannot exceed the available bandwidth of the link. To ensure the availability of the link, the available bandwidth of the link is  $\beta$  times the link bandwidth capacity, and  $(1 - \beta)$  times the link bandwidth needs to be reserved for the emergency. Formula (10) represents the flow conservation

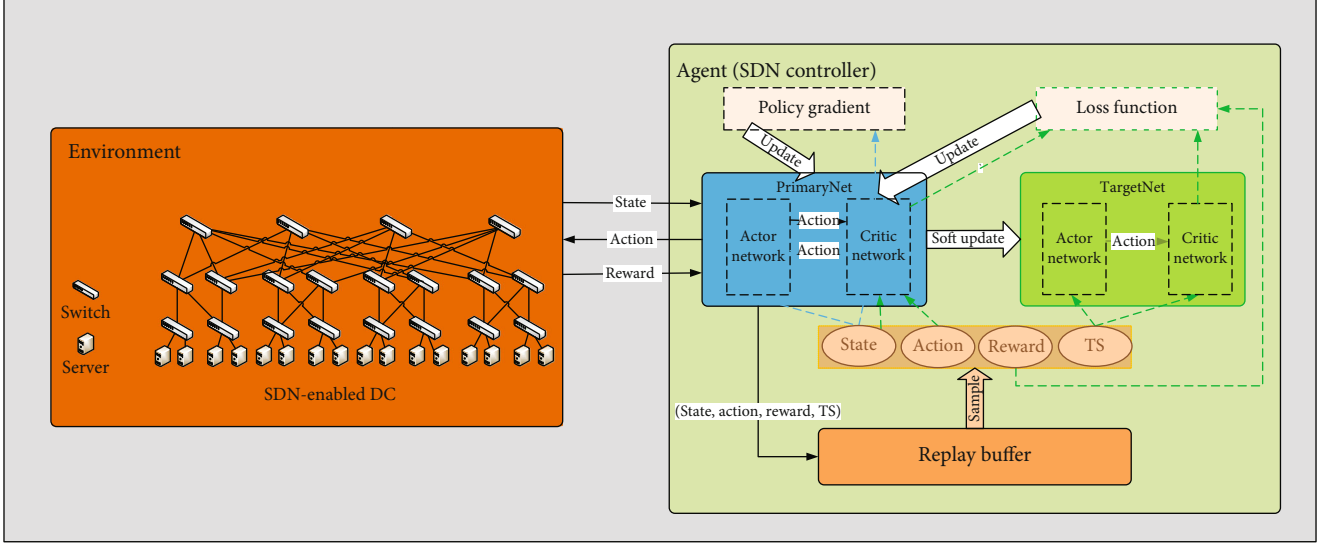


FIGURE 3: An overview of the DDPG-EEFS algorithm.

limit. The flow from the source switch is equal to the flow into the destination switch, and the flows from all intermediate switches are equal to the flows out.  $N(u)$  represents the neighbor switch set of the switch  $u$ , and  $f_i^{uv}$  represents the flow bandwidth deployed on the link. In the case of flow transmission over a simple path, this value is  $b_i$  or 0.

#### 4. DDPG-Based Energy-Efficient Flow Scheduling Algorithm

For the energy-saving scheduling optimization model established above, we model it as a Markov decision process (MDP) with a state space, action space, and reward function. This paper tries to apply the DDPG method to seek the most energy-saving transmission rate for each flow while minimizing the MCT of flows. Firstly, we propose a DDPG-EEFS architecture and describe the components. State, action, and reward are also outlined. Secondly, the process of the DDPG-EEFS algorithm is presented.

**4.1. DDPG-EEFS Architecture.** The DDPG-EEFS architecture is shown in Figure 3. It mainly includes environment and agent.

**4.1.1. Environment.** The environment is an SDN-enabled data center network, which consists of switches, links, and servers.

**4.1.2. Agent.** An agent is used to communicate with the environment. Once it observes the consequences, it learns to change its behavior and action in response to the reward. When DDPG is applied in the system, the SDN controller has a global view to get the state of the network environment, and it can be seen as an agent to make decisions based on observation and carry out a series of actions and take actions to the current state and provide flexible policy deployment.

DDPG combines the DQN method and the DPG method with actor critical framework and uses a neural network to fit

policy function and Q function to form a more efficient and stable control model. To improve the convergence and stability of the network, the important idea of experience replay and target network are used in the DDPG algorithm. The purpose of the former is to disrupt the correlation between the data, so that the sequence meets the independence and identical distribution. The latter regularly copies the online network parameters to the target network with the same structure and then uses the target network to update online network parameters.

(1) *Primary Network.* The primary network is used to determine an action based on the current state with the corresponding critic value. So, its input is the current state, and its output is an action. The primary network consists of an actor network and critic network. The actor network is the online policy network and is responsible for selecting the current action  $a_i$  according to the current state  $s_i$  and used to interact with the environment to generate the next state  $s_{i+1}$  and reward  $r_i$ . The critic network is used to approximate the value function  $Q(s, a|\theta^Q)$  of the state action pair (i.e., the output of actor network) and to provide gradient information and helps the actor to learn the gradient of the policy.

(2) *Target Network.* The target network is the same as the primary network model and consists of a target actor network and target critic network. The target actor network is responsible for selecting the optimal next action  $a_{i+1}$  according to the next state  $s_{i+1}$  sampled in the replay buffer, and its input is the transformed state (TS)  $(s_i, s_{i+1})$ , and the network parameter  $\theta^{\mu'}$  is periodically copied from  $\theta^\mu$ . The target critic network is responsible for calculating the value function  $Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$ , and the network parameter  $\theta^{Q'}$  is copied from  $\theta^Q$  periodically.

(3) *Replay Buffer.* The concept of experience replay is used to extract training samples during neural network training. The

observed state transition process is first stored in a replay buffer. After the samples in the replay buffer have accumulated to a certain extent, they will be randomly chosen to update the network. The main reason is that the samples obtained by randomly exploring the surrounding environment by different flows are a sequence associated with time. Due to the temporal correlation, if the data is directly used as a sample for training, the system convergence will be greatly affected, thereby the random sampling method solves the time correlation problem. This random extraction approach disrupts the correlation between experiences and makes neural network updates more efficient. In summary, the replay buffer is an especially important part of the DRL method, which greatly improves the system performance of DRL.

(4) *State*. The state in DDPG should reflect the situation of the environment. For the problem of flow with hard-deadline flow scheduling, the state of the environment mainly refers to the state of links, so we set  $s = [x_{e_1}(t), \dots, x_{e_i}(t), \dots, x_{e_k}(t)]$ , which is the sum of the transmission rates of the flows on each active link in the DCN, where  $k$  is the number of active links.

(5) *Action*. The agent focuses on mapping the space of state to the space of action and in identifying the optimal policy. The energy-saving flow scheduling is a continuous problem; its action space includes the bandwidth allocation of each flow in the available transmission interval, which is a continuous variable. DDPG is a preponderant method to solve it.

The available transmission time of each flow is divided into  $m_i = w_i/T$  time periods, where  $T$  is the minimum scheduling time unit. Action  $a$  refers to the transmission rate allocation of each flow in different time periods  $v_i = [v_{i,1}, \dots, v_{i,j}, \dots, v_{i,m_i}]$ ,  $1 \leq j \leq m_i$  and  $\sum_{j=1}^{m_i} v_{i,j} = 1$ . According to the flow transmission rate allocation, the real transmission rates of flows in the transmission interval can be calculated. By adjusting the transmission rate allocation situation of flows during different time periods, different flow scheduling schemes are obtained. Action  $a$  is defined as follows:

$$a = [[v_{1,1}, \dots, v_{1,j}, \dots, v_{1,m_1}], \dots, [v_{i,1}, \dots, v_{i,j}, \dots, v_{i,m_i}], \dots, [v_{l,1}, \dots, v_{l,j}, \dots, v_{l,m_l}]]. \quad (11)$$

(6) *Reward*. The reward  $r$  of agent is related to the adopted network operation and maintenance strategy. These control policies can be adjusted by changing the reward settings. The immediate reward is defined by analyzing the objective function. Since the objective function is to find the minimum energy consumption and mean completion time of flows, and the smaller the weighted sum of both items, the larger the reward, so the reciprocal of weighted sum can be immediately used as an immediate reward in Formula (12)

$$r = \begin{cases} \frac{1}{\rho\phi' + (1-\rho)\text{MCT}'}, & \text{if meet the constraints (8) - (10),} \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

For those states that do not satisfy the bandwidth constraints (8)–(10), the immediate reward is 0.

4.2. *DDPG-EEFS Algorithm Process*. The general process of DDPG optimizing flow scheduling is as follows: Firstly, the agent obtains an accurate network state. Secondly, the agent determines an action; then, the SDN controller produces rules and distributes them to the switches in the data plane. Finally, the agent obtains the rewards and new network state after the implementation of the new scheduling scheme. The training goal of the DDPG agent is to find the optimal action  $a$  according to the input state  $s$  to maximize the reward  $r$ . The process is shown with black solid lines in Figure 3.

The general DDPG framework cannot clearly define how to explore, and the TE-aware random search method [23] uses the basic TE solution  $a_{\text{base}}$  as the baseline to guide the exploration process. So here, we use the TE-aware random search method to improve the efficiency of exploration. The DRL agent generates action  $a_{\text{base}} + \varepsilon \cdot \mathbb{N}$  with probability  $\varepsilon$ , and action  $a + \varepsilon \cdot \mathbb{N}$  with probability  $(1 - \varepsilon)$ .  $a$  is the output of the actor network. The parameter  $\mathbb{N}$  is uniformly distributed random noises.  $a_{\text{base}}$  can be obtained by different methods. Every flow is supposed to transfer uniformly in the interval between arrival time and its deadline. Although it is not the best solution, it is enough as a fundamental solution. We decide to increase the probability of a random noise to the basic TE solution and the action of actor network instead of directly adopting  $a_{\text{base}}$  and  $a$ . Moreover,  $\varepsilon$  decreases with the increase of epoch because the more learning, the more action output will be adopted.

Deep neural networks are introduced in these networks, whose parameters are updated according to learning. To make efficient use of hardware optimizations, DDPG explores policy with an off-policy algorithm in minibatches, rather than online. For the flow scheduling problem, it can be considered as a continual control environment.

At each time step, the actor primary network and critic primary network are updated by sampling a minibatch from the replay buffer. Sampling is made up of a series of transitions, which contains the state, the action, the reward, and TS. Their updating is introduced as follows.

The reverse transmission of the policy gradient of the DPG neural network for actor module is shown in Equation (13), the related data flow is also shown in Figure 3 with blue dotted lines.

$$\begin{aligned} \nabla_{\theta^\mu} J &= \text{grad}[Q] * \text{grad}[\mu] \\ &\approx \frac{1}{N} \sum_i \nabla_a Q(s, a | \theta^Q) \Big|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu} \mu(s | \theta^\mu) \Big|_{s_i}, \end{aligned} \quad (13)$$

where  $J$  is the performance function to measure the performance of the policy  $\mu$ . The former in the right of equal sign is the action gradient from the critical network, which is used to characterize the direction of movement on which the action gets a higher return; the latter is the parameter gradient for the actor network, which is used to characterize how should the neural network of actor adjust its own parameters in order to make the neural network to select the action with the highest return with higher possibility. The combination of the two items means that the neural network of the actor module moves towards the direction with a higher possibility of getting a higher return to modify its own parameters.

The update process of the DQN network in a critical module is shown in Equation (14), which is also shown in Figure 3 with green dotted lines. TD\_Error (TD: temporal difference)  $L$  is equal to the mean square error of  $Q$  value of target network and the  $Q$  value of the online network, where the  $Q$  value of target network is shown in Equation (15), and it is based on the  $Q$  value of next state  $s_{i+1}$  and next action  $a_{i+1}$ . The next action here comes from the target network of the actor module.

$$L = \frac{1}{N} \sum_i \left( y_i - Q(s_i, a_i | \theta^Q) \right)^2, \quad (14)$$

$$y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1} | \theta^{\mu'}) | \theta^{Q'}). \quad (15)$$

Therefore, for the network energy-efficient flow scheduling problem, the DDPG-EEFS algorithm only trains some state data to get optimal results.

The simple uniform sampling method ignores the significance of the samples in the pool. Priority experience replay assigns a priority to each sample, and the DRL agent chooses one sample based on the priority to learn experience from transitions more effectively. Priority experience replay is applied to the TE problem [26]. DDPG includes actor and critical networks, so the priority also includes the following two parts. The first part is corresponding to the training of the critical network. The second part is related to the actor network training. Combining the two parts, the priority of samples is shown in Formula (16). When  $|\nabla_a Q|$  is the mean value of the absolute value of  $\nabla_a Q$ , parameter  $\varphi$  determines the relative importance between and  $\nabla_a Q$ .  $\zeta$  is a small constant to avoid the error of edge cases of transition, i.e., if the error is 0, it will need to be revisited. The probability of transition sampling is calculated by Formula (17), where the exponent  $\beta_0$  describes the priority. When  $\beta_0 = 0$ , the transition sampling becomes uniform sampling.

$$p_i = \varphi \cdot (L + \zeta) + (1 - \varphi) \cdot |\nabla_a Q|, \quad (16)$$

$$P(i) = \frac{p_i^{\beta_0}}{\sum_j p_j^{\beta_0}}. \quad (17)$$

## 5. Simulation and Results

To verify the effectiveness of the proposed DDPG-EEFS algorithm, simulation is conducted in the SDN-enabled data center network with Fat-Tree topology.

*5.1. Simulation Environment and Setting.* Under the Windows 10 system, the Python language is used to program the algorithm. The hardware platform is configured as a 2.4 GHz CPU and 64 GB memory. This work selects the commonly used Fat-Tree data center network topology, which is set to consist of 20 four-point switches, 16 hosts, and 48 links.

We mainly use the network energy-saving percentage  $P$  as the evaluation matrix of energy-saving effectiveness, that is, the network energy consumption saved by using the method  $A$  accounts for the percentage of the total network energy consumption  $NEC_{full}$  when all the links are full load. The specific definition is as shown in Formula (18):

$$P = 1 - \frac{NEC_A}{NEC_{full}}. \quad (18)$$

*5.2. Simulation Results and Analysis.* To verify the validity and performance of the proposed DDPG-EEFS algorithm, we mainly design the simulation from two parts.

Firstly, to verify the effectiveness of the algorithm, we design 64 flows, which belong to 4 different pods, and there are half of flows that go through the core switch. To deal with emergencies and failure recovery, the parameter of redundancy  $\delta$  is taken as 0.8 in our experiment. Our goal is to find the optimal scheduling scheme to make the objective function minimized. Through learning and constantly adjusting various parameters, we finally obtain the actual parameters in the stable convergence algorithm. The parameters in the algorithm are given in Table 1.

After the training of the DDPG-EEFS algorithm is completed, the model is saved and then tested, and the network will find a relatively ideal scheduling scheme. For the test results, the normalized optimization objective is counted every 100 steps, which are shown in Figure 4. It is found that after reaching 800 steps, the algorithm approaches convergence and the objective function tends to be stable.

Secondly, because energy-saving and performance optimization sometimes restrict each other, pure DDPG-EEFS, i.e., DDPG-EEFS without considering the synchronous optimization of network flow completion time performance, is chosen to compare with the base solution  $a_{base}$  proposed in Section 4, i.e., every flow is supposed to transfer uniformly in the interval between arrival time and its deadline, and the Most-Critical-First [4] algorithm to evaluate the energy-saving effect and network performance. As shown in Figure 5, we mainly use the network energy-saving percentage  $P$  as an evaluation index of the energy-saving effect. We use the energy cost when the network link load is full as the benchmark of comparison in the network energy-saving percentage. Figure 5 shows the network energy-saving percentage of different algorithms under different network loads. The difference between the energy-saving effects of pure DDPG-EEFS and the Most-Critical-First scheduling

TABLE 1: The parameters of the DDPG-EEFS algorithm.

Parameters	Values
Parameter in the objective function $\rho, (1 - \rho)$	0.9, 0.1
Learning rate $\lambda$ for actor and critic network	0.002, 0.01
Discount factor $\gamma$	0.98
Exponent $\beta_0$	0.6
Constant $\zeta$	0.01
Parameter $\varphi$	0.6
Batch size $N$	64
Buffer size $D$	10000
Training round $e$	1200

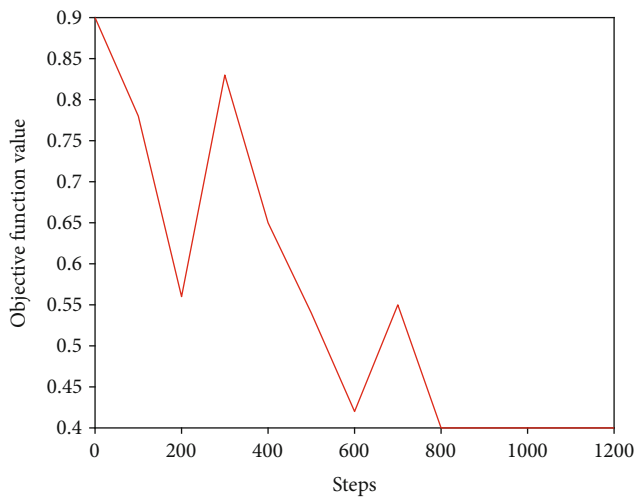


FIGURE 4: The process of searching the optimal scheduling scheme.

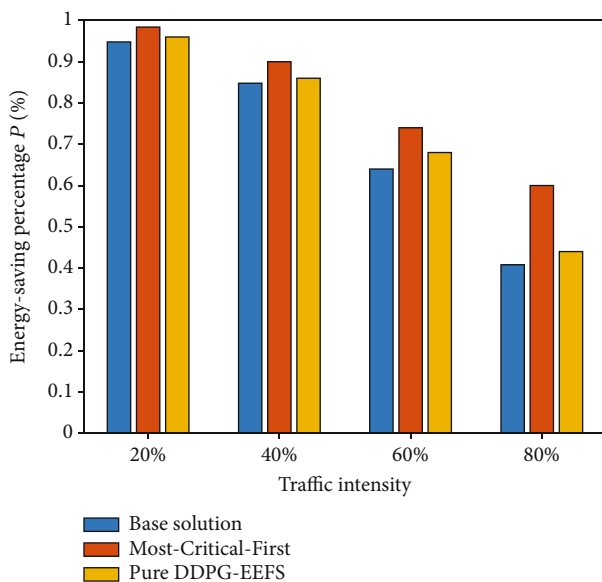


FIGURE 5: Energy-saving percentage at different traffic intensities.

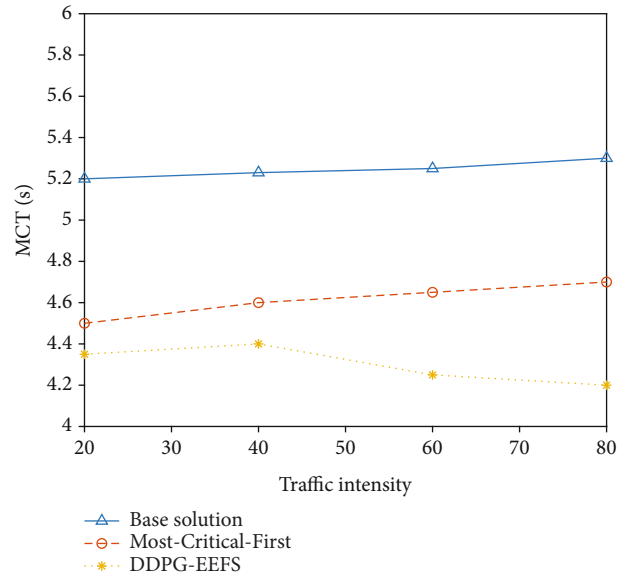


FIGURE 6: MCT of the three algorithms at different traffic intensities.

mechanism is about 2.4%. The static scheduling mechanism knows the information of all flows in advance, and it can carry out global optimization. Compared with the base solution  $a_{\text{base}}$ , the network energy-saving percentage of the pure DDPG-EEFS mechanism is increased by 3.2%.

As shown in Figure 6, we can see that under the different load conditions, the MCT by using DDPG-EEFS is lower than that of the baseline solution and Most-Critical-First algorithm. Compared with the base solution, the MCT of flows of DDPG-EEFS is reduced by about 15%.

## 6. Conclusions and Future Work

SDN is proposed as the promising technology in data center networks, and it can provide central network management and global traffic control. In this paper, we propose the dual optimization goals of the energy-saving and MCT of flows and design the DDPG-EEFS algorithm to solve it. For network operators, the shorter the average completion time is, the better the performance of the data plane is. Compared to other heuristic algorithms, DDPG-EEFS is easy to design the variable transmission rate and achieve good effect of energy saving, and good QoS.

### Data Availability

This work selects the commonly used Fat-Tree data center network topology, which is set to consist of 20 four-point switches, 16 hosts, and 48 links. We design 64 flows, which belong to 4 different pods, and there are half of flows that go through the core switch.

### Conflicts of Interest

The authors declare that there is no conflict of interest.

## Acknowledgments

This work was supported by the National Key R&D Program of China (2018YFE0205502).

## References

- [1] C. Zhang, Y.-L. Ueng, C. Studer, and A. Burg, "Artificial intelligence for 5G and beyond 5G: implementations, algorithms, and optimizations," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 10, no. 2, pp. 149–163, 2020.
- [2] L. Chettri and R. Bera, "A comprehensive survey on internet of things (IoT) toward 5G wireless systems," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 16–32, 2020.
- [3] X. Zheng, Z. Cai, J. Li, and H. Gao, "A study on application-aware scheduling in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 1787–1801, 2017.
- [4] T. Zhu, T. Shi, J. Li, Z. Cai, and X. Zhou, "Task scheduling in deadline-aware mobile edge computing systems," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4854–4866, 2019.
- [5] Z. Cai and Q. Chen, "Latency-and-coverage aware data aggregation scheduling for multihop battery-free wireless networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 1770–1784, 2021.
- [6] L. Yu, L. Chen, Z. Cai, H. Shen, Y. Liang, and Y. Pan, "Stochastic load balancing for virtual resource management in datacenters," *IEEE Transactions on Cloud Computing*, vol. 8, no. 2, pp. 459–472, 2020.
- [7] A. Tzanakaki, M. P. Anastasopoulos, and D. Simeonidou, "Converged optical, wireless, and data center network infrastructures for 5G services," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 11, no. 2, pp. A111–A122, 2019.
- [8] W. Yaqi, F. Baochuan, W. Zhengtian, and G. Shuang, "A review on energy-efficient technology in large data center," in *2018 Chinese Control And Decision Conference (CCDC)*, pp. 5109–5114, Shenyang, 2018.
- [9] A. Taha, S. Gosali, Z. Gong, N. Sollenberger, and J. Wright, "Method and system for dynamic voltage and frequency scaling (DVFS)," US Patent 12/190,029, 2009.
- [10] S. Samanta, R. Beddingfield, I. Wong, and S. Bhattacharya, "Efficient power transfer to data center racks using medium voltage inductive coupling," in *2019 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 1125–1130, Baltimore, MD, USA, 2019.
- [11] Y. Berezovskaya, A. Mousavi, V. Vyatkin, and X. Zhang, "Smart distribution of IT load in energy efficient data centers with focus on cooling systems," in *IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*, pp. 4907–4912, Washington, DC, 2018.
- [12] F. Cuomo, A. Abbagnale, A. Cianfrani, and M. Polverini, "Keeping the connectivity and saving the energy in the internet," in *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*, pp. 319–324, IEEE, 2011.
- [13] B. Heller, S. Seetharaman, P. Mahadevan et al., *ElasticTree: saving energy in data center networks/Proc of the 7th USENIX Conference on Networked Systems Design and Implementation*, USENIX, Berkeley, 2010.
- [14] M. Wei, J. Zhou, and Y. Gao, "Energy efficient routing algorithm of software defined data center network," in *2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN)*, IEEE, pp. 171–176, Guangzhou, China, 2017.
- [15] A. Fernandez-Fernandez, C. Cervello-Pastor, and L. Ochoa-Aday, "Achieving energy efficiency: an energy-aware approach in SDN," in *2016 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–7, Washington, DC: IEEE Press, 2016.
- [16] Z. Wu, X. Ji, Y. Wang, X. Chen, and Y. Cai, "An energy-aware routing for optimizing control and data traffic in SDN," in *2018 26th International Conference on Systems Engineering (ICSEng)*, pp. 1–4, Sydney: IEEE Press, 2018.
- [17] J. Ba, Y. Wang, X. Zhong, S. Feng, X. Qiu, and S. Guo, "An SDN energy saving method based on topology switch and rerouting," in *2018 IEEE/IFIP Network Operations and Management Symposium*, pp. 1–5, Taipei: IEEE Press, 2018.
- [18] D. Li, Y. Shang, W. He, and C. Chen, "EXR: greening data center network with software defined exclusive routing," *IEEE Transactions on Computers*, vol. 64, no. 9, pp. 2534–2544, 2015.
- [19] M. Andrews, A. F. Anta, L. Zhang, and W. Zhao, "Routing for power minimization in the speed scaling model," *IEEE/ACM Transactions on Networking*, vol. 20, no. 1, pp. 285–294, 2012.
- [20] G. Xu, B. Dai, B. Huang, J. Yang, and S. Wen, "Bandwidth-aware energy efficient flow scheduling with SDN in data center networks," *Future Generation Computer Systems*, vol. 68, pp. 163–174, 2017.
- [21] Y. Zan, W. Ying, Q. Xuesong, and W. Yuqi, "Routing optimization algorithm for SD-DCN based on delay and energy consumption," *Journal of Beijing University of Posts and telecommunication*, vol. 43, no. 2, pp. 46–51, 2020.
- [22] K. Fan, Y. Wang, J. Ba, W. Li, and Q. Li, "An approach for energy efficient deadline-constrained flow scheduling and routing," in *2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pp. 469–475, Washington DC, USA, 2019.
- [23] D. Zats, T. Das, P. Mohan, D. Borthakur, and R. Katz, "Detail: reducing the flow completion time tail in datacenter networks," in *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication - SIGCOMM '12*, pp. 139–150, Helsinki, Finland, 2012.
- [24] X. Huang, T. Yuan, G. Qiao, and Y. Ren, "Deep reinforcement learning for multimedia traffic control in software defined networking," *IEEE Network*, vol. 32, no. 6, pp. 35–41, 2018.
- [25] N. C. Luong, D. T. Hoang, S. Gong et al., "Applications of deep reinforcement learning in communications and networking: a survey," *IEEE Communications Surveys & Tutorials, Fourth quarter*, vol. 21, no. 4, pp. 3133–3174, 2019.
- [26] L. A. N. Julong, Y. U. Changhe, H. U. Yuxiang, and L. I. Ziyong, "A SDN routing optimization mechanism based on deep reinforcement learning," *Journal of Electronics and Information Technology*, vol. 41, no. 11, pp. 2669–2674, 2019.
- [27] Z. Xu, J. Tang, J. Meng et al., "Experience-driven networking: a deep reinforcement learning based approach," in *IEEE INFOCOM 2018- IEEE Conference on Computer Communications*, Honolulu, HI, USA, 2018.
- [28] M. B. Hossain and J. Wei, "Reinforcement learning-driven QoS-aware intelligent routing for software-defined networks," in *2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Ottawa, ON, Canada, 2019.



- [29] T. Ma, H. Yuxiang, and Z. Xiaohui, "Data center network coflow scheduling mechanism based on deep reinforcement learning," *Acta Electronica Sinica*, vol. 46, no. 7, pp. 1617–1624, 2018.
- [30] Z. Yao, Y. Wang, and X. Qiu, "DQN-based energy-efficient routing algorithm in software-defined data centers," *International Journal of Distributed Sensor Networks*, vol. 16, no. 6, 2020.
- [31] P. L. Timothy, J. H. Jonathan, P. Alexander et al., *Continuous Control with Deep Reinforcement Learning*, Proc. ICLR, San Juan, Puerto Rico, 2016.

## Research Article

# A Mobility-Aware and Sociality-Associate Computation Offloading Strategy for IoT

Yanfei Lu,<sup>1</sup> Zengzi Chen ,<sup>1</sup> Qinghe Gao ,<sup>1</sup> Tao Jing ,<sup>1</sup> and Jin Qian <sup>2</sup>

<sup>1</sup>School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China

<sup>2</sup>College of Computer Science & Technology, Taizhou University, Jiangsu, China

Correspondence should be addressed to Qinghe Gao; qhgao@bjtu.edu.cn

Received 13 March 2021; Revised 31 May 2021; Accepted 9 June 2021; Published 28 June 2021

Academic Editor: Yan Huang

Copyright © 2021 Yanfei Lu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Mobile edge computing, a promising paradigm, brings services closer to a user by leveraging the available resources in an edge network. The crux of MEC is to reasonably allocate resources to satisfy the computing requirements of each node in the network. In this paper, we investigate the service migration problem of the offloading scheme in a power-constrained network consisting of multiple mobile users and fixed edge servers. We propose an affinity propagation-based clustering-assisted offloading scheme by taking into account the users' mobility prediction and sociality association between mobile users and edge servers. The clustering results provide the candidate edge servers, which greatly reduces the complexity of observing all edge servers and decreases the rate of service migration. Besides, the available resource of candidate edge servers and the channel conditions are considered to optimize the offloading scheme to guarantee the quality of service. Numerical simulation results demonstrate that our offloading strategy can enhance the data processing capability of power-constrained networks and reach computing load balance.

## 1. Introduction

The arrival and evolution of the 5G delivers a transformative solution to an ultimate high-quality end-user experience. Such a high-speed, high-capacity, low-latency 5G network can well meet the demands of increasing data-intensive applications in the current Internet of Things (IoT) and Artificial Intelligence (AI) era. The numerous applications require high-speed internet connectivity and high computation power, which is not possible in a mobile device with limited memory and storage capacity [1–3]. In such situations, it is feasible to transfer resource-intensive tasks to external platforms like cloud, grid, and edge servers. This process is known as task offloading, which decides when and how a task should be offloaded to external platforms to execute.

During the development of 5G, the mobile edge computing (MEC) technique has been playing an important role [4, 5]. Massive smart edge devices in various IoT systems hide huge distributed computing capabilities [6, 7]. The reasonable utilization of distributed capability can reduce the amount and frequency of information transferring to a dis-

tant centralized cloud server, which can greatly decrease transfer latency and omit the cloud server processing latency. Although existing research can reduce the delay in some scenarios, many severe problems remain to be solved. Especially, how to migrate services as mobile users move is a critical problem. If users move far away from the MEC server that is responding to their request, this could result in significant quality of service (QoS) and quality of experience (QoE) degradation and service interruption due to long transmission latency of the offloaded task [8]. To ensure the continuity of service, services need to be migrated to nearby MEC servers that can cover the user's current location when it moved out of previous service coverage.

The migration of user-generated data in edge networks is involved in transmission costs, the mobility of users, transmission resources, etc. Thus, an efficient edge server selection algorithm is needed to select the optimal target edge server. In general, two factors should be taken into account: users' trajectory and QoS utility. On the one hand, existing research works rarely explore users' trajectory data and the prediction of their movement and adopt a random mobility model

instead [9]. However, users' mobility pattern (e.g. direction and velocity) has a significant influence on the construction of the candidate edge server set and the users' trajectory data can be used to predict users' movement. On the other hand, existing literatures pay less attention on the effect of QoS utility (network latency, energy consumption, and cost) on the selection of edge servers in service migration and, therefore, hardly select the edge server with the highest QoS utility [10–13]. Without considering users' trajectory data and QoS utility, the accuracy of edge server selection and the efficiency of service migration decrease. To develop a QoS-aware algorithm to improve edge server selection, we should overcome the problems such as how to integrate user's trajectory data and QoS utility into the server selection algorithm.

In this paper, we investigate an offloading scheme by considering users' trajectory data, sociality associations, and QoS parameters. First, we employ a Kalman filtering algorithm to do mobility prediction of users' trajectory; then, we compute the sociality association between mobile users and edge servers by using the history connection relationship. By using the mobility prediction and the social association results as input parameters of an affinity propagation algorithm, we propose a mobility-aware and sociality-associate clustering algorithm (MASACA), where MEC servers are divided into candidate sets associated with different users. Moreover, we devise a QoS utility function for each MEC server based on their available computation resource and the channel links' quality, which is used to determine the appropriate MEC server for a user to offload task. The main contributions of this paper are summarized as follows:

- (1) A mobility model of mobile users is considered, which is combined with a Kalman filter to make prediction about the location of users
- (2) The sociality association is taken into account, which guarantees the continuity of service
- (3) Clustering has been shown to achieve efficient and reliable management and reduce data congestion. We construct an AP-based algorithm to obtain stable clusters considering the trajectory data and the sociality association of mobile users

The paper is organized as follows: Section 2 gives a review of related works in the literature. Section 3 presents the system model and problem formulation. Section 4 gives the details of AP-MASACA. The QoS utility function encompassing outage probability and average coverage is derived in Section 5. Section 6 presents the performance of the proposed scheme by comparing with existing offloading schemes. And a summary is provided in Section 7.

## 2. Related Works

In order to optimize the delay and energy consumption of mobile devices, a computational offload strategy is adopted in mobile edge computing [14]. Mobility is one key challenging topic in MEC systems, which has effects on decisions in several domains such as caching, connected vehicles, and

especially computation offloading [15]. However, the mobility of mobile users and the limited coverage of edge servers can result in significant network performance degradation, dramatic drop in QoS, and even interruption of ongoing edge services [16]. Usually, the types of UE mobility can be categorized as random mobility, short-term predictable mobility, and fully known mobility, depending on whether the future location of the UE is known. Because of the limited resources of servers, the mobility of users, and the low latency requirements of service requests, computing offloading and service migrations are expected to occur in MEC systems regularly. The authors in [17] proposed a glimpse mobility prediction model based on the seq2seq model, which provides useful coarse-grain mobility information for Mobility-Aware Deep Reinforcement Learning training. However, the traditional offloading approaches (e.g., auction-based and game-theory approaches) cannot adjust the strategy according to changing environment when dealing with the mobility problem, in order to keep the system performance for a long time, an online user-centric mobility management scheme is proposed in [18] to maximize the edge computation performance while keeping the energy consumption of user's communication low by using Lyapunov optimization and multiarmed bandit theories. These proposed methods overcome the challenges of mobility in MEC, making contribution in reducing service delays or energy consumption of devices. However, by nature, minimizing the smart devices' energy consumption and tasks' execution time does not always coincide and may conflict in IoT.

Beside delay and energy consumption, the MEC system performance can be further improved by utilizing the recent developments of social networks [13, 19] and energy harvesting methods. Social IoT (SIoT) has been proposed as a promising way to create and maintain the collaborative relations among smart SIoT devices. The idea of SIoT is to build social collaborative networking for smart IoTs to achieve locally distributed data processing according to the rules set by owners [20]. The collaboration between SIoT and MEC is crucial to minimize network communications and computation, where the SIoT collaboration can locally process user requests and further reduce the communications and computation in MEC networks. Nevertheless, the load sharing among SIoT devices, MEC, and remote servers brings about new challenges for the communication and computation trade-off, cross-layer design in SIoT, and forwarding and aggregation trade-off [14, 21]. To solve this issue, Wang et al. in [22] formulated a new optimization problem SIoT Collaborative Group and Device Selection Problem (SCGDSP) and propose Optimal Collaborative Group Selection (OCGS) for a fundamental SCGDSP to find the intrinsic properties of collaborative group construction under the deployment of SIoTs in a grid street map to reduce the running time and transmission delay. However, one of the main challenges of social attribute is the selfish nature of the user, and some users only response what they are interested in and do not care about other people's needs. The authors in [23] proposed a physical-social-based cooperative cache framework to maximizing the social group utility and meet the data request of users. By considering the diversity of

TABLE 1: Summary of notations.

Notations	Meanings
$\mathcal{U}$	The set of mobile users
$\mathcal{S}$	The set of edge servers
$\mathcal{A}_i$	The task set of mobile user $U_i$
$h_{i,j}$	The channel fading coefficient between $U_i$ and $S_j$
$d_{i,j}$	The distance between $U_i$ and $S_j$
$N_0$	The power of additive white Gaussian noise (AWGN)
$R_{i,j}$	The channel rate between $U_i$ and $S_j$
$A_{in}^{i,k}$	The input size of task $a_k$ for $U_i$
$A_{out}^{i,k}$	The size of computation results task $a_k$ for $U_i$
$A_{com}^{i,k}$	The number of CPU clock cycles required by $a_k$ of $U_i$
$f_{loc}^i$	The computation capability of $U_i$
$F_{sev}^j$	The computation capability of $S_j$
$f_{sev}^{i,j,k}$	The actual CPU frequency of $S_j$ for the task $a_k$ of $U_i$
$T_{loc}^{i,k}$	The time delay for $U_i$ locally executing the task $a_k$
$T_{in}^{i,j,k}$	The time delay for $U_i$ transmitting the task $a_k$ to $S_j$
$T_{out}^{i,j,k}$	The time delay for $S_j$ sending computation results of the task $a_k$ to $U_i$
$T_{sec}^{i,j,k}$	The time delay for $U_i$ offloading task $a_k$ to $S_j$ without service migration
$T_{mig}^{i,k}$	The time delay for migrating computing results of the task $a_k$ of $U_i$ between two edge servers
$T_{sev,m}^{i,j,j,k}$	The time delay for $U_i$ offloading task $a_k$ to $S_j$ with service migration
$T_{sum}^{i,k}$	The time delay for completing the task $a_k$ for $U_i$
$T_{sum}$	The total time delay for completing all tasks of all users
$P_{loc}^i$	The average power when the CPU of $U_i$ is working
$P_{i,0}$	The average power $U_i$ is idle
$P_{i,R}$	The average power $U_i$ is receiving computation results
$P_{i,j}$	The transmission power of $U_i$ for transmitting tasks to $S_j$
$E_{loc}^{i,k}$	The energy consumption at $U_i$ for locally executing $a_k$
$E_{sev}^{i,j,k}$	The energy consumption at $U_i$ for offloading $a_k$ to $S_j$ without migration
$E_{sev,m}^{i,j,j,k}$	The energy consumption at $U_i$ for offloading $a_k$ to $S_j$ with migration
$E_{sum}^i$	The energy consumption at $U_i$ for completing its all tasks
$\lambda_{i,j,k}$	1 if $U_i$ transmits $a_k$ to $S_j$ , and otherwise 0
$\mu_{i,j,j',k}$	1 if $S_j$ transmits computing results for $a_k$ of $U_i$ to $S_{j'}$ , and otherwise 0
$\bar{\nu}$	The average service ratio

users, the authors of [24, 25] combined MEC with the mobile crowdsensing approach and proposed a social-based method to optimize the share of contents among users by exploiting

their mobility and sociality, which improved the performance of a content-sharing scenario.

There is no common definition of social strength in current researches about SIoT. Jung et al. [26] proposed a model named social strength prediction model, which infers social connections among smart objects and predicts the strength of the connections using the cousage data of the objects by two major components: (1) entropy-based and (2) distance-based social strength computations. These two components capture different properties of couages of objects, namely, the diversity and spatiotemporal features, which are all essential factors that contribute to the values of social strength. Wang et al. [27] put forward a method to measure the social strength generally by the available storage capacity and betweenness centrality. In this paper, we propose a definition that utilizes the information entropy to calculate social strength.

### 3. System Model and Problem Formulation

In this section, we give the system model and computation model of the task offloading and then formulate the offloading problem. For the sake of clarity, we summarize the notations in Table 1.

*3.1. System Model.* We consider a multiuser MEC model as shown in Figure 1 with a set of mobile users (MUs) denoted as  $\mathcal{U} = \{U_1, U_2, \dots, U_i, \dots, N\}$  and a set of edge servers as  $\mathcal{S} = \{S_1, S_2, \dots, S_j, \dots, S_M\}$ . According to physical positions of each user, they are clustered into different service cells surrounding edge servers. Assume that each MU has  $K$  computation tasks. Denote  $\mathcal{A}_i = \{a_1, a_2, \dots, a_k, \dots, a_K\}$  as the task set of  $U_i$ , where each  $a_k$  is independent and decomposable. For each task, a mobile user can proceed it locally or offload it to corresponding edge server within its service cell.

However, the service cell for an MU may change due to the MU's mobility. To guarantee seamless services, service migration is necessary. Specifically, as shown in Figure 2,  $U_2$  initially belonged to the service cell surrounding  $S_1$  and now is within the service cell surrounding  $S_2$ . If it offloaded computing tasks to  $S_1$  before moving out the initial cell and the computation results did not return to  $U_2$  before it enters into the current cell, computation results need to be migrated from  $S_1$  to  $S_2$ , then returned to  $U_2$ .

Thus, there are three computation offloading models: local execution, offloading without service migration, and offloading with service migration. In following sections, we will investigate how to select from these three computation offloading models to minimize the time delay for completing all tasks of all mobile users under the guarantee of successful offloading.

*3.2. Computation Offloading Model.* Define  $A_{in}^{i,k}$  as the input size of task  $a_k$  for  $U_i$ ,  $A_{out}^{i,k}$  the size of computation results returned to  $U_i$  for  $a_k$ , and  $A_{com}^{i,k}$  the total number of CPU clock cycles required to complete task  $a_k$ .

*3.2.1. Local Computing Model.* Let  $f_{loc}^i$  denote the computation capability of  $U_i$ , which is dependent on the intrinsic

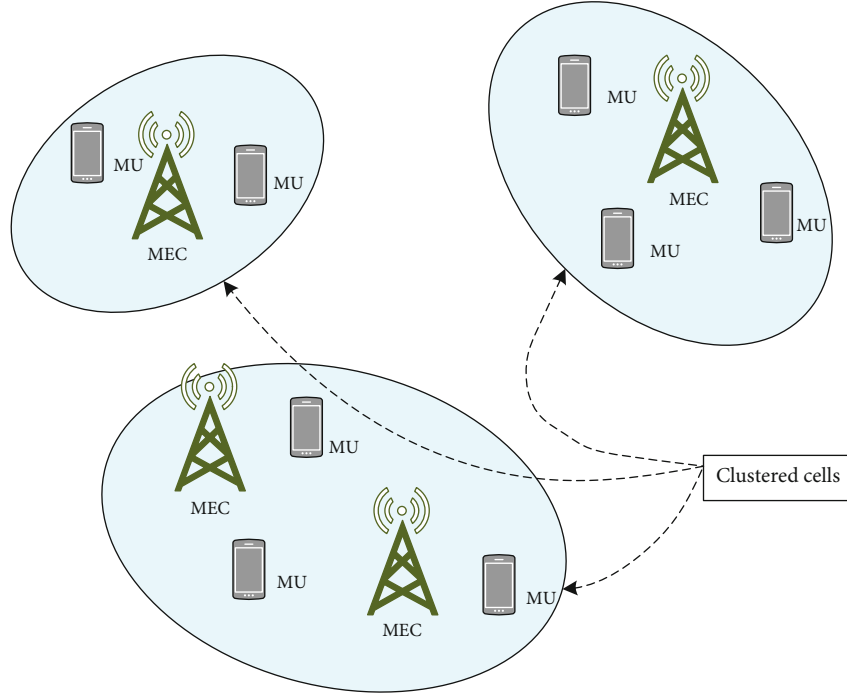


FIGURE 1: System model: service cells are clustered by mobile edge computing servers and mobile users.

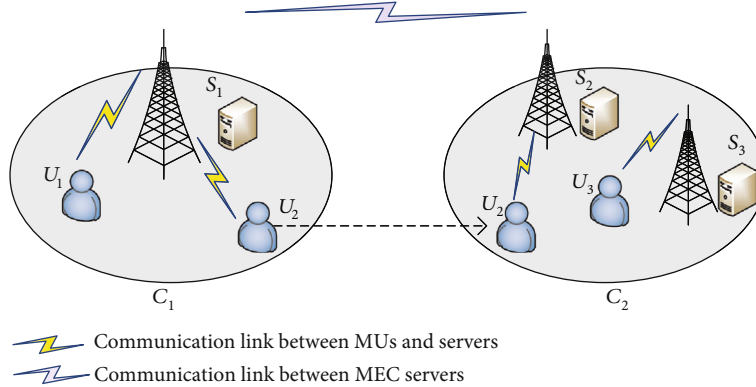


FIGURE 2: Mobility-aware MEC network.

nature of the MU, i.e., CPU cycles per second. The time  $T_{\text{loc}}^{i,k}$  for locally processing the request of task  $a_k$  at  $U_i$  is expressed as

$$T_{\text{loc}}^{i,k} = \frac{A_{\text{com}}^{i,k}}{f_{\text{loc}}^i}. \quad (1)$$

Assume that the energy consumption per second when the CPU is working is  $p_{\text{loc}}^i$ . Then, the energy consumption  $E_{\text{loc}}^{i,k}$  of local executing the task  $a_k$  for  $U_i$  can be given as

$$E_{\text{loc}}^{i,k} = p_{\text{loc}}^i T_{\text{loc}}^{i,k}. \quad (2)$$

**3.2.2. Offloading without Service Migration.** Three phases are needed to offload computation task to an edge server

without service migration: (i) an MU  $U_i$  transmits task request to an edge server  $S_j$ ; (ii) edge server  $S_j$  executes computing task; and (iii) edge server  $S_j$  sends computing results to  $U_i$ .

Assume that the wireless channel bandwidth between  $U_i$  and  $S_j$  is  $W_{i,j}$ . Let  $p_{i,j}$  represent the transmission power of  $U_i$  for transmitting tasks to  $S_j$ . The transmission rate  $R_{i,j}$  between  $U_i$  and  $S_j$  can be calculated by

$$R_{i,j} = W_{i,j} \log_2 \left( 1 + \frac{p_{i,j} h_{i,j}^2}{d_{i,j}^\theta N_0} \right), \quad (3)$$

where  $h_{i,j}$  represents the transmission channel fading coefficient,  $d_{i,j}$  is the distance from  $U_i$  to  $S_j$ ,  $\theta$  is the standard path loss propagation exponent, and  $N_0$  denotes the power of

additive white Gaussian noise (AWGN). Then, the transmission phase  $T_{in}^{i,j,k}$  from  $U_i$  to  $S_j$  for the task  $a_k$  can be given as

$$T_{in}^{i,j,k} = \frac{A_{in}^{i,k}}{R_{i,j}}. \quad (4)$$

The computing capability of an edge server is denoted as  $F_{sev}^j$ . Let  $f_{sev}^{i,j,k}$  denote the actual CPU frequency when edge server  $S_j$  executes the task  $a_k$  received from  $U_i$ , where  $0 \leq f_{sev}^{i,j,k} \leq F_{sev}^j$ . Then, the time  $T_{com}^{i,j,k}$  for edge server  $S_j$  processing the request of task  $a_k$  from  $U_i$  can be calculated as

$$T_{com}^{i,j,k} = \frac{A_{com}^{i,k}}{f_{sev}^{i,j,k}}. \quad (5)$$

After completing the computation task, edge server  $S_j$  needs to send back the computation results to  $U_i$ . The time delay for this transmission phase is expressed as

$$T_{out}^{i,j,k} = \frac{A_{out}^{i,k}}{R_{i,j}}. \quad (6)$$

Then, the total time delay for offloading task  $a_k$  to an edge server  $S_j$  without service migration can be computed as

$$T_{sec}^{i,j,k} = T_{in}^{i,j,k} + T_{com}^{i,j,k} + T_{out}^{i,j,k}. \quad (7)$$

Assume the energy consumption per second for  $U_i$  during the idle duration and the receiving phase are denoted as  $p_{i,0}$  and  $p_{i,R}$ , respectively. In total, the energy consumption at  $U_i$  for executing the task  $a_k$  on MEC sever  $S_j$  without service migration can be formulated as

$$E_{sev}^{i,j,k} = p_{i,j} T_{in}^{i,j,k} + p_{i,0} T_{com}^{i,j,k} + p_{i,R} T_{out}^{i,j,k}. \quad (8)$$

**3.2.3. Offloading with Service Migration.** If  $U_i$  moves from a cell surrounding  $S_j$  to current cell surrounding  $S_j'$  before  $S_j$  completes the computing task  $a_k$ , one more migration phase is needed to transmit computation results from edge server  $S_j$  to current edge server  $S_j'$ . The time delay of this migration phase is determined by the transmission rate  $R_s$  between these two edge servers as follows:

$$T_{mig}^{i,k} = \frac{A_{out}^{i,k}}{R_s}, \quad (9)$$

where  $R_s$  is a fixed value because of optical fiber communications between edge servers. Then,  $S_j'$  forwards the computation results to  $U_i$ . Similar to (6), the transmission time can be given as

$$T_{out}^{i,j',k} = \frac{A_{out}^{i,k}}{R_{i,j'}}, \quad (10)$$

where the transmission rate  $R_{i,j'}$  can be calculated in a similar way with (3). Thus, the total time delay  $T_{sev,m}^{i,j,j',k}$  for offloading with service migration can be calculated as

$$T_{sev,m}^{i,j,j',k} = T_{in}^{i,j,k} + T_{com}^{i,j,k} + T_{mig}^{i,k} + T_{out}^{i,j',k}. \quad (11)$$

Likewise as (8), the energy consumption at  $U_i$  for executing the task  $a_k$  by offloading with service migration can be calculated as

$$E_{sev,m}^{i,j,j',k} = p_{i,j} T_{in}^{i,j,k} + p_{i,0} (T_{com}^{i,j,k} + T_{mig}^{i,k}) + p_{i,R} T_{out}^{i,j',k}. \quad (12)$$

**3.3. Problem Formulation.** For convenience, we define a binary variable  $\lambda_{i,j,k}$  to equal to 1 if  $a_k$  of  $U_i$  is offloaded to  $S_j$  and otherwise equal to 0. When  $a_k$  of  $U_i$  is offloaded to  $S_j$ , we define  $\mu_{i,j,j',k}$  to be 1 if service migration from  $S_j$  to  $S_j'$  is necessary and otherwise equal to 0. Thus, the time delay for completing the task  $a_k$  for  $U_i$  can be expressed as

$$\begin{aligned} T_{sum}^{i,k} = & T_{loc}^{i,k} * \left( 1 - \sum_j \lambda_{i,j,k} \right) + T_{sev}^{i,j,k} \lambda_{i,j,k} \left( 1 - \sum_{j'} \mu_{i,j,j',k} \right) \\ & + T_{sev,m}^{i,j,j',k} \lambda_{i,j,k} \sum_{j'} \mu_{i,j,j',k}. \end{aligned} \quad (13)$$

The total time delay  $T_{sum}$  for completing all tasks of all users can be given by

$$T_{sum} = \sum_{i=1}^N \sum_{k=1}^K T_{sum}^{i,k}. \quad (14)$$

In summary, the problem of minimizing the total time delay by optimizing the transmission power matrix  $\mathbf{P} = \{\mathbf{p}_1, \dots, \mathbf{p}_i, \dots, \mathbf{p}_N\} \in \mathbb{R}^{M \times N}$ , the offloading matrix  $\mathbf{\Lambda} \in \mathbb{R}^{M \times N \times K}$ , and the migration matrix  $\mathbf{\Gamma} \in \mathbb{R}^{M \times N \times K}$  can be formulated as follows:

$$\begin{aligned} (\mathbf{P1}): \quad & \min_{\mathbf{P}, \mathbf{\Lambda}, \mathbf{\Gamma}} T_{sum} \\ \text{s.t.} \quad & C_1 : 0 \leq f_{sev}^{i,j,k} \leq F_{sev}^j \\ & C_2 : 0 \leq p_{i,j} \leq p_{max}^i \\ & C_3 : \lambda_{i,j,k} \in \{0, 1\}, \forall i, j, k \\ & C_4 : \mu_{i,j,j',k} \in \{0, 1\}, \forall i, j, j', k \\ & C_5 : \sum_j \lambda_{i,j,k} \leq 1, \forall i, k \\ & C_6 : \sum_{j', j''} \mu_{i,j,j',k} \leq 1, \forall i, k \\ & C_7 : E_{sum}^i \leq E_{max}^i, \quad \forall i, \end{aligned} \quad (15)$$

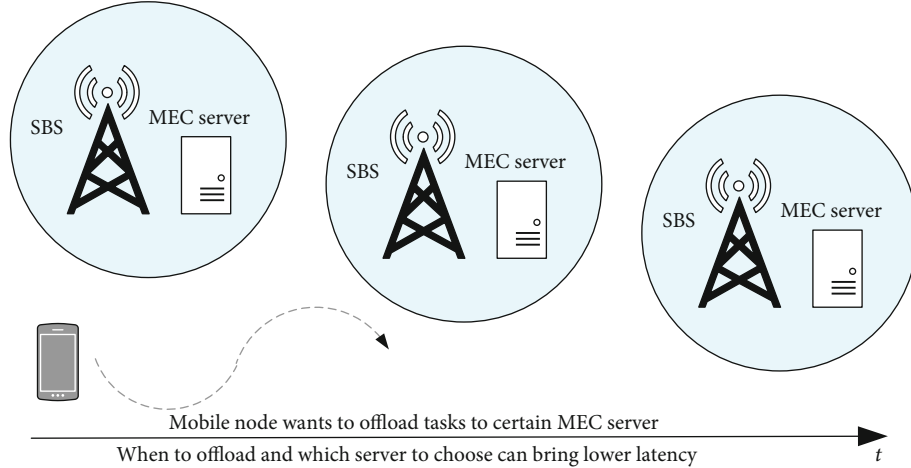


FIGURE 3: Mobile node wants to get optimized offloading strategies during the movement.

where

$$E_{\text{sum}}^i = \sum_k^K \left[ E_{\text{loc}}^{i,k} * \left( 1 - \sum_j \lambda_{i,j,k} \right) + E_{\text{sev}}^{i,j,k} \lambda_{i,j,k} \left( 1 - \sum_{j'} \mu_{i,j,j'}^{',k} \right) + E_{\text{sev},m}^{i,j,k} \lambda_{i,j,k} \sum_{j'} \mu_{i,j,j'}^{',k} \right]. \quad (16)$$

Constraint  $C_1$  is the computing capacity limitation of edge server, and constraint  $C_2$  is the power limitation with  $p_{\text{max}}^i$  being the maximum tolerable transmission power at  $U_i$ . Constraints  $C_3$  and  $C_4$  indicate that  $\lambda_{i,j,k}$  and  $\mu_{i,j,j'}^{',k}$  are binary variables. Constraint  $C_5$  gives the requirement that each task that can be totally offloaded to an edge server or be executed locally as a whole. Constraint  $C_6$  ensures the service migration occurs between one-hop edge servers. The constraint  $C_7$  is the energy constraint at  $U_i$  with  $E_{\text{max}}^i$  being the maximum available energy of  $U_i$ . Due to the integer constraints  $C_3$ - $C_6$ ,  $\mathbf{P1}$  is a mixed integer programming problem and NP-hard. In order to reduce the complexity of solving this NP-hard problem, in the following sections, we propose a mobility-aware and sociality-associate affinity propagation clustering algorithm, and then design an offloading scheme based on the clustering results.

#### 4. Mobility-Aware and Sociality-Associate Clustering Algorithm (MASACA)

As shown in Figure 3, a MU may meet different edge servers during its movement towards its destination. One of method to decrease the complexity of solving  $\mathbf{P1}$  is cutting down the number of candidate edge servers. In this section, we propose an affinity propagation- (AP-) based clustering by utilizing mobility prediction and sociality association results.

The MU can determine to randomly offload its tasks to an adjacent edge server at the moment the tasks arrive. In this

case, service migration may occur due to the movement of MU. When the server completes the computing tasks, the MU may have moved to a remote position and the connection with this server may be lost, leading to the recalculation of tasks and bringing additional overhead. To avoid much service migration and decrease the lost rate of computation results, we use Kalman filtering method to make trajectory prediction. According to the prediction results, the mobile user can select the best timing to offload its tasks to a better edge server.

**4.1. Mobility Prediction.** The movement of the mobile user is considered as a traditional dynamic system satisfying the following state equation at time  $t = aT$ :

$$\mathbf{s}_a = \mathbf{A}\mathbf{s}_{a-1} + \mathbf{B}\mathbf{u}_{a-1} + \omega_a, \quad (17)$$

where  $\mathbf{s}_a$  is the state vector and equals  $\{p_x, p_y, v_x, v_y\}_a$  representing the user's real-time position  $\{p_x, p_y\}_a$  and its velocity  $\{v_x, v_y\}_a$  in a two-dimensional space;  $\mathbf{A}$  is the transition matrix;  $\mathbf{u}_{a-1}$  is the input signal from previous time, namely, the moving velocity of the user;  $\mathbf{B}$  is the input matrix;  $\omega_a$  is a deterministic process noise and is usually assumed as a Gaussian noise with  $\omega \sim \mathcal{N}(0, \mathbf{Q})$ ; and  $\mathbf{Q}$  is called the process noise covariance matrix. The observation model is denoted as

$$\mathbf{z}_a = \mathbf{H}\mathbf{s}_a + \mathbf{y}_a, \quad (18)$$

where  $\mathbf{z}_a$  is the observation vector,  $\mathbf{H}$  is the observation matrix, and  $\mathbf{y}_a$  is the observation noise and also seen as a Gaussian noise with  $\mathbf{y} \sim \mathcal{N}(0, \mathbf{R})$ .  $\mathbf{R}$  is called the observation noise covariance matrix. Note that  $\omega_a$  and  $\mathbf{y}_a$  describe deviations such as collision, slippage, and friction that caused by objective conditions. Thus, in practice, the process noise covariance  $\mathbf{Q}$  and the observation matrix  $\mathbf{R}$  might change within each time step and are assumed to be temporally and spatially independent with each other.

According to above analyses, let  $\widehat{\mathbf{s}}_a$  denote the estimate value of  $\mathbf{s}_a$  when the measurement value of  $\mathbf{z}_a$  is obtained. The dynamic estimation process includes two steps:

(i) The time update phase follows the next equations as

$$\begin{cases} \widehat{\mathbf{s}}_a^- = \mathbf{A}\widehat{\mathbf{s}}_{a-1} + \mathbf{B}\widehat{\mathbf{u}}_a, & a \geq 1, \\ \mathbf{G}_a^- = \mathbf{A}\mathbf{G}_{a-1}\mathbf{A}^* + \mathbf{Q}, \end{cases} \quad (19)$$

where  $(\cdot)^-$  is the prediction value of a variable and  $(\cdot)^*$  and  $(\cdot)^{-1}$  represent the transpose matrix and inverse matrix, respectively.  $\mathbf{G}$  is the error covariance matrix

(ii) The measurement update phase is as follows:

$$\begin{cases} \mathbf{K}_a = \mathbf{G}_a^- \mathbf{H}^* (\mathbf{H}\mathcal{P}_a^- \mathbf{H}^* + \mathbf{R})^{-1}, \\ \widehat{\mathbf{s}}_a = \widehat{\mathbf{s}}_a^- + \mathbf{K}_a (\mathbf{z}_a - \mathbf{H}\widehat{\mathbf{s}}_a^-), \\ \mathbf{G}_a = (\mathbf{I} - \mathbf{K}_a \mathbf{H}) \mathbf{G}_a^-, \end{cases} \quad (20)$$

where matrix  $\mathbf{K}$  is the Kalman filtering gain and  $\mathbf{I}$  represents unit matrix

Thus, the Kalman filtering prediction algorithm starts with an initial predicted state  $\widehat{\mathbf{s}}_0^-$  and a certain error covariance  $\mathbf{G}_0^- = \mathbf{I}\mathbf{I}_0$  then executes the time update and measurement update by iterations until satisfying a certain termination condition. Finally, we can get the predicted trajectory of each mobile user, including the position and the velocity information at each time slot.

**4.2. Sociality Association.** In this subsection, we consider the influence of sociality association on choosing appropriate edge servers for offloading tasks. Since there is an acquaintance effect, we mainly take the history connections into account. And we use the Renyi entropy to describe the selection preference. A higher entropy represents higher uncertainty on a specific selection. Specifically, the Renyi entropy is denoted as

$$H_\alpha^i(P) = \frac{1}{1-\alpha} \log \left( \sum_{j=1}^M p_{i,j}^\alpha \right), \quad (21)$$

where  $\alpha$  is a general parameter in Renyi entropy. Different values of  $\alpha$  represent different entropy; for example, when  $\alpha = 1$ , it is the Shannon entropy as  $\lim_{\alpha \rightarrow 1} H_\alpha(P) = \sum_{j=1}^M \log(1/p_{i,j})$ . And  $p_{i,j}$  is the probability of  $U_i$  communicates with  $S_j$ . A smaller value of  $H_\alpha^i(P)$  means a higher possibility that  $U_i$  will offload its tasks to the server  $j_m$  with the maximum value of  $j_m = \arg \max p_{i,j}, \forall j \in \{1, 2, \dots, M\}$ .

Let  $\mathcal{V}$  represent the set of social association matrix between edge servers and mobile users.  $\mathcal{V}_{i,j,a} = \langle i, j, t_a \rangle$  represents that  $U_i$  communicates with  $S_j$  during  $t = aT$ . For example,  $\mathcal{V}_{i,\cdot} = (\langle t_3 \rangle, \langle t_1, t_{15} \rangle, 0, \dots, \langle t_{10}, t_{12}, t_{19} \rangle)$  is the association vector for  $U_i$ , demonstrating that  $U_i$  has con-

nected with  $S_1$  at  $t_3$ ,  $S_2$  at  $t_1$  and  $t_{15}$ , and  $S_M$  at  $t_{10}, t_{12}, t_{19}$  and never connected with  $S_3$ . The set of request user  $i$  was proposed for edge server  $j$ . Let  $w_{i,j}$  denote the communication frequency between  $U_i$  and  $S_j$ , e.g.,  $w_{i,1} = 1, w_{i,2} = 2, w_{i,3} = 0, \dots, w_{i,M} = 3$ . Define  $\Omega_i = \sum_j w_{i,j}$  representing the total number of  $U_i$  communicate with all edge servers. Thus, the probability  $p_{i,j}$  is calculated as

$$p_{i,j} = \frac{w_{i,j}}{\Omega_i}, \quad \text{for } U_i \in \mathcal{U}, S_j \in \mathcal{S}. \quad (22)$$

Similarly,  $\mathcal{V}_{\cdot,j}$  can be easily obtained, denoting the social association between  $S_j$  and all users and  $q_{j,i}$  can be also calculated as

$$q_{j,i} = \frac{w'_{j,i}}{\Omega'_j}, \quad \text{for } U_i \in \mathcal{U}, S_j \in \mathcal{S}. \quad (23)$$

**4.3. AP Clustering.** To accelerate the determination on connection with appropriate edge servers, we propose mobility-aware and sociality-associate clustering algorithm. Without the need of giving the number of clusters, the affinity propagation- (AP-) based clustering is employed. For the sake of striking the balance between accuracy and complexity, the update window is set as  $L$  slots.

**4.3.1. Similarity Definition.** The AP algorithm takes similarity,  $s(i, j)$ , as a measure describing the resemblance between nodes.  $s(i, j) \geq s(i, j')$  when node  $U_i$  has higher similarity with  $S_j$  than with  $S_{j'}$ . In the mobility model, the similarity is probable to be higher when two nodes have closer distance between them. Thus, we define the similarity as the reciprocal of the average distance between nodes with future  $L$  slots according to the mobility prediction results.

$$\begin{aligned} d(i, j) &= \frac{\sum_{t=aT}^{t=[a+(L-1)]T} |(\widehat{p}_x, \widehat{p}_y)_t^i - (p_x, p_y)_t^j|}{L}, \\ d(i, i') &= \frac{\sum_{t=aT}^{t=[a+(L-1)]T} |(\widehat{p}_x, \widehat{p}_y)_t^i - (\widehat{p}_x, \widehat{p}_y)_t^{i'}|}{L}, \\ d(j, j') &= \frac{\sum_{t=aT}^{t=[a+(L-1)]T} |(p_x, p_y)_t^j - (p_x, p_y)_t^{j'}|}{L}. \end{aligned} \quad (24)$$

Thus, we have

$$\begin{aligned} s(i, j) &= \frac{1}{d(i, j)}, \\ s(i, i') &= \frac{1}{d(i, i')}, \\ s(j, j') &= \frac{1}{d(j, j')}. \end{aligned} \quad (25)$$



```

Input:
•The user set  $\mathcal{U}$  and edge servers set  $\mathcal{S}$ ;
•The positions of edge servers  $(p_x, p_y)^j$ ;
•History connection relationship  $\mathcal{V}$  between  $U_i$  and  $S_j$ ;
Output:
• $\mathcal{F}$ : the partition of all nodes into clusters;
• $\mathcal{CH}$ : the set of cluster centers;
1: function KALMANFILTERING $\mathcal{U}$ .
2:   Initialize  $\mathbf{s}_0$  and  $\mathcal{P}_0$ ;
3:   for Each  $a \in [1, L-1]$  do
4:     Time Update according to (19);
5:     Measurement Update according to (20);
6:   end for
7:   Output mobility prediction results:
8:    $\mathbf{s}^i = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_{L-1}^i\}$ .
9: end function
10: function PROBABILITYCOMPUTATION $\mathcal{V}$ 
11:   Build matrix  $V_{i,:}$  and  $V_{:,j}$ ;
12:   Calculate  $w_{i,j}$ ,  $\Omega_i$ , then compute  $p(i, j)$  according to (22);
13:   Calculate  $w'_{j,i}$ ,  $\Omega'_j$ , then compute  $q(j, i)$  according to (23);
14:   Output  $p(i, j)$  and  $q(j, i)$ ,  $\forall U_i \in \mathcal{U}, S_j \in \mathcal{S}$ .
15: end
16: function APCLUSTERING( $p_x, p_y$ ) $^j, \mathbf{s}^i, p(i, j), q(j, i)$ 
17:   Calculate the distances  $d(n, m)$  based on (24);
18:   Calculate the similarities  $s(n, m)$  based on (25);
19:   Initialize  $a(n, m)$  to be zero;
20:   Initialize  $r(n, m)$  according to the type of nodes;
21:   Update the responsibility  $r(n, m)$  based on (26);
22:   Update the availability  $a(n, m)$  based on (27);
23:    $\mathcal{CH} = \{CH \mid CH = \arg_m \max [a(n, m) + r(n, m)]\}$ ;
24:   Calculate  $\mathcal{F}_{CH_n} = \{m \mid CH_m = CH_n\}$ ;
25:    $\mathcal{F} = \{\mathcal{F}_{CH_n} \mid CH_n \in \mathcal{CH}\}$ ;
26:   Output  $\mathcal{CH}$  and  $\mathcal{F}$ 
27: end function

```

ALGORITHM 1: AP-based mobility-aware and sociality-associate clustering algorithm (AP-MASACA).

The smaller the distance is, the bigger the similarity and then, the higher the possibility that  $U_i$  and  $S_j$  belong to a same cluster. Assume the distance between any two different nodes is larger than 1; thus, the maximum value of similarity is set 1 such that  $s(i, i) = s(j, j) = 1$ . Note that  $i, i'$  refer to the subscripts of mobile users and  $j, j'$  are the subscripts of edge servers.

**4.3.2. AP-Based Clustering Mobility-Aware and Sociality-Associate Clustering Algorithm.** During the clustering, we generally use subscripts  $n, m$  to denote all nodes in the system. After similarity measures, there are other two important messages transferred between nodes [28, 29]: (i) responsibility,  $r(n, m)$ , is sent from node  $n$  to candidate exemplar node  $m$ , indicating how well the node  $m$  is suitable as the exemplar of node  $n$ ; (ii) availability,  $a(n, m)$ , is sent from candidate exemplar node  $m$  to node  $n$ , reflecting how appropriate it would be for the node  $n$  choose node  $m$  as its exemplar. The initial values are given according to the social association, i.e., for the case of  $n = i$  and  $m = j$ ; calculate  $r(i, j) = q(j, i)$

and  $a(i, j) = p(i, j)$ ; otherwise, the initial value is  $a(n, m) = 0$ , e.g.,  $a(i, i) = a(i, i') = a(j, j') = a(j, j) = 0$ .

The update formulas for responsibility and availability are stated as

$$r(n, m) \leftarrow s(n, m) - \max_{m' \neq m} a(n, m') + s(n, m'), \quad (26)$$

$$a(n, m) \leftarrow \min_{n \neq m} \left\{ 0, r(n, m) + \sum_{\forall n' \notin \{n, m\}} \max \{0, r(n', m)\} \right\}, \quad (27)$$

$$a(m, m) \leftarrow \sum_{n' \neq m} \max \{0, r(n', m)\}. \quad (28)$$

In order to avoid numerical oscillations, the AP algorithm introduces a convergence coefficient  $\lambda$  when updating information. Each piece of information is set to  $\lambda$  times its previous iteration updating value plus  $1 - \lambda$  times the current information updating value. Among that, the attenuation

TABLE 2: Parameter settings.

Parameters	Values
The initial state vector in Kalman filter $\mathbf{s}^{(0)}$	$(0, 0, 0, \nu)^T$
The number of tasks to be executed	[10, 50]
The processing capacity of MEC servers	10
The transmission power $P_{T_x}$	-10 (dBm)
The standard deviation of shadowing $\sigma$	8 (dB)
The desired SNR $\gamma_{th}$	10 (dB)
The AWGN $N_0$	-95 (dBm)

coefficient  $\lambda$  is a number between 0 and 1, while in our simulation  $\lambda = 0.5$  is used.

$$r_{t+1}(n, m) \leftarrow (1 - \lambda)r_{t+1}(n, m) + \lambda r_t(n, m), \quad (29)$$

$$a_{t+1}(n, m) \leftarrow (1 - \lambda)a_{t+1}(n, m) + \lambda a_t(n, m). \quad (30)$$

Availabilities and responsibilities are combined to select exemplar. The maximum value of  $a(n, m) + r(n, m)$  identifies the exemplar. Upon convergence, each node's cluster head CH is defined by

$$CH = \arg \max_m \{r(n, m) + a(n, m)\} \quad (31)$$

A detailed AP-based clustering algorithm is shown in Algorithm 1.

## 5. QoS-Aware Offloading Scheme

According to the AP clustering results, each MU has a high probability to offload its tasks to the edge servers within a same cluster. During designing the clustering algorithm, we have considered the influence of history connections and the mobility prediction. To realize the objective given in P1, we take two more factors into account, encompassing the service ratio and the channel conditions between MU and edge servers.

Although it is overloaded to make edge servers execute all of computation tasks, edge servers are expected to serve as many mobile users as possible. In addition in most cases, users prefer to communicate with edge servers with more available resources rather than the ones that are trapped in resource competition. So as the available resource of edge servers decreases, their corresponding social strength may be affected to decrease. To visually evaluate the quality of computation offloading based on the AP-MASACA, we defined a QoS parameter average service ratio, which follows

$$\text{Sev}(i) = \frac{n_{\text{sev}_i}}{n_d}, i = 1, 2, \dots, k, \quad (32)$$

$$\overline{\text{Sev}} = \frac{\sum \text{Sev}(i)}{k}, \quad (33)$$

where  $\text{Sev}(i)$  represents the service ratio of  $i$ -th edge server,  $n_{\text{sev}_i}$  represents the actual number of mobile devices attach-

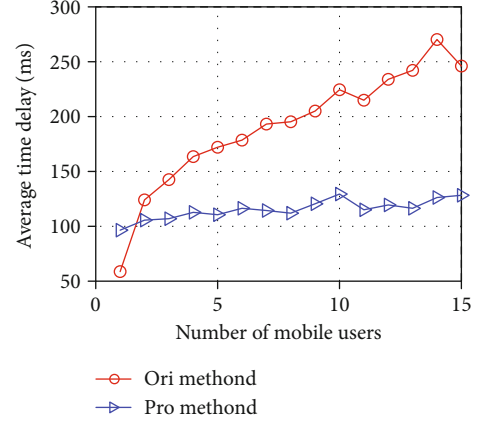


FIGURE 4: The average offloading time delay.

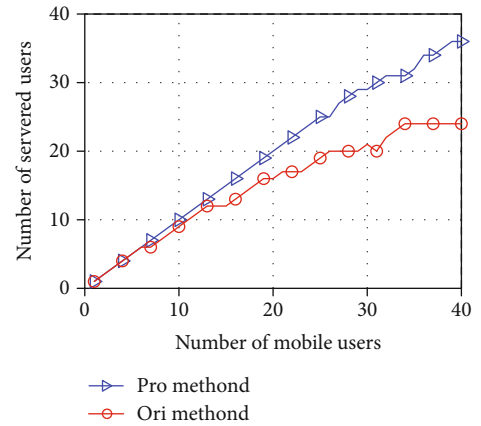


FIGURE 5: The service ratio with different user scales.

ing the  $i$ -th edge server (assuming that every mobile user has only one device), and  $n_d$  represents the actual number of mobile devices served by any edge server.

By applying MASACA strategies, the system cost of computation offloading could be decreased. When signal propagates through wireless channels, it undergoes deleterious effects mainly characterized by path loss, multipath fading, and shadowing, while the computation links between mobile users and edge servers could be interrupted. The SINR outage probability is an important channel quality measure of communication links operating over composite fading/shadowing channels, which is defined as the probability that the received SNR  $\gamma$  drops below a certain threshold  $\gamma_{th}$  for a certain average SNR  $\bar{\gamma}$  [30]:

$$P_{\text{out}}(\bar{\gamma}, r_{th}) = E[\theta(\gamma_{th} - \gamma)]. \quad (34)$$

Note that  $P_{\text{out}}(\bar{\gamma}, r_{th})$  requires the SNR distribution. Considering that the SNR could be affected by the distance of links, SNR  $\gamma(d)$  could be obtained following

$$P_{R_x}(d) = P_{T_x} - L(d) + W, \quad (35)$$

$$\gamma(d) = P_{R_x}(d) - N_0, \quad (36)$$

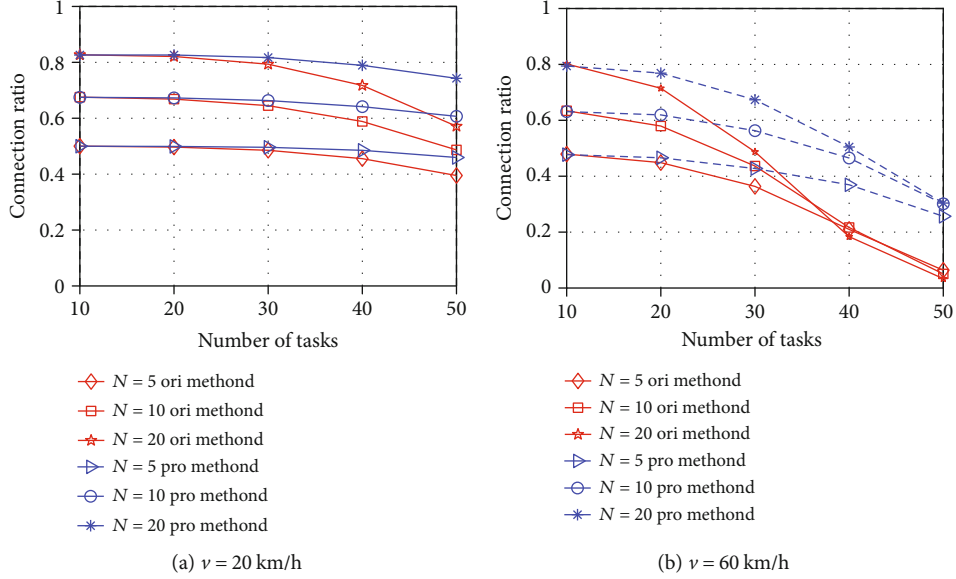


FIGURE 6: The connection ratio vs. number of tasks in single user scenario.

where  $P_{T_x}$  represents the transmission power,  $L(d)$  represents the path loss, and  $W$  represents the shadowing gain factor. And  $N_0$  represents the noise power of additive white Gaussian noise (AWGN) in a wireless communication environment.

Furthermore, according to the composite lognormal/gamma PDF introduced by [31], we could get the final expression about outage probability as follows:

$$f_{\sigma}(\sigma) = \frac{1}{\sqrt{2\pi}\zeta\alpha} \exp\left(-\frac{(\ln(\sigma) - \mu)^2}{2\zeta^2}\right), \quad (37)$$

$$P_{\text{out}}(d) = \frac{1}{2} \left\{ 1 + \operatorname{erf} \frac{\gamma - (P_{R_x} - L(d) - N_0)}{\sqrt{2\sigma^2}} \right\}. \quad (38)$$

We define the connection ratio to visually describe the performance for maintaining the sustainability of connections with edge servers in different cases:

$$P_{\text{con}}(d) = 1 - P_{\text{out}}(d). \quad (39)$$

## 6. Experiments and Result Analysis

In this section, we conduct simulation experiments to verify the efficiency of our proposed offloading scheme by comparing with an existing method employed in [32, 33] where mobile users randomly offload computation tasks to edge servers they have searched.

In our simulation, MEC servers are uniformly distributed within an area with side length of 200 m. The simulation settings are listed in Table 2. Under different numbers of MEC servers  $N$  and different moving speed of mobile users  $v$ , we conduct experiments to show the performance in terms of the time delay, the service ratio, and the connection ratio. The results are averaged over 1000 independent runs.

Figure 4 shows the influence of the number of mobile users on the average time delay. We can see that the average offloading time delay increases with the expansion of mobile users' scale for each method. This is an intuitive result because the computation capability of MEC servers is limited and more users need to wait for a specific period before being serviced. For the same number of mobile users, our proposed scheme obtains a lower average delay, outperforming the original random offloading method. This demonstrates the effectiveness of our proposed scheme.

Figure 5 shows that the number of mobile users serviced by edge servers will increase with the expansion of mobile users' scales. Compared with the case that mobile users offload computation tasks randomly, we can see the improvements that applying AP-MASACA could bring more chances for mobile users to be served. At the same time, it can be seen that the system performance was not greatly stable as the number of mobile users increases, which means that there may be available resources of some edge servers not fully utilized and the offloading strategy can be further optimized.

Figure 6 shows the connection ratio under different velocities in the single-user scenario. We can see that the connection ratio of each method declines with the increase of the number of computation tasks. This is because the competition for resources becomes more intense with the increase of task amount. Meanwhile, the decline rate of our proposed scheme is slower than that of the original random case. For the same number of computation tasks, our proposed scheme has a higher connection ratio than the original random method. This demonstrates our proposed scheme can improve the sustainability of connections between MU and MEC servers. Moreover, it is obvious that the larger the number of MEC server,  $N$ , is, the higher the connection ratio. By comparing the results shown in Figures 6(a) and 6(b), we can know that the connection ratio decreases with the increasing speed. This is because the distance between

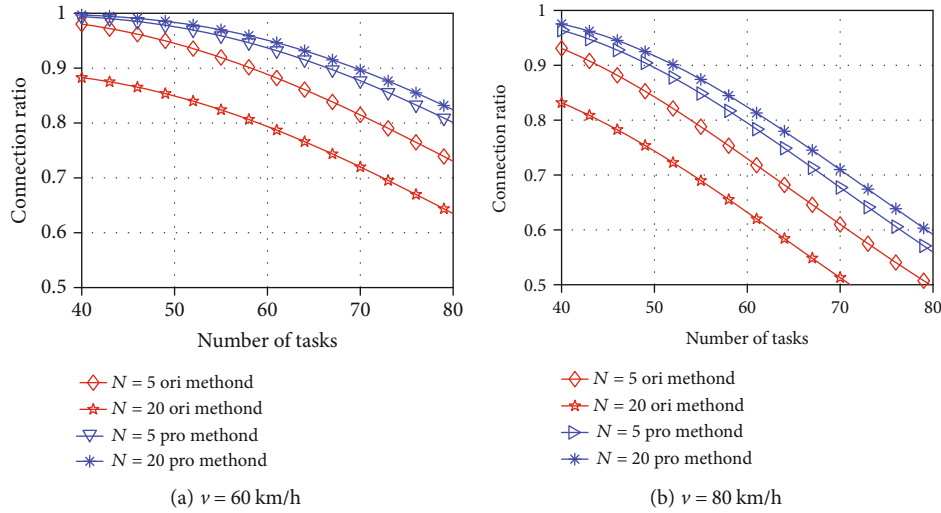


FIGURE 7: The connection ratio vs. number of tasks in multiuser scenario.

the MU and the selected MEC servers changes faster with the increasing speed.

Figure 7 shows the connection ratio of multiuser scenario. As the number of MEC servers increases, the system performance is getting better. And it shows that the performance in the multiuser scenario is better than that in the single-user scenario. Figure 6 shows the connection ratio of multiuser when the moving speed is 80 km/h. From the simulation results, it has been shown that the connection ratio is improved by applying the proposed method especially when mobile users move at a high speed.

## 7. Conclusion

In this paper, we design a task offloading strategy by considering the mobility, the sociality, and QoS factors. First, we employ the Kalman filtering algorithm to do mobility prediction of mobile users' trajectory; then, we compute the sociality association parameter between mobile users and edge servers based on the history connection relationship. By applying the mobility prediction results and the social association parameter as input parameters of affinity propagation algorithm, we propose the mobility-aware and sociality-associate clustering algorithm (MASACA). By applying the clustering results as candidate edge server set, we devise QoS utility function of a given edge server based on the available service ratio and the quality of channel link. At last, based on the designed QoS utility function, we select the candidate edge server with the highest QoS utility as the target edge server to offload tasks. Numerical simulation results demonstrate that our offloading strategy can enhance the data processing capability of power-constrained networks and cut down the computation delay.

## Data Availability

In the simulation section, we use the social network data "usense" provided at <http://www.crowdad.org/copelabs/usense/20170127/index.html>. In the data set, we employ the

distances between users, the encounter durations, and other related data as the input parameters of our proposed clustering algorithm.

## 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 and the Key Project of the National Natural Science Foundation of China under Grant 61931001.

## References

- [1] 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.
- [2] Y. Huo, C. Meng, R. Li, and T. Jing, "An overview of privacy preserving schemes for industrial internet of things," *China Communications*, vol. 17, no. 10, pp. 1–18, 2020.
- [3] 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.
- [4] M. Patel, B. Naughton, and C. Chan, "Mobile-edge computing introductory technical white paper," *Mobile-Edge Computing (MEC) Industry Initiative*, pp. 1089–7801, 2014.
- [5] F. Vhora and J. Gandhi, "A comprehensive survey on mobile edge computing: challenges, tools, applications," in *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 49–55, Erode, India, March 2020.
- [6] Z. He, Z. Cai, S. Cheng, and X. Wang, "Approximate aggregation for tracking quantiles and range countings in wireless sensor networks," *Theoretical Computer Science*, vol. 607, no. 3, pp. 381–390, 2015.

- [7] 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.
- [8] Y. Ding, C. Liu, K. Li, Z. Tang, and K. Li, "Task offloading and service migration strategies for user equipments with mobility consideration in mobile edge computing," in *2019 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BD-Cloud/SocialCom/SustainCom)*, pp. 176–183, Xiamen, China, December 2019.
- [9] Y. Zhang, D. Niyato, and P. Wang, "Offloading in mobile cloudlet systems with intermittent connectivity," *IEEE Transactions on Mobile Computing*, vol. 14, no. 12, pp. 2516–2529, 2015.
- [10] N. Ahmed and B. K. Bhargava, "Towards dynamic QoS monitoring in service oriented architectures," in *Proceedings of the 5th International Conference on Cloud Computing and Services Science - CLOSER*, pp. 163–171, Lisbon, Portugal, 2015.
- [11] S. Guo, B. Xiao, Y. Yang, and Y. Yang, "Energy-efficient dynamic offloading and resource scheduling in mobile cloud computing," in *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pp. 1–9, San Francisco, CA, USA, April 2016.
- [12] C. A. Kamienski, F. F. Borelli, G. O. Biondi, I. Pinheiro, I. D. Zyrianoff, and M. Jentsch, "Context design and tracking for IoT-based energy management in smart cities," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 687–695, 2018.
- [13] Z. He, Z. Cai, J. Yu, X. Wang, Y. Sun, and Y. Li, "Cost-efficient strategies for restraining rumor spreading in mobile social networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2789–2800, 2017.
- [14] J. Li, S. Cheng, Y. Li, and Z. Cai, "Approximate holistic aggregation in wireless sensor networks," in *2015 IEEE 35th International Conference on Distributed Computing Systems*, pp. 740–741, Columbus, OH, USA, June 2015.
- [15] M. Mehrabi, H. Salah, and F. H. P. Fitzek, "A survey on mobility management for MEC-enabled systems," in *2019 IEEE 2nd 5G World Forum (5GWF)*, pp. 259–263, Dresden, Germany, September 2019.
- [16] S. Wang, J. Xu, N. Zhang, and Y. Liu, "A survey on service migration in mobile edge computing," *IEEE Access*, vol. 6, pp. 23511–23528, 2018.
- [17] C.-L. Wu, T.-C. Chiu, C.-Y. Wang, and A.-C. Pang, "Mobility-aware deep reinforcement learning with glimpse mobility prediction in edge computing," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pp. 1–7, Dublin, Ireland, June 2020.
- [18] J. Xu, Y. Sun, L. Chen, and S. Zhou, "E2M2: energy efficient mobility management in dense small cells with mobile edge computing," in *2017 IEEE International Conference on Communications (ICC)*, pp. 1–6, Paris, France, May 2017.
- [19] 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.
- [20] C.-M. Huang, C.-H. Shao, S.-z. Xu, and H. Zhou, "The social internet of thing (S-IOT)-based mobile group handoff architecture and schemes for proximity service," *IEEE Transactions on Emerging Topics in Computing*, vol. 5, no. 3, pp. 425–437, 2017.
- [21] 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, July 2019.
- [22] C.-H. Wang, J. J. Kuo, D. N. Yang, and W. T. Chen, "Collaborative social Internet of Things in mobile edge networks," *IEEE Internet of Things Journal*, vol. 7, no. 12, pp. 11473–11491, 2020.
- [23] H. Zhou, L. Xu, and J. Zhang, "A physical-social-based group utility maximization framework for cooperative caching in mobile networks," in *2018 10th International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 1–7, Hangzhou, China, October 2018.
- [24] D. Belli, S. Chessa, L. Foschini, and M. Girolami, "A social-based approach to mobile edge computing," in *2018 IEEE Symposium on Computers and Communications (ISCC)*, pp. 00 292–00 297, Natal, Brazil, June 2018.
- [25] 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.
- [26] J. Jung, S. Chun, X. Jin, and K.-H. Lee, "Quantitative computation of social strength in social internet of things," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 4066–4075, 2018.
- [27] Y. Wang, M. Ding, Z. Chen, and L. Luo, "Caching placement with recommendation systems for cache-enabled mobile social networks," *IEEE Communications Letters*, vol. 21, no. 10, pp. 2266–2269, 2017.
- [28] Q. Gao, Y. Wang, X. Cheng, J. Yu, X. Chen, and T. Jing, "Identification of vulnerable lines in smart grid systems based on affinity propagation clustering," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5163–5171, 2019.
- [29] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 12, pp. 927–976, 2007.
- [30] F. Yilmaz, "On the relationships between average channel capacity, average bit error rate, outage probability, and outage capacity over additive white gaussian noise channels," *IEEE Transactions on Communications*, vol. 68, no. 5, pp. 2763–2776, 2020.
- [31] M.-J. Ho and G. Stuber, "Co-channel interference of microcellular systems on shadowed Nakagami fading channels," in *IEEE 43rd Vehicular Technology Conference*, pp. 568–571, Secaucus, NJ, USA, 1993.
- [32] Y. Liu, S. Wang, J. Huang, and F. Yang, "A computation offloading algorithm based on game theory for vehicular edge networks," in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–6, Kansas City, MO, USA, May 2018.
- [33] G. Fan, W. Bai, X. Gan, X. Wang, and J. Wang, "Social-aware content sharing in D2D communications: an optimal stopping approach," in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–6, Kansas City, MO, USA, May 2018.

## Research Article

# Privacy-Aware Online Task Offloading for Mobile-Edge Computing

Dali Zhu,<sup>1,2</sup> Ting Li ,<sup>1,2</sup> Haitao Liu ,<sup>1,2</sup> Jiyan Sun,<sup>1</sup> Liru Geng,<sup>1</sup> and Yinlong Liu <sup>1,2</sup>

<sup>1</sup>*Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China*

<sup>2</sup>*School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100049, China*

Correspondence should be addressed to Yinlong Liu; [liuyinlong@iie.ac.cn](mailto:liuyinlong@iie.ac.cn)

Received 1 December 2020; Revised 1 April 2021; Accepted 18 May 2021; Published 11 June 2021

Academic Editor: Yan Huang

Copyright © 2021 Dali Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Mobile edge computing (MEC) has been envisaged as one of the most promising technologies in the fifth generation (5G) mobile networks. It allows mobile devices to offload their computation-demanding and latency-critical tasks to the resource-rich MEC servers. Accordingly, MEC can significantly improve the latency performance and reduce energy consumption for mobile devices. Nonetheless, privacy leakage may occur during the task offloading process. Most existing works ignored these issues or just investigated the system-level solution for MEC. Privacy-aware and user-level task offloading optimization problems receive much less attention. In order to tackle these challenges, a privacy-preserving and device-managed task offloading scheme is proposed in this paper for MEC. This scheme can achieve near-optimal latency and energy performance while protecting the location privacy and usage pattern privacy of users. Firstly, we formulate the joint optimization problem of task offloading and privacy preservation as a semiparametric contextual multi-armed bandit (MAB) problem, which has a relaxed reward model. Then, we propose a privacy-aware online task offloading (PAOTO) algorithm based on the transformed Thompson sampling (TS) architecture, through which we can (1) receive the best possible delay and energy consumption performance, (2) achieve the goal of preserving privacy, and (3) obtain an online device-managed task offloading policy without requiring any system-level information. Simulation results demonstrate that the proposed scheme outperforms the existing methods in terms of minimizing the system cost and preserving the privacy of users.

## 1. Introduction

In the recent years, with the advent of 5G network, as well as the fast popularization of mobile devices, a myriad of new applications is emerging, such as augmented reality (AR)/virtual reality (VR) [1, 2], online 3D games [3], and connected cars [4]. Specifically, the recent Cisco Annual Internet Report expects that the number of global mobile devices will grow from 8.8 billion in 2018 to 13.1 billion by 2023 and the vast majority of mobile data traffic (99%) will originate from these mobile devices [5]. However, due to their limited computing units and battery energy, mobile devices struggle to resist to such traffic explosion and become unable to meet the stringent requirements of computing-demanding and latency-sensitive applications.

To get rid of such limitations, a novel paradigm of mobile edge computing (MEC) [6] is proposed as an extension of

remote-centralized clouds [7] by the European Telecommunications Standards Institute (ETSI). The key idea beneath MEC is to deploy computing and storage resources from the core network to the radio access network (RAN) in the fifth generation (5G) networks [8]. In such computing paradigm, computation tasks will be offloaded to nearby MEC servers via wireless channels by mobile devices, which can meet the requirements of computing intensive applications and achieve ultrashort processing latency.

Despite the benefits, MEC still has shortcomings in terms of security and privacy leakage [9]. For example, the location privacy and usage pattern privacy problem [10] are investigated in this paper, which are related to the MEC task offloading feature. Intuitively, when a mobile device is to obtain optimal offloading performance, it tends to offload all its tasks to the MEC server. Accordingly, an honest-but-curious MEC server can infer the location privacy and usage

pattern privacy of users who are privacy sensitive, which may prevent these users from accessing the MEC system if not properly addressed. Although these two privacy issues have been extensively studied in other fields, one challenge still needs to be addressed in MEC systems, which is how to protect both the location privacy and usage pattern privacy while minimizing the delay and energy consumption cost.

Most existing task offloading schemes that want to achieve optimal system performance, such as [11, 12], largely ignore these privacy problems. And current privacy-preserving techniques of cloud computing are not always applicable for the MEC system, such as the works in [13, 14]. Therefore, the more challenging problem is how to prevent unintentional leakage of user's privacy while still maintaining the optimal delay and energy consumption performance. The most related works probably are [10, 15], which studied the optimization of delay and energy consumption cost while considering both location privacy and usage pattern privacy. The former scheme formulates this problem as a constrained Markov decision process (CMDP), and the latter one applies a Dyna-Q architecture based on the CMDP to achieve a better privacy-aware offloading policy. However, both of them are system-level solutions. They rely on the assumption of the wireless channel power gain that is formulated as a Markov chain model, in which some system-side information should be known in advance. Such assumption is not relevant to infrastructure-free scenarios, such as individual combat in military scenarios, forest fire rescue [16], and heterogeneous IoT [17] which are more applicable user-level schemes.

In order to minimize system cost (e.g., latency and energy consumption) and protect user's privacy without requiring any system-level information as a prior knowledge, we propose a device-level and privacy-preserving task offloading scheme for the MEC system. This scheme is based on a semiparametric contextual multi-armed bandit (MAB) problem, which can address the trade-offs inherent in the sequential decision problem and overcome the challenges of lacking system-side information. To the best of our knowledge, this user-level scheme is the first to be proposed to solve the privacy problem of MEC. The location privacy and usage pattern privacy of users will be preserved in this paper. An online-learning algorithm will be proposed to make adaptive task offloading decisions under dynamic network environment. The main contributions of this paper are summarized as follows:

- (1) *MAB-based problem modelling.* We study a joint optimization problem of task offloading and privacy preservation in the MEC system. And then, this problem is transformed as a semiparametric contextual MAB problem to overcome the challenge of unknown network dynamics, which can utilize the contextual feature vector to describe user-side information for decoupling the time dependency
- (2) *Privacy-aware optimal offloading decision.* We propose a privacy-aware online learning algorithm, called PAOTO (privacy-aware online task offloading), to make device-level task offloading decisions

while protecting user's location privacy and usage pattern privacy. By utilizing the transformed Thompson sampling (TS) architecture, we can make adaptive task offloading decisions at the user-side perspective

- (3) *Extensive simulation-based performance evaluation.* We carry out simulations to demonstrate the effectiveness of the proposed algorithm. The results show that the PAOTO algorithm performs close-to-optimal and far better than the newly proposed Dyna-Q algorithm in [15].

The remaining parts of this paper are organized as follows. In Section 2, we discuss the motivation and related works. We will formally describe the system model and problem formulation in Section 3. Next, we present the algorithm design and simulation evaluation in Sections 4 and 5, respectively. Finally, we draw some conclusions and highlight the direction for future work in Section 6.

## 2. Motivation and Related Work

*2.1. Motivation.* The problem of determining the privacy-aware and user-level task offloading decisions for mobile devices requires solving two important challenges: (1) how to best prevent leakage of user's privacy while still maintaining the optimal delay and energy consumption performance and (2) how to design a task offloading policy that can online determine the optimal execution platform (i.e., local processing unit, MEC servers, or buffer) for users at the user-side perspective?

To address the first challenge, we propose a privacy metric to jointly quantify the location and usage pattern privacy and utilize a semiparametric MAB to incorporate the privacy metric into the performance model. This can strike a balance between the privacy-preserving level and system cost (e.g., processing latency and energy consumption cost). Previous works require system-level information to design an optimal task offloading strategy, but this is not applicable to infrastructure-free scenarios (e.g., individual combat in military scenarios, forest fire rescue and heterogeneous IoT). We address this second challenge by utilizing the contextual feature vector in the contextual MAB model to describe user-side information and applying the Thompson sampling (TS) algorithm to estimate and learn the performance model based on the contextual information.

*2.1.1. Characterizing Privacy Metric.* Recently, MEC has been increasing in popularity but issues relating to the security and privacy in the MEC system still has shortcomings. On one hand, some security issues such as authentication, private data storage, and intrusion detection have received attentions but these security issues are inherited from the conventional cloud computing framework and are less relevant to the key technologies in the MEC system. On the other hand, based on the simulation results and considered setting in [10], we can find that the privacy problems relating to MEC unique wireless task offloading technology remains less explored, which are user's location privacy and usage pattern privacy.

According to [10], the offloading pattern can be observed as a mobile device may offload all its tasks to the MEC server when the wireless channel state is good, while it will not offload any tasks otherwise. Accordingly, the honest but curious MEC server (it may be controlled by adversary) can be based on the offloading pattern and historical statistics to obtain the number of tasks offloaded in each period. Hence, the wireless channel condition (it is only assumed as good or bad and can be extended to the multistate case) and user's actual usage pattern can be inferred by adversary. Specifically, the wireless channel condition is highly related to the distance between the user and the MEC server. If a mobile user communicates with multiple MEC servers, the location privacy may be inferred by these MEC servers based on the surveillance of the wireless channel state. The user's location privacy is leaked. Moreover, when someone's office is near the AP, its wireless channel state may be always good and it may always offload all its tasks to the MEC server. Thus, the adversary can obtain the total number of tasks offloaded that is determined by the user's actual device usage pattern. The user's usage pattern privacy may be leaked.

Particularly, it is very important for privacy-sensitive users to solve the problem of leaking location privacy and usage pattern privacy that are induced into the unique wireless task offloading feature in the MEC system. If they are not properly addressed, it may prevent these privacy-sensitive users from accessing the MEC system. Significantly, although these two privacy problems have already been studied in other system, protecting user's location privacy and usage pattern privacy while minimizing delay and energy consumption cost in the MEC system still poses a critical challenge.

Therefore, to address this challenge, it is desirable to design a metric to jointly quantify the location and usage pattern privacy. Next, we formulate the task offloading and privacy preservation problem as a contextual MAB problem with a semiparametric reward model based on processing latency, energy consumption cost, and this privacy metric. This is aiming to strike a balance between the privacy-preserving level and system cost. That is, according to problem formulation and proposed algorithm, we can obtain the optimal delay and energy consumption performance while protecting user's location and usage pattern privacy, which can be seen in Sections 3 and 4.

*2.1.2. User-Level Task Offloading.* With mobile data traffic growing explosively, the mobile devices with limited resources cannot meet the stringent requirements of computing-demanding and latency-sensitive applications. Therefore, designing a desirable task offloading strategy of the MEC system has attracted tremendous attention in the industry and academia. This strategy can determine the optimal task execution platform for the user, executing in the local processing unit, offloading to MEC server or queueing in the buffer.

Many previous works (e.g., [10, 15]) on task offloading generally assume that the system-side information is always available. Such assumption is more applicable to the infrastructure-assisted edge computing scenarios where the

infrastructure (e.g., an access point or base station) is available for obtaining system-side information in advance [18]. However, some infrastructure-free scenarios, such as individual combat in military scenarios, forest fire rescue and heterogeneous IoT, are not suitable for previous system-level solutions, because these mobile devices in infrastructure-free scenarios are operating in a scattered manner and the system-side network information is missing for them. Especially, if they want to explore system-level information in advance, it may cause additional system cost, such as scarce bandwidth usage and additional energy consumption cost.

In this case, it is desirable to design a user-level task offloading strategy for overcoming the challenge of lacking the system-side information. In response to the challenge that the system-level information may not be readily available in some infrastructure-free scenarios, we propose an online task offloading scheme at the user perspective. In this scheme, the user-side information will be described as contextual feature vector and the Thompson sampling (TS) algorithm will be applied to estimate and learn the performance model based on the contextual information. It can adaptively decide where to execute the offloaded task for the mobile user without any system level information. It can be seen in Section 4 for details.

*2.2. Related Work.* In recent years, the task offloading strategies have attracted significant efforts to minimize total delay and energy consumption cost in MEC systems. For example, Xu et al. proposed an online algorithm based on Lyapunov optimization and Gibbs sampling, which jointly optimized dynamic service caching and task offloading to reduce computation latency while keeping energy consumption low [19]. Wei et al. studied the problem of task offloading and channel resource allocation based on MEC in 5G ultra-dense networks (UDN) [20]. The authors formulated task offloading as an integer nonlinear programming problem and proposed an efficient task offloading and channel resource allocation scheme based on differential evolution algorithm. Dab et al. proposed a joint radio resource allocation and task assignment strategy based on a Q-learning algorithm to minimize the energy consumption cost under both the latency and device's computation resource constraints [21]. Li and Cai discussed the incentive mechanism design for collaborative task offloading in the MEC network [22]. They proposed an online truthful mechanism integrating computation and communication resource allocation to address social welfare maximization problem by considering each task's specific requirements in terms of data size, delay, and preference. However, none of the works mentioned above considered user's privacy issues.

There are a few works considering both task offloading and privacy preservation. For example, He et al. identified a new privacy vulnerability caused by the wireless offloading feature of MEC-enabled IoT. To address this vulnerability, the authors developed an offloading strategy for MEC-enabled IoT, which can learn a good offloading strategy while protecting the devices' location privacy [23]. However, the extra prior information was required. In [24], Zhang et al. proposed a strategy that can achieve an efficient task



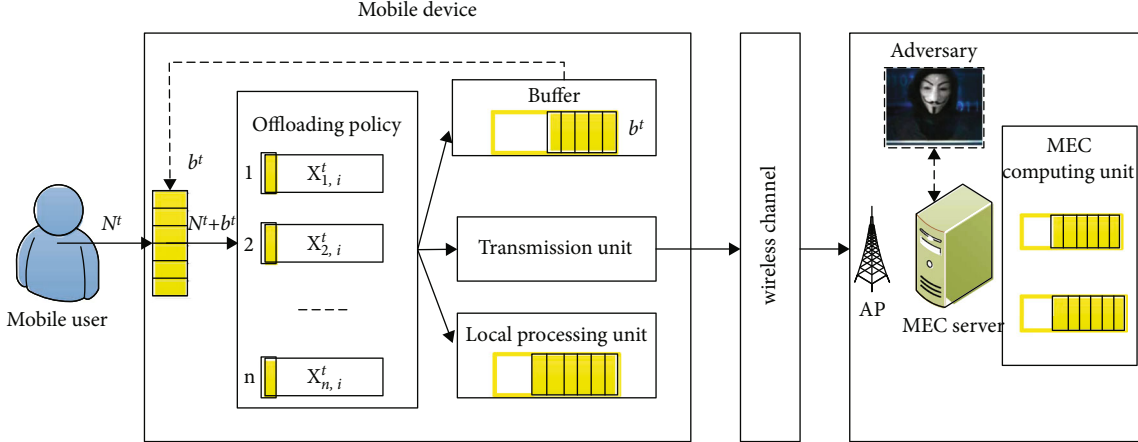


FIGURE 1: An illustration of task offloading in the MEC system.

scheduling policy on edge while ensuring privacy. In [25], Zhou proposed a novel context-aware task allocation framework for mobile crowdsensing in the scenario of edge computing. The task allocation was performed in both the cloud computing layer and the edge computing layer. In the cloud layer, authors proposed a privacy-preserving and contextual online learning algorithm to manage the participants' reputation. But this scheme was implemented at the system level and required a priori network information.

Besides, He et al. identified location privacy and usage pattern privacy issues, which are induced by the wireless task offloading feature of MEC [10]. To address these privacy issues, authors proposed a constrained Markov decision process- (CMDP-) based privacy-aware task offloading scheduling algorithm to achieve the best possible system performance while protecting user's privacy. Min and Wan proposed a reinforcement learning- (RL-) based privacy-aware offloading scheme, which enables the IoT device to make the task offloading decisions and protect both the user location privacy and the usage pattern privacy for the MEC system [15]. Nevertheless, both the works were implemented at the system level. They all need to explore system-level information in advance, which is difficult to obtain and may cause additional system cost, such as scarce bandwidth usage and energy consumption of network devices.

In general, none of the aforementioned works consider both task offloading and privacy protection problem at the user level. These aforementioned studies mainly face two challenges. First, they only consider the simple task offloading strategies for minimizing total delay and energy consumption cost in MEC systems. However, the privacy issues related to the task offloading pattern were ignored in their works, which may be very important for the privacy-sensitive users. Second, some works considering both task offloading and privacy preservation were all implemented at the system level. That is, these works generally assume that the system-side information is always available. However, this is applicable for the infrastructure-assisted scenarios where the infrastructure (e.g., an access point (AP)) is available for obtaining system-side information in advance. For the infrastructure-free scenarios (e.g., individual combat in

military scenarios, forest fire rescue, and heterogeneous IoT), these mobile devices will operate in a decentralized manner and the system-side information is difficult to obtain and may cause additional system cost.

To conquer this challenge, we propose a novel privacy-aware task offloading scheme based on an online learning algorithm that just requires device-level information and it can achieve the best possible system performance while protecting the user's privacy.

### 3. System Model and Problem Formulation

**3.1. System Model.** In this section, the task offloading model will be presented. As illustrated in Figure 1, we consider a scenario in which the mobile user/device communicates with the MEC server through the access point (e.g., Wi-Fi or 5G base station) via the wireless channel. For ease of exposition, the bandwidth constraint of the wireless channel is not considered in this paper and we will consider it in the next work. The mobile device has computing-intensive computation tasks that are required to be completed as soon as possible. Due to its limited battery energy and computing capabilities, the mobile device can offload some computation tasks to the MEC server, which has powerful computing capabilities. As such, the mobile device has three ways to process these computation tasks, that is, computing in the local processing unit, offloading to the MEC server through the transmission unit, and queuing in the buffer for processing in the next time slot.

Without loss of generality, we assume that the task offloading policies are made in a slotted structure and its timeline is discretized into time slots  $t \in \mathcal{T} = \{1, 2, \dots, T\}$ . At each time slot  $t$ , the mobile user will newly generate  $N^t$  computation tasks to the mobile device, denoted by a set of  $\mathbb{N} = \{1, 2, \dots, N_{\max}^t\}$  ( $N_{\max}^t$  is the maximum possible number of generated tasks), which depends on the user's usage pattern. And the  $b^t \in \mathcal{B} = \{1, 2, \dots, b_{\max}\}$  (with the maximum buffer size  $b_{\max}$ ) can be denoted as the number of tasks in the buffer at time slot  $t$ .

A widely used three-parameter model [26] can be used to describe each task  $n$ , denoted by a set of  $\mathcal{N}^t = \{1, 2, \dots, (N^t$

$+ b^t\}$ . The three-parameter model consists of the input data size  $\lambda_n$  (bits), computation intensity  $\delta_n$  (CPU cycles/bit), and maximum allowed latency  $\tau_n$  (seconds). Whereupon, the computation demand for each task can be obtained by  $m_n = \lambda_n * \delta_n$  (CPU cycles). In each time slot  $t$ , all the  $(N^t + b^t)$  tasks (including the newly generated  $N^t$  tasks and the  $b^t$  tasks in the buffer) will be either locally executed, buffered, or remotely offloaded according to the proposed task offloading policy. More specifically, the mobile device will explore the optimal offloading policy for each task.

Based on [10], in order to minimize computing delay and energy consumption, the mobile device tends to offload all its tasks to the MEC server if the wireless channel state is good and processes all its tasks locally if the wireless channel state is bad. Under such circumstances, the user's location and usage pattern privacy are easily spied by the attacker. Hence, the proposed task offloading policy in this paper takes the privacy preservation into account. And the wireless channel power gain will not be assumed as the Markov model, which allows the proposed task offloading policy to be executed on the device level. More explicitly, the mobile device can only observe its local information (e.g., the number of tasks and the computation demand of each task) but the system-side information is not observable. Key parameter notations in this paper are listed in Table 1 for ease of reference.

**3.2. Problem Formulation.** We focus on privacy-aware and user-level task offloading optimization problems in this paper. In this section, we firstly formulate the task offloading decision making and then the model system cost (including processing latency and energy consumption cost) and privacy level as the performance metrics. Finally, the objective function will be presented.

**3.2.1. Task Offloading Decision Making.** To maintain satisfactory quality of service, an available and reliable task offloading policy should be considered. And the tasks can be dynamically offloaded to the three different positions  $i$  by the mobile device, denoted by  $\mathcal{F} = \{l, s, b\}$ , where  $l$ ,  $s$ , and  $b$  represent the local processing unit, MEC server, and buffer, respectively. At each time slot  $t$ , the mobile device (also called the earner or operator) makes the task offloading decision for each task  $n$ . Here, we design a binary indicator  $x_{n,i}^t$  to denote the dynamic task offloading decision variable; let  $x_{n,i}^t = 1$  if the task  $n \in \mathcal{N}^t$  is offloaded to platform  $i \in \mathcal{F}$  at time slot  $t$  and  $x_{n,i}^t = 0$  otherwise. Note that at a given time slot  $t$ , each task  $n$  can be offloaded to only one execution platform ( $l$ ,  $s$ , or  $b$ ). We have the following constraints for  $x_{n,i}^t$ :

$$x_{n,i}^t \in \{0, 1\}, \quad \forall t, n, i, \quad (1)$$

$$x_{n,l}^t + x_{n,s}^t + x_{n,b}^t = 1, \quad \forall n, t, \quad (2)$$

$$\sum_{n \in (\mathcal{N}^t \cup \mathcal{B})} \sum_{i \in \mathcal{F}} x_{n,i}^t = N^t + b^t, \quad \forall t. \quad (3)$$

Equation (1) indicates that whether offloading the task  $n$  in platform  $i$  in the time slot  $t$ . Equation (2) indicates that

TABLE 1: List of main parameter notations.

Notation	Definition
$N^t$	The number of newly generated tasks at time slot $t$
$n \in \mathcal{N}^t$	Tasks to be offloaded
$\lambda_n$	Input data size of task $n$ (bits)
$\delta_n$	Computation intensity of task $n$ (CPU cycles/bit)
$m_n$	Computation demand for task $n$ (CPU cycles)
$b^t \in \mathcal{B}$	Tasks in the buffer
$i \in \mathcal{F} = \{l, s, b\}$	Offloading execution platforms (local, MEC, and buffer)
$\gamma_i$	Available computing capability in execution platform $i$
$d_{n,i}^t$	Processing delay of each task
$D^t$	Total processing delay for processing all tasks
$e_{n,i}^t$	Energy consumption of each task
$E^t$	Total energy consumption
$q^t$	The number of tasks offloaded to the MEC server
$\xi$	Weighting factor of location privacy
$K$	The metric of the usage pattern privacy
$\Delta^t$	The difference between $N^t$ and $q^t$
$P^t$	Privacy metric

only one of  $x_{n,l}^t$ ,  $x_{n,s}^t$ , and  $x_{n,b}^t$  for task  $n$  in the time slot  $t$  can be nonzero. Equation (3) indicates that the  $(N^t + b^t)$  tasks will be offloaded to execution platform  $i$  in the time slot  $t$ . Based on the above definition, the system cost model (including processing latency and energy consumption) and privacy model will be further described.

**3.2.2. System Cost Model.** Similar to [27, 28], we consider a system cost that accounts for processing latency and energy consumption cost, which are associated with the task offloading. They depend on both the tasks and the processing platforms where the tasks are computed.

**(1) Processing Latency.** In this paper, total processing latency consists of three parts, i.e., queuing delay in buffer, computing delay in either the local processing unit or the MEC server. For ease of exposition, we assume that the queuing delay in the buffer can be converted to computing delay and the buffer can be treated as a microprocessor, which has much lower computing capability than the local processing unit. In our system, each task  $n \in (\mathcal{N}^t \cup \mathcal{B})$  will be offloaded to execution platform  $i \in \mathcal{F}$  by the mobile device in time slot  $t$ . We use  $\gamma_i$  to denote the available computing capability (i.e., CPU cycles per second) of execution platform  $i$  for task processing at time slot  $t$ . Then, the processing delay  $d_{n,i}^t$  of each task  $n$  can be expressed as follows:

$$d_{n,i}^t = \frac{m_n^t}{\gamma_i}, \quad (4)$$

where  $m_n^t$  is the computation demand of each task  $n$  at time slot  $t$ . Therefore, given the task offloading decision  $x_{n,i}^t$ , the total processing latency required to process  $(N^t + b^t)$  tasks within time slot  $t$  can be further expressed as follows:

$$D^t = \sum_{n=0}^{N^t+b^t} \sum_{i \in \mathcal{F}} d_{n,i}^t x_{n,i}^t = \sum_{n=0}^{N^t+b^t} \sum_{i \in \mathcal{F}} \frac{m_n^t}{\gamma_i} x_{n,i}^t. \quad (5)$$

(2) *Energy Consumption.* Task offloading will consume the energy of the mobile device, whose battery storage capacity is rather limited. Thereby, further investigation on how to minimize the total energy consumption of the mobile device is one of the objectives of this paper. The energy consumption cost of the mobile device may include the CPU cycles, transmitting energy and electric energy. They are associated with the tasks executed in the local processing unit, offloaded to the MEC server, and queued in the buffer. To better characterize these energy consumption costs, we let the  $e_{n,i}^t$  be the energy consumption in time slot  $t$  for offloading task  $n$  to execution platform  $i$  ( $i \in \mathcal{F} = \{b, l, s\}$ ). Thus, when considering the task offloading decision  $x_{n,i}^t$ , the overall energy consumption at time slot  $t$  can be expressed as follows:

$$E^t = \sum_{n=0}^{N^t+b^t} \sum_{i \in \mathcal{F}} e_{n,i}^t x_{n,i}^t, \quad (6)$$

3.2.3. *Privacy Model.* As more and more people enjoy the benefits of MEC, the location privacy and usage pattern privacy of MEC have become a major concern. According to the simulation results and considered setting in [10], we can observe the offloading pattern that the mobile device may offload all its tasks to the MEC server when the wireless channel state is good, while it will not offload any tasks otherwise. For simplicity, it is assumed that the wireless channel states are only good and bad in this work. It can be extended to the multistate case. The wireless channel gain is highly related to the distance between the user and the MEC server. Thus, the honest-but-curious MEC server (it may be controlled by adversary) can infer not only the wireless channel state but also the distance to the mobile device based on the offloading pattern and historical statistics.

Accordingly, when the mobile device communicates with multiple MEC servers, its location information may be jointly inferred by these MEC servers. Besides, if a mobile device always maintains a good channel state (e.g., its office near the base station), it will always offload all its tasks to the MEC server. The total number of tasks is highly related to the user's usage pattern (i.e., user's app running if a certain pattern exists in the number of tasks generated by the app), which may be very important for the privacy-sensitive users. Hence, the MEC server may be able to infer the personal information of the user through monitoring the total number of offloading tasks and analyzing the historical statistics.

Hence, from the privacy perspective, we propose a metric to jointly quantify the location and usage pattern privacy and strike a balance between the privacy-preserving level and system cost. Firstly, the total number of tasks offloaded to the

MEC server at the end of time slot  $t$  is defined as  $q^t$  and we have

$$q^t = \sum_{n=0}^{N^t+b^t} x_{n,s}^t, \quad \forall t. \quad (7)$$

Then, the privacy metric of  $P^t$  can be obtained by

$$P^t = \mathbb{1}(\Delta^t = 0) \cdot K + \mathbb{1}(\Delta^t \neq 0) \cdot \left[ \mathbb{1}(q^t = 0) \cdot \xi + \mathbb{1}(q^t \neq 0) \cdot \frac{\hat{\xi}}{\Delta^t} \right], \quad (8)$$

where the  $\mathbb{1}$  represents the indicator function that equals 1 if the statement is true and 0 otherwise;  $\Delta^t$  indicates the difference between  $N^t$  and  $q^t$ , and it has  $\Delta^t = |N^t - q^t|$ ;  $\xi$  and  $\hat{\xi}$  are the weighting factors reflecting the importance of the location privacy over the usage pattern privacy in different situations;  $K \in [1, N^t]$  denotes the metric of usage pattern privacy, which is the number of dummy tasks. The dummy tasks may sacrifice some system performance but will increase the privacy level, and the proposed algorithm will balance them.

The first term of equation (8) represents that if the mobile device offloaded all its tasks to the MEC server ( $\Delta^t = 0$ ), in order to protect the usage pattern privacy, it will continue to offload  $K$  dummy tasks to the MEC server to confuse the attacker. As such, the attacker cannot pinpoint the number of tasks actually generated by the user. According to the second term of equation (8), there are two situations correspond to the  $\Delta^t \neq 0$ . In the first situation,  $q^t = 0$  denotes that the tasks either queued in the buffer or processed locally otherwise. In order to protect the location privacy, the mobile device needs to offload  $\xi$  tasks (which is queuing in the buffer,  $0 \leq \xi \leq b^t$ ) to the MEC server for preventing the attacker from inferring the wireless channel status. In the second situation of  $\Delta^t \neq 0$ , some tasks are offloaded to the MEC server ( $q^t \neq 0$ ) and the privacy level can be achieved by  $\hat{\xi}/\Delta^t$ . It denotes the importance of the location privacy over the usage pattern privacy  $\hat{\xi}$ , which will increase as  $\Delta^t$  decreases.

3.2.4. *Objective Function.* In order to achieve a desirable trade-off between the system cost (i.e., computing delay and energy consumption) and the user's privacy level, we design different weights  $\omega_{\text{delay}}^t$ ,  $\omega_{\text{energy}}^t$ , and  $\omega_{\text{privacy}}^t$  to indicate the different preference device. These weights also can convert the privacy level and system cost into the same dimension. Thus, the objective of this paper is to achieve robust minimization of a weighted sum of the privacy level and system cost for the mobile device. Based on [29], given a finite time horizon  $T$ , the problem can be formulated as

$$\begin{aligned} \min \quad & \sum_{t=0}^T \omega_{\text{delay}}^t D^t + \omega_{\text{energy}}^t E^t + \omega_{\text{privacy}}^t \frac{1}{P^t}, \\ \text{s.t.} \quad & (1) - (3), \end{aligned} \quad (9)$$

From the mobile device perspective, it is difficult for them to explore the system-wide information (e.g., the wireless channel states and resource availability) in advance and it may need extremely expensive energy cost. Therefore, devising a device-level adaptive privacy-preserving task offloading policy is highly desirable, in which the future system-level information will not be needed.

#### 4. Algorithm Design

In this section, we focus on the privacy preserving task offloading problem in the MEC-enabled network and propose a device-level privacy-aware online learning scheme to minimize the objective in equation (9) for the mobile device without knowing the system-side information.

Firstly, we transform the information-constrained multi-objective optimization problem to a contextual multi-armed bandit (MAB) problem [30] with a semiparametric reward model. Then, we propose a privacy-aware online task offloading (PAOTO) algorithm which can accommodate the network dynamics at the device level and learn the optimal offloading policy for the mobile device while maintaining the user's privacy.

**4.1. Problem Transformation.** In this work, we focus on the device-level and privacy-aware task offloading problem, which is a typical sequential decision problem. For decoupling the time dependency, we formulate this problem as a contextual MAB problem with a relaxed, semiparametric reward model in [30]. It is an extended version of the conventional contextual MAB that has a linear reward model [31]. Both versions can utilize the contextual feature vector to indicate the use-side information for overcoming the challenges of lacking future system information. However, why we use the contextual MAB with a relaxed, semiparametric reward model is that the privacy metric in our model is difficult to formulate as a linear reward model. The semiparametric reward model can provide a more relaxed reward model, and this proof can be found in the literature [30].

Accordingly, in order to learn the network dynamics and take the privacy protection into consideration, the problem in this paper can be transformed as a semiparametric contextual MAB problem [13], which can address the tradeoffs inherent in the sequential decision problem and has a relaxed, semiparametric reward model. This model can be described as

$$\mathbb{E} \left( \sum_{n \in \mathcal{N}^t} r_{n,i}(t) \mid \mathcal{F}^{t-1} \right) = \nu(t) + \sum_{n \in \mathcal{N}^t} b(t)^\top \bar{\mu}_{n,i}(t), \quad (10)$$

where  $r_{n,i}(t)$  is the received cost of offloading task  $n$  to execution platform  $i$ ;  $\nu(t)$  is a nonparametric component;  $b(t)$  is a current contextual feature vector;  $\bar{\mu}_{n,i}(t)$  is a fixed but unknown underlying expectation of the feature vector  $\mu_{n,i}(t)$ ; the  $\mathcal{F}^{t-1}$  is the union of historical information and  $b(t)$ . Furthermore, it has assumptions about the upper bound of some parameters, which is  $\|b(t)\|_2 \leq 1$ ,  $\|\bar{\mu}_{n,i}(t)\|_2 \leq 1$ ,  $\|\nu(t)\|_2 \leq 1$ , and  $\|\cdot\|_2$  denotes the  $L_2$  norm.

When the computation tasks arrive, the task offloading decision can be executed for each task by the mobile device. Nonetheless, only the device-side status information can be observable, which can be described as a contextual feature vector  $b(t) = [\omega_{\text{delay}}^t \mathbb{M}^t, \omega_{\text{energy}}^t \mathbb{S}^t] \in \mathbb{R}^{(2(N_{\text{max}}^t + b_{\text{max}}))}$  for arriving tasks. More specifically,  $\mathbb{M}^t \in \mathbb{R}^{(N_{\text{max}}^t + b_{\text{max}})}$  denotes the computation demand vector of tasks. The first  $N^t + b^t$  values of  $\mathbb{M}^t$  are corresponding computation demand  $m_n^t$  of each task  $n$ , and the remaining values are 0;  $\mathbb{S}^t \in \mathbb{R}^{(N_{\text{max}}^t + b_{\text{max}})}$  is a transition vector, which denotes the number of tasks in time slot  $t$ . The first  $N^t + b^t$  values of  $\mathbb{S}^t$  are 1, and the remainder are 0. According to the system cost (including processing latency and energy consumption cost) defined in Section 3, we transform them as a feature vector  $\mu_{n,i}(t) = [(1/\gamma_i^t), e_{n,i}^t] \in \mathbb{R}^{(2(N_{\text{max}}^t + b_{\text{max}}))}$  to better learn the network uncertainty and resource availability, which is related to  $\bar{\mu}_{n,i}(t)$ . Besides, the privacy level in this task offloading policy will be formulated as the aggregated nonparametric component  $\nu(t) = \omega_{\text{privacy}}^t (1/P^t)$  based on the reward model of contextual MAB in equation (10). The reason of this is that it cannot be directly formulated as a linear component like other metrics (such as computing delay and energy consumption). We assume that it can be calculated when all task decisions are completed at the end of  $t$ .

Hereinafter, we define  $\mathcal{H}^{t-1} = \{\mathcal{A}(\tau), r_{\mathcal{A}(\tau)}(\tau), b(\tau)\}$ ,  $\tau = \{1, 2, \dots, t-1\}$  as the historical observations until  $t-1$ , where  $\mathcal{A}(\tau)$  represents the set of actions for all tasks at time slot  $\tau$  and  $r_{\mathcal{A}(\tau)}(\tau)$  denotes the total received cost at time slot  $\tau$ . And the  $\mathcal{F}^{t-1} = \{\mathcal{H}^{t-1}, b(t)\}$  can be denoted as the union of historical information  $\mathcal{H}^{t-1}$  and the current contextual feature vector  $b(t)$ . Given that  $\mathcal{F}^{t-1}$ , we assume that the expectation of the total received cost  $r_{\mathcal{A}(t)}(t)$  can be decomposed into a time-invariant linear component  $\sum_{n=0}^{N^t+b^t} b(t)^\top \bar{\mu}_{n,\mathcal{A}(t)}(t)$  (associated with processing delay and energy consumption cost) and a nonparametric component  $\nu(t)$  (associated with the privacy-preserving level). Therefore, according to equation (10), we have

$$\begin{aligned} \mathbb{E} \left( r_{\mathcal{A}(t)}(t) \mid \mathcal{F}^{t-1} \right) &= \mathbb{E} \left( \sum_{n=0}^{N^t+b^t} r_{n,i}(t) \mid \mathcal{F}^{t-1} \right) \\ &= \nu(t) + \sum_{n=0}^{N^t+b^t} b(t)^\top \bar{\mu}_{n,i}(t) \\ &= \omega_{\text{privacy}}^t \frac{1}{P^t} + \sum_{n=0}^{N^t+b^t} b(t)^\top \bar{\mu}_{n,i}(t). \end{aligned} \quad (11)$$

The task offloading scheme needs to select an execution platform  $i$  (or called an arm at MAB) for every task  $n$  at time slot  $t$ . Specifically, we let  $a_n(t) \in \mathcal{A}(t)$  denote the choice for every task and let the optimal action to be  $a_n^*(t) \in \mathcal{A}^*(t)$  based on equation (10). Additionally, it must be noted that the nonparametric component  $\nu(t)$  in equation (11) depends on time and historical information, but not on the current action [30]. Hence, the optimal received cost  $r_{n,i}(t)$  of each

task can be obtained by minimum  $b(t)^\top \bar{\mu}_{n,i}(t)$  and we can achieve the optimal offloading decision of each task by  $a_n^*(t) = \arg\min_i b(t)^\top \bar{\mu}_{n,i}(t)$ . Indeed, the privacy level  $v(t)$  will have an impact on the aggregated received cost of all tasks at the end of time slot  $t$  and this aggregated received cost will be used to update the contextual feature vector for the next interval  $t$ .

Beyond that, the regret at time slot  $t$  is defined as the difference between the average cost of the optimal choices and the universal choices for all tasks and it does not depend on  $v(t)$  either. Hence, the regret can be expressed as

$$\text{Regret}(t) = b(t)^\top \bar{\mu}_{\mathcal{A}^*(t)} - b(t)^\top \bar{\mu}_{\mathcal{A}(t)}. \quad (12)$$

Moreover, given a finite time horizon  $T$ , the total regret can be described as

$$R(t) = \sum_{t=1}^T \text{regret}(t) = \sum_{t=1}^T b(t)^\top \bar{\mu}_{\mathcal{A}^*(t)} - b(t)^\top \bar{\mu}_{\mathcal{A}(t)}. \quad (13)$$

This regret is used to evaluate the effectiveness of task offloading decision making based on the online learning of system-level information.

**4.2. Privacy-Aware Online Task Offloading Algorithm.** In order to minimize the total system cost (e.g., delay and energy consumption cost) and protect user's privacy without exploring any system-level information, a novel PAOTO algorithm is proposed in this work. In particular, this algorithm keeps the framework of the Thompson sampling (TS) with a semiparameter reward model [32]. Its key idea is to estimate and learn the device's performance by selecting different actions over time based on the contextual information. At the same time, the privacy metric can be abstracted into the semiparametric reward model.

In the proposed PAOTO algorithm, the mobile device will learn the network information while executing the task offloading policy. As time goes by, the mobile device learns abundant information and it can estimate how to offload these  $(N^t + b^t)$  tasks for achieving the optimal system cost and privacy-preserving level. According to the aforementioned MAB transformation of our problem, it is known that the optimal offloading decision and the received cost  $r_{n,i}(t)$  of each task mainly depend on the current contextual feature vector  $b(t)$  and the fixed but unknown feature vector  $\bar{\mu}_{n,i}(t)$ . Through the previous trial and error, the underlying relationship between the feature vectors and received cost will be learned by the mobile device. The  $v(t)$  is related to the privacy metric  $P^t$ , and it can be achieved after all tasks are offloaded at the end of time slot  $t$ . Hence, we let  $\hat{\mu}_{n,i}(t)$  denote the estimate of the feature vector  $\mu_{n,i}(t)$  and  $B_i(t)$  represent the cumulative contextual vector. The estimate of feature vector  $\hat{\mu}_{n,i}(t)$  and the cumulative contextual vector  $B_i(t)$  can be denoted as

$$\begin{aligned} \hat{\mu}_{n,i}(t) &= (I_d + \Sigma \wedge_t + \Sigma_t)^{-1} \sum_{\tau=1}^{t-1} 2X_\tau r_{\mathcal{A}(\tau)}(\tau) = B_i(t)^{-1} \sum_{\tau=1}^{t-1} 2X_\tau r_{\mathcal{A}(\tau)}(\tau), \\ B_i(t) &= I_d + \hat{\Sigma}_t + \Sigma_t = I_d + \sum_{\tau=1}^{t-1} X_\tau X_\tau^\top + \sum_{\tau=1}^{t-1} \mathbb{E}(X_\tau X_\tau^\top | \mathcal{F}_{\tau-1}), \end{aligned} \quad (14)$$

where  $X_\tau = b(\tau) - \mathbb{E}(b(\tau) | \mathcal{F}_{\tau-1})$ ;  $I$  is a  $d$  dimensional identity matrix, where  $d = 2(N_{\max}^t + b_{\max})$ .

For ease of exposition, the  $\mathbb{E}(b(\tau) | \mathcal{F}_{\tau-1})$  in  $X_\tau$  can be denoted as  $\bar{b}(\tau)$  and it can be calculated as

$$\bar{b}(\tau) = \mathbb{E} \left( \sum_{n=1}^{N^t+b^t} \sum_{i=1}^3 I(a_n(\tau) = i) b(\tau) | \mathcal{F}_{\tau-1} \right) = \sum_{n=1}^{N^t+b^t} \sum_{i=1}^3 \pi_{n,i}(\tau) b(\tau), \quad (15)$$

where  $\pi_{n,i}(\tau) = \mathbb{P}(a_n(\tau) = i | \mathcal{F}_{\tau-1})$  is the probability of offloading task  $n$  to the  $i$ th execution platform at time  $\tau$ .

Besides, we can calculate the covariance  $\mathbb{E}(X_\tau X_\tau^\top | \mathcal{F}_{\tau-1})$  as follows:

$$\mathbb{E}(X_\tau X_\tau^\top | \mathcal{F}_{\tau-1}) = \sum_{n=1}^{N^t+b^t} \sum_{i=1}^3 \pi_{n,i}(\tau) (b(\tau) - \bar{b}(\tau))(b(\tau) - \bar{b}(\tau))^\top. \quad (16)$$

Accordingly, the mobile device can continuously explore and then gather the relationship between the feature vector of each task and the system cost of the chosen execution platform. Then, it also measures the corresponding privacy level to estimate which execution platform is likely to give the minimum system cost while maintaining a good privacy level.

In this paper, the TS-based online learning algorithm will be applied to learn the underlying relation between the feature vector and received cost. Hence, we should construct a distributional likelihood function to sample the estimated cost. Firstly, the standard deviation of the estimated cost  $\hat{r}_{n,i}(t) = b(t)^\top \hat{\mu}_{n,i}(t)$  can be defined as  $\hat{s}_{n,i}(t) = \sqrt{b(t)^\top B_i(t)^{-1} b(t)}$  and the standard deviation of the sampling cost  $\tilde{r}_{n,i}(t) = b(t)^\top \tilde{\mu}_{n,i}(t)$  can be denoted as  $\tilde{s}_{n,i}(t) = v \sqrt{b(t)^\top B_i(t)^{-1} b(t)} = v \hat{s}_{n,i}(t)$ , where  $v = (2R + 6) \sqrt{6d \log(T/\delta)}$  is a control parameter.

According to Bayes' theorem  $P(B_i | A) \propto P(B_i)P(A | B_i)$ , we have:

$$\Pr(b(t)^\top \tilde{\mu}_{n,i}(t) | \tilde{r}_{n,i}(t)) \propto \Pr(\tilde{r}_{n,i}(t) | b(t)^\top \tilde{\mu}_{n,i}(t)) \Pr(b(t)^\top \tilde{\mu}_{n,i}(t)). \quad (17)$$

Based on the TS algorithm in [31], if the prior for received cost  $r_{n,i}(t) = b(t)^\top \bar{\mu}_{n,i}(t)$  at time slot  $t$  is given by  $\mathcal{N}(b(t)^\top \hat{\mu}_{n,i}(t), v^2 b(t)^\top B_i(t)^{-1} b(t))$ , it is easy to compute the posterior distribution at time slot  $t + 1$ , i.e.,  $\mathcal{N}(b(t+1)^\top \hat{\mu}_{n,i}(t+1), v^2 b(t+1)^\top B_i(t+1)^{-1} b(t+1))$  (details of this

**Input:**  $N^t, N_{\max}^t, b^t, b_{\max}, m_n^t, i \in \mathcal{F}$ ;

**Output:** total received cost  $r_{\mathcal{A}(t)}(t)$ .

1: **Initialization:** Initialize the cumulative contextual vector  $B_i = I_d$ , where  $I_d$  is a  $2(N_{\max}^t + b_{\max})$  dimensional identity matrix, the cumulative contextual system cost  $y_i = 0_d$ , and a control parameter  $\nu = (2R + 6) \sqrt{6d \log(T/\delta)}$ ,  $\delta \in (0, 1)$ .

2: **End initialization**

3: **for**  $t = 1, 2, \dots, T$  **do**

4:   **for**  $n = 1, 2, \dots, N^t + b^t$  **do**

5:     **for**  $i = 1, 2, 3$  **do**

6:       **Compute** the estimated feature vector  $\hat{\mu}_{n,i}(t) = B_i^{-1}y_i$ .

7:       **Sample** the cost  $\tilde{r}_{n,i}(t)$  independently for each task  $n$  and each execution platform  $i$  from the Gaussian distribution  $\mathcal{N}(b(t)^\top \hat{\mu}_{n,i}(t), \nu^2 b(t)^\top B_i(t)^{-1} b(t))$ .

8:       **Compute** the probability of offloading task  $n$  to the  $i$ -th execution platform  $\pi_{n,i}(t) = \mathbb{P}(a_n(t) = i | \mathcal{F}_{t-1})$ .

9:     **End for**

10:    **Select** the offloading execution platform  $a_n(t) = \arg \min_i \tilde{r}_{n,i}(t)$ .

11:    **Record** the selection  $a_n(t)$  into offloading decisions vector  $\mathcal{A}(t)$ .

12:    **End for**

13:    **Compute** the aggregated received cost  $r_{\mathcal{A}(t)}(t)$  via  $\mathcal{A}(t)$ , Equation (5), (6), (8) and (9).

14:    **Update**  $B$  and  $y$ :

15:     $B_{\mathcal{A}(t)} \leftarrow B_{\mathcal{A}(t)} + (b(t) - \bar{b}(t))(b(t) - \bar{b}(t))^\top + \sum_{n=1}^{N^t+b^t} \sum_{i=1}^{i=3} \pi_{n,i}(t) (b(t) - \bar{b}(t))(b(t) - \bar{b}(t))^\top$

16:     $y_{\mathcal{A}(t)} \leftarrow y_{\mathcal{A}(t)} + 2(b(t) - \bar{b}(t))r_{\mathcal{A}(t)}(t)$ .

17: **End for**

ALGORITHM 1: Privacy-aware online task offloading (PAOTO) algorithm.

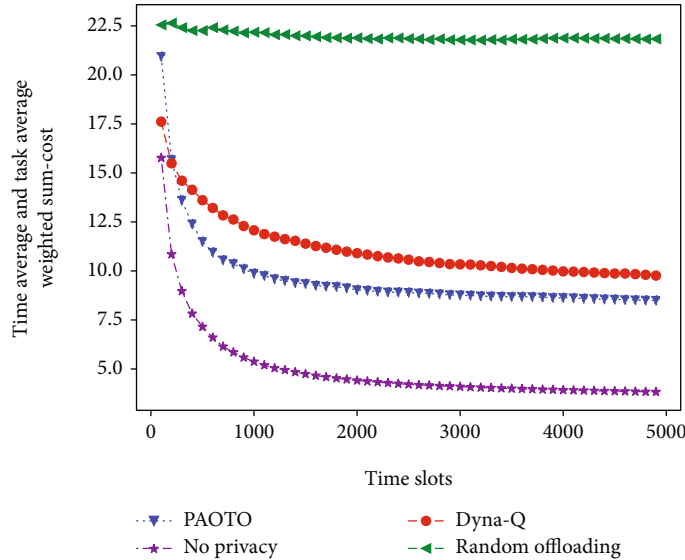


FIGURE 2: The simulation result comparison of time average and task average weighted sum-cost for the PAOTO algorithm, random offloading algorithm, Dyna-Q algorithm, and no privacy scenario.

computation can be seen in Appendix A.1 of [31]). Hence, at every time step  $t$ , we can use this Gaussian likelihood function  $\mathcal{N}(b(t)^\top \hat{\mu}_{n,i}(t), \nu^2 b(t)^\top B_i(t)^{-1} b(t))$  to sample the cost  $\tilde{r}_{n,i}(t)$  for offloaded task  $n$  at execution platform  $i$  in our algorithm. Then, the sampling cost  $\tilde{r}_{n,i}(t)$  will be used to estimate the performance of offloaded task  $n$  at execution platform  $i$  and finally the execution platform that has minimum  $\tilde{r}_{n,i}(t)$ .

Hence, guided by the problem transformation and key vectors mentioned above, we introduce the PAOTO algorithm in Algorithm 1.

Algorithm 1 gives the details of exploring the optimal solution that can make an adaptive task offloading decision and preserve the privacy of users. It estimates the offloading cost  $r_{n,i}(t)$  of each task based on context information  $b(t)$  and performance feature vector  $\mu_{n,i}(t)$  and selects the best offloading action based on the minimum received cost  $r_{n,i}(t)$ . At the same time, it calculates the offloading probability  $\pi_{n,i}(t)$  to fit the MAB problem with a semiparametric reward model for privacy preservation. At the end, it utilizes total received cost  $r_{\mathcal{A}(t)}(t)$  to update cumulative

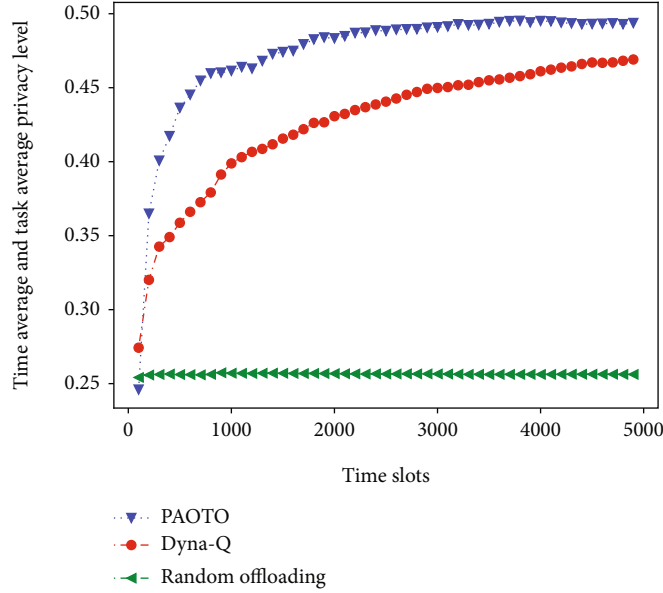


FIGURE 3: The simulation result comparison of the time average and task average privacy level for the PAOTO algorithm, random offloading algorithm, and Dyna-Q algorithm.

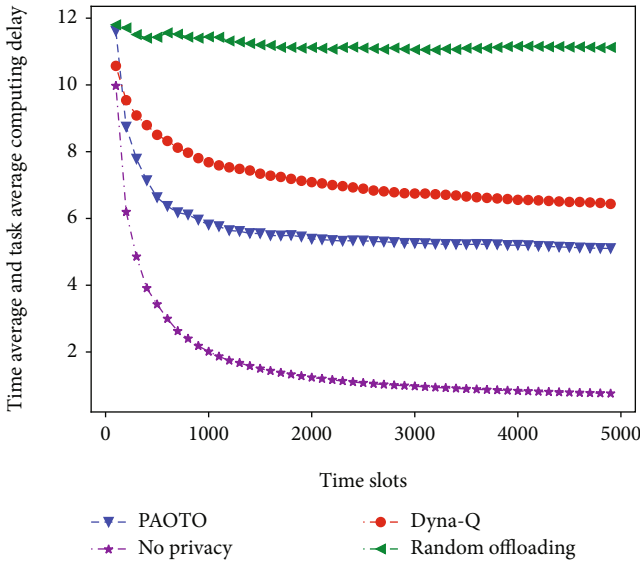


FIGURE 4: The simulation result comparison of the time average and task average computing delay for the PAOTO algorithm, random offloading algorithm, Dyna-Q algorithm, and no privacy scenario.

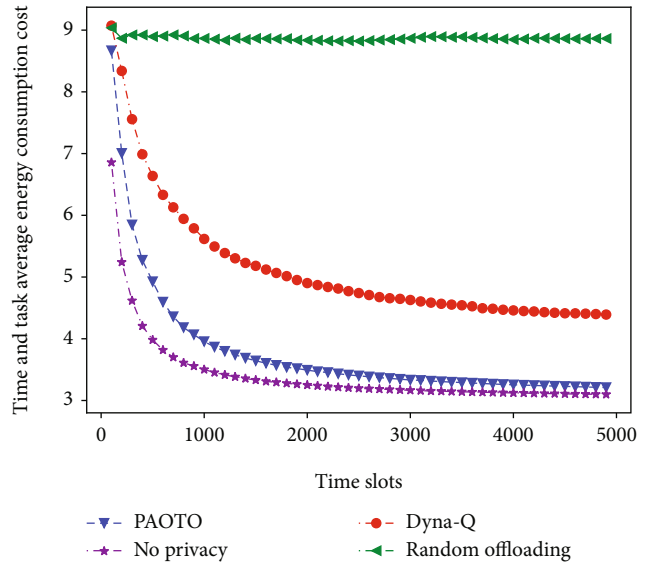


FIGURE 5: The simulation result comparison of the time average and task average energy consumption cost for the PAOTO algorithm, random offloading algorithm, Dyna-Q algorithm, and no privacy scenario.

contextual vector  $B$  and cumulative contextual system cost  $y$  corresponding to the decisions vector  $\mathcal{A}(t)$  of all tasks at every time slot  $t$ .

### 5. Simulation Results

In this section, extensive simulations are conducted to evaluate the performance of the proposed PAOTO algorithm under different scenarios. We build our simulations in Python 3.6. The implementations are conducted on a Lenovo desktop PC equipped with Intel(R) core (TM) i7-4500U CPU

@1.80 GHz processor and 12.0 GB (11.7 GB available) RAM. The simulation settings, algorithm benchmarks, and performance evaluation are elaborated below.

*5.1. Simulation Settings.* In our simulation environment, we consider a MEC system, in which the access point is deployed with the MEC server. The  $N^t$  computing tasks are randomly generated by the mobile device at every time slot  $t$ , where the maximum of  $N^t$  is in the range from 10 to 60. The maximum buffer capacity  $b_{\max}$  of the mobile device can be set to 10. The length of each time slot is 1 s. The total computation capacity

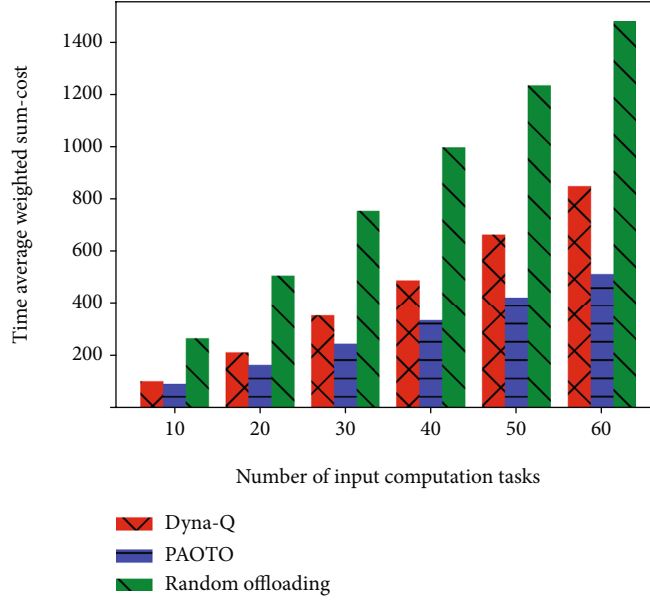


FIGURE 6: The simulation result comparison of time average weighted sum-cost for the PAOTO algorithm, random offloading algorithm, and Dyna-Q algorithm under different numbers of input computation tasks.

for MEC server  $\gamma_s$  is uniformly distributed in (10, 15) GHz. In order to accommodate the dynamics, we assume that the computation capacity of the mobile device  $\gamma_l$  is determined randomly from 1 to 3 GHz and the converted computation capacity of buffer  $\gamma_b$  is uniformly distributed in (0.1, 0.15) GHz. Based on [33], the data size of each task  $\lambda_n$  is distributed in (300 K, 800 K) bits and the computation intensity  $\delta_n$  is taken randomly within (250, 1000) CPU cycles/bit. Thus, we can get the required CPU cycles of a computing task  $m_n$  by  $m_n = \lambda_n * \delta_n$  (CPU cycles). Besides, the energy consumption for transmitting one task to the MEC server is uniformly distributed in (0.1, 0.5) J and the mobile device consumes 0.8 J to 3 J to locally compute one task and 0.5 J to 1 J to buffer one task. The weights of processing delay, energy consumption, and privacy metric, which are  $\omega_{\text{delay}}^t$ ,  $\omega_{\text{energy}}^t$  and  $\omega_{\text{privacy}}^t$ , respectively, can be dynamically set by the users according to the user's preferences and the running application demands. In our simulation, we set them to 1, 1, and 10, respectively. The control parameter  $\nu$  of the PAOTO algorithm is usually set to 1.

**5.2. Benchmarks.** The simulations are carried out based on the above setting. In order to better manifest the advantages and effectiveness of the proposed algorithm, two typical benchmarks are implemented for comparison with the PAOTO algorithm, which are presented as follows:

- (1) *Random offloading algorithm:* the random offloading algorithm is chosen as one of the baselines, which will arrange the offloading in a random way. This is the method for the resource-constrained mobile device to decentralize computing tasks. However, it does not consider the privacy preservation and performance optimization. The purpose of this benchmark is to evaluate the necessity of the proposed algorithm

- (2) *Dyna-Q algorithm:* we implemented the Dyna-Q algorithm as one of the benchmarks in our simulations. The implementation details may be slightly different from that of [15], but the main framework is the same. The Dyna-Q in [15] is a reinforcement learning- (RL-) based privacy-aware offloading scheme. It is an improvement of the Q-learning method, combining the model-independent and model-dependent methods. But, it requires more system-level information (e.g., assumption of the Markov model) than the proposed algorithm. As the most state-of-the-art and relevant scheme to our works, the implementation of Dyna-Q can bring more reliable performance guarantees for our algorithm evaluation

- (3) *No privacy scenario:* the scenario that does not consider privacy protection is also used as one of our baselines. According to [10], when mobile devices do not consider privacy protection but focus solely on optimizing delay and energy consumption, the optimal latency and energy consumption performance can be obtained. Comparing with a scenario that does not consider privacy protection, it can reflect that the proposed algorithm will compromise the system performance in order to protect user's privacy

As such, our algorithms are comparing the performance with these two benchmarks for analysis and these values match those used in previous works.

**5.3. Numerical Results.** In this section, the numerical results are presented to evaluate the effectiveness of the proposed algorithm. The weighted sum-cost, privacy level, computing delay, and energy consumption cost of the PAOTO algorithm in a period of time are compared with the two



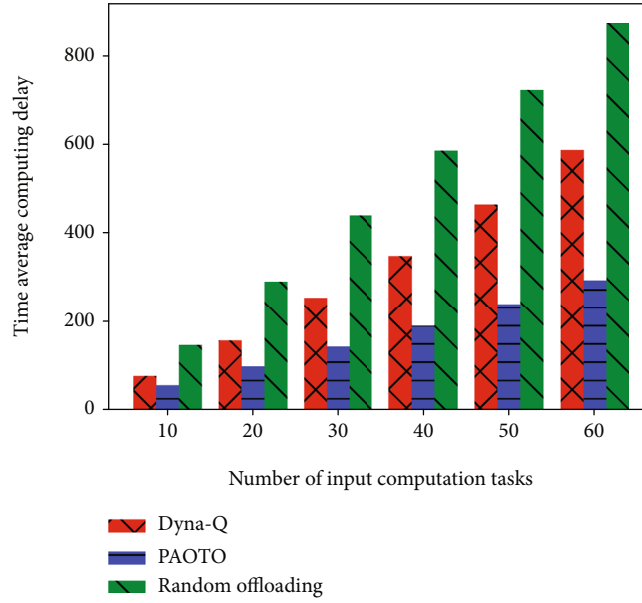


FIGURE 7: The simulation result comparison of time average computing delay for the PAOTO algorithm, random offloading algorithm, and Dyna-Q algorithm under different numbers of input computation tasks.

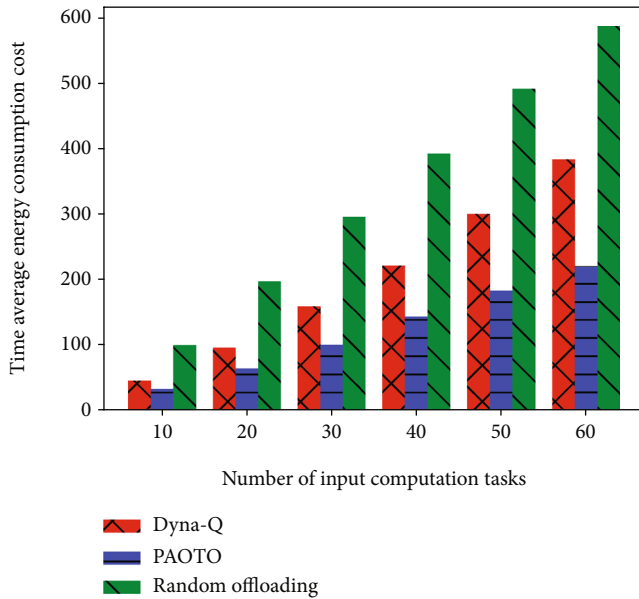


FIGURE 8: The simulation result comparison of time average energy consumption for the PAOTO algorithm, random offloading algorithm, and Dyna-Q algorithm under different numbers of input computation tasks.

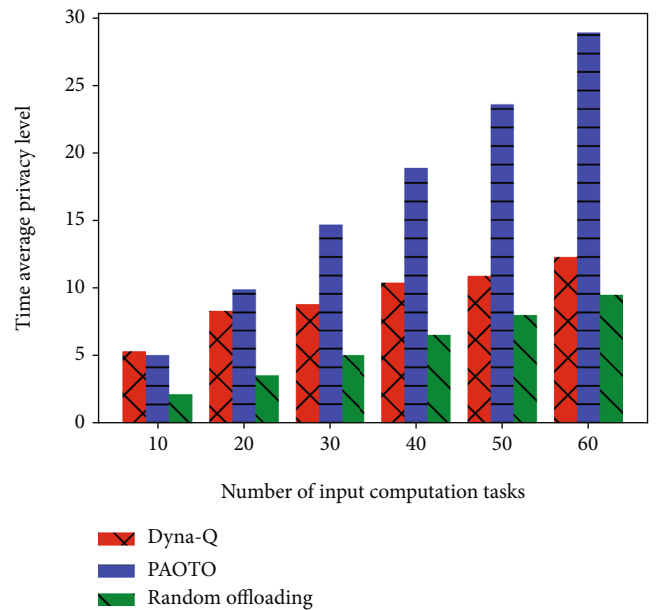


FIGURE 9: The simulation result comparison of the time average privacy level for the PAOTO algorithm, random offloading algorithm, and Dyna-Q algorithm under different numbers of input computation tasks.

benchmarks to evaluate the performance of the proposed algorithm.

**5.3.1. The First Set of Simulations.** In the first set of simulations, we randomly generate some tasks for the mobile device at each time slot  $t$ , which are the same for the three algorithms. The number of newly generated tasks are taken randomly within [5, 30]. The task offloading policy is executed for each task per round. Since the number of tasks in each

round is dynamic, the simulation results are averaged for each task. The results of average weighted sum-cost, privacy level, computing delay, and energy consumption cost are reported as the following and the results are plotted at every 100 time slots.

As shown in Figure 2, we trace the average weighted sum-cost of the PAOTO algorithm, random offloading algorithm, and Dyna-Q algorithm at each time slot  $t$ . It can be seen that the PAOTO algorithm can obtain lower system cost for each

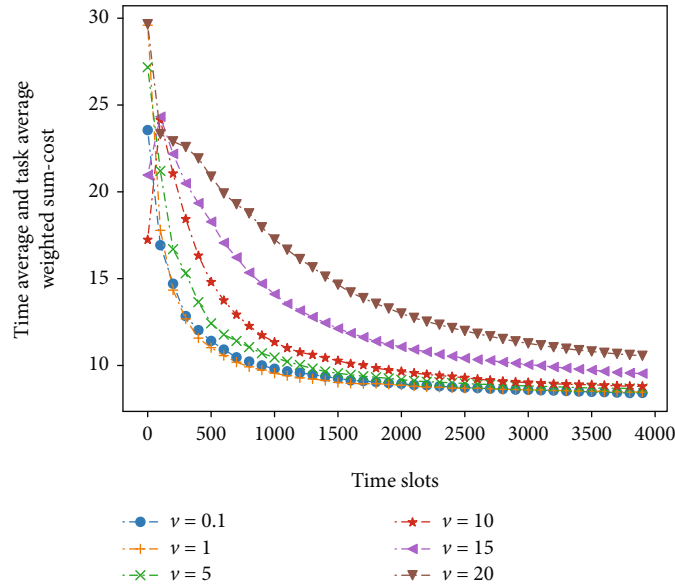


FIGURE 10: Performance comparison with different values of  $\nu$ .

task with about 23.0% reduction comparing to the Dyna-Q algorithm and about 50.1% to the random offloading at the 1000th time slot. However, compared to another scenario that the privacy is not considered, the proposed PAOTO algorithm has a higher system cost. This shows that the PAOTA algorithm has compromised the cost in order to protect privacy, which obtains the suboptimal solution.

According to Figure 3, we can see that the proposed algorithm achieves better performance of the privacy-preserving level comparing to the random offloading and Dyna-Q algorithm. The proposed algorithm improves 4.8% and 19.1% of the privacy level compared with the Dyna-Q scheme and random offloading algorithm, respectively, at the 2000th time slot. The performance comparison of computing delay and energy consumption cost also verifies the improvement of the proposed algorithm, which are shown in Figures 4 and 5, respectively. For instance, compared to Dyna-Q, the computing delay and the energy consumption cost of the PAOTO algorithm decrease by 22.5% and 25.4%, respectively, at the 1000th time slot. It is a pity that compared to the scenarios that privacy is not considered (i.e., the optimal solution), the latency and energy consumption performance of the proposed algorithm is slightly worse. Because it sacrifices some performance in order to preserve privacy. Besides, the simulation results of the random offloading scheme are very poor, which further proves that it is of great significance to study the task offloading and preserve the privacy of the users in the MEC system. Given these facts in the first set of simulations, it can be observed that the PAOTO algorithm outperforms the other two benchmarks and it obtains suboptimal task offloading performance while protecting user's privacy.

**5.3.2. The Second Set of Simulations.** In the second set of simulations, we investigate the performance of the proposed algorithm with different maximum number of input computation tasks  $N_{\max}^t$ , which ranged from 10 to 60. The data size of each task is uniformly distributed in (300, 800 K) bits.

These simulation results in the second set are averaged over the first 2000 time slots. As shown in Figures 6–8, the PAOTO algorithm can get a lower average weighted sum-cost, computing delay, and the energy consumption cost than the other two benchmarks. And the improvements of these performances (histogram difference) increase as the  $N_{\max}^t$  increases from 10 to 60. For instance, when the number of computing tasks is 20, compared with Dyna-Q, the average sum-cost, computing delay, and the energy consumption cost of the PAOTO algorithm increase by 28.1%, 44.7%, and 28.6%, respectively. Whereas, when the number of computing tasks is 60, they are 29.3%, 51.6%, and 37.2%, respectively. The reason is that as the total number of tasks increases, the Dyna-Q algorithm requires more time to learn, and the random offloading does not have any performance optimization effects, but the proposed algorithm has stable processing efficiency to obtain a lower cost.

Additionally, as shown in Figure 9, with the increment of the number of tasks, the privacy level of the proposed algorithm will increase significantly but the privacy level of the Dyna-Q algorithm will decrease slightly. It is also because the processing efficiency of the Dyna-Q algorithm will decrease as the number of tasks increases. Besides, the privacy level of the random offloading algorithm is not affected by the number of tasks. Hence, the simulation results of the second set validate that the PAOTO algorithm has superior and stable system performance and privacy-preserving level for increasing computing-intensive tasks.

From the two set of simulations mentioned above, it can be seen that the PAOTO algorithm meets the objective of this paper that receives the close-to-optimal delay and energy consumption performance for MD while protecting the user's privacy. And it has a significant performance improvement comparing to the other two benchmarks.

**5.3.3. The Third Set of Simulations.** In order to analyze the effect of different key parameters (i.e.,  $\delta$  and  $K$ ) on the

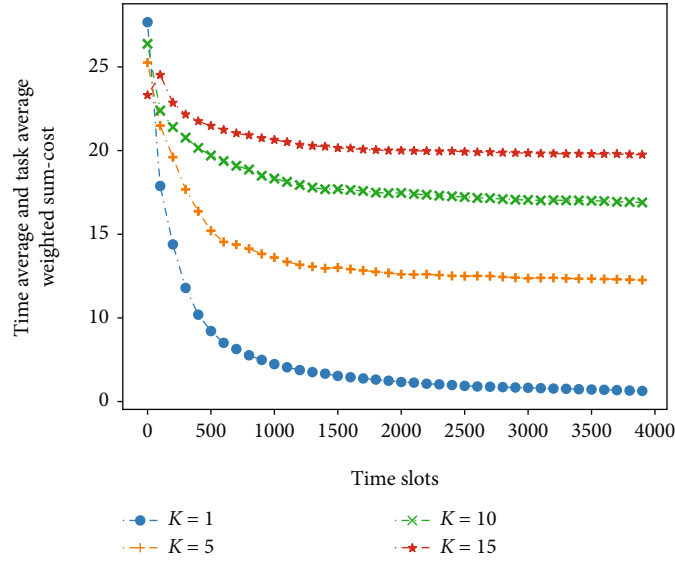


FIGURE 11: Performance comparison with different values of  $K$ .

PAOTO algorithm, the weighted sum-cost is plotted under different values of  $\nu$  and the number of dummy tasks  $K$ . First, the parameter  $\nu$  in the PAOTO algorithm is associated with the standard deviation of the sampling, where  $\nu = (2R + 6)\sqrt{6d \log(T/\delta)}$ ,  $\delta \in (0, 1)$ . Thus, we set the values of  $\nu$  as 0.1, 1, 5, 10, 15, and 20. As shown in Figure 10, we can observe that the values of  $\nu$  and the average weighted sum-cost of the PAOTO algorithm are positively correlated when  $\nu \geq 1$ , such as the curves  $\nu = 1$ ,  $\nu = 5$ , and  $\nu = 10$ . As the value of  $\nu$  is larger, the convergence of the PAOTO algorithm becomes worse. However, when  $\nu < 1$ , the average cost of the PAOTO algorithm will increase with the decrease of  $\nu$ , such as curves  $\nu = 1$  and  $\nu = 0.1$ . The reason is the cost trade-off in the theoretical bound, and there are different effects before and after reaching the bound.

Second, the different number of dummy tasks  $K$  that is related to privacy metric  $P^t$  is simulated for weighted sum-cost of the proposed algorithm. As shown in Figure 11,  $K$  can be set as 1, 5, 10, and 15. Then, we can observe that as  $K$  increases, the weighted sum-cost will also increase. However, as  $K$  is larger, the increment of the weighted sum-cost will decrease. According to equations (8) and (9) in Section 3,  $K$  is directly proportional to the privacy metric  $P^t$  and the privacy metric  $P^t$  is inversely proportional to the weighted sum-cost. When the number of dummy tasks  $K$  increases, the system cost will increase at the beginning. Nevertheless, taking dummy tasks into the privacy metric can restrict the increment of the weighted sum-cost. That is, there is a tradeoff between the system cost and the privacy metric.

## 6. Conclusions

In this paper, we investigated joint task offloading and privacy preservation for the small-size and low-power mobile devices without any system-level network information in the MEC system. The objective is to minimize a weighted sum of the computing delay, energy consumption cost, and

reciprocal of the privacy metric. In particular, the joint optimization problem has been formulated as a contextual MAB problem with a semiparametric reward model to accommodate network dynamics, in which the privacy metric is taken into account. Subsequently, a privacy-aware online task offloading (PAOTO) algorithm is proposed to explore the balance between the optimal system cost and the privacy level. The simulation results show that the proposed algorithm can provide near-optimal solutions in a short computing time. In the future, we will extend our work to the scenarios that have multiple MEC servers with distinct computing capability and take the bandwidth constraint into account.

## Data Availability

The (DATA TYPE) data used to support the findings of this study are included within the article.

## Disclosure

Additionally, a preliminary version of this work was accepted by WASA 2020. However, we have extended the conference paper significantly and the difference between the journal version and the conference version is above 50%.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant no. XDC02040300.

## References

- [1] J. Liu and Q. Zhang, "Code-partitioning offloading schemes in mobile edge computing for augmented reality," *IEEE Access*, vol. 7, pp. 11222–11236, 2019.
- [2] R. Gu, L. Yu, and J. Zhang, "Mefill: a multi-edged framework for intelligent and low latency mobile IOT services," in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, Seoul, Korea, 2020.
- [3] X. He, J. Liu, R. Jin, and H. Dai, "Privacy-aware offloading in mobile edge computing," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pp. 1–6, Singapore, 2017.
- [4] T. Yang, F. Wolff, and C. Papachristou, "Connected car networking," in *NAECON 2018 - IEEE National Aerospace and Electronics Conference*, pp. 60–64, Dayton, OH, USA, 2018.
- [5] CISCO, *Cisco Annual Internet Report (2018–2023) White Paper*, 2020, <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>.
- [6] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: a survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, 2018.
- [7] 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.
- [8] Q. Tang, R. Xie, T. Huang, and Y. Liu, "Jointly caching and computation resource allocation for mobile edge networks," *IET Networks*, vol. 8, no. 5, pp. 329–338, 2019.
- [9] J. Zhang, B. Chen, Y. Zhao, X. Cheng, and F. Hu, "Data security and privacy-preserving in edge computing paradigm: survey and open issues," *IEEE Access*, vol. 6, pp. 18209–18237, 2018.
- [10] X. He, R. Liu, and H. Dai, "Privacy-aware offloading in mobile-edge computing," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pp. 1–6, Singapore, 2017.
- [11] L. Huang, S. Bi, and Y. J. Zhang, "Deep reinforcement learning for online computation offloading in wireless powered mobile-edge computing networks," *IEEE Transactions on Mobile Computing*, vol. 19, no. 11, pp. 2581–2593, 2020.
- [12] N. Eshraghi and B. Liang, "Joint offloading decision and resource allocation with uncertain task computing requirement," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pp. 1414–1422, Paris, France, 2019.
- [13] H. Kim, H. Kim, and J. Chang, "A privacy-preserving kNN classification algorithm using Yao's garbled circuit on cloud computing," in *2017 IEEE 10th International Conference on Cloud Computing (CLOUD)*, pp. 766–769, Honolulu, CA, USA, 2017.
- [14] W. Juang and Y. Shue, "A secure and privacy protection digital goods trading scheme in cloud computing," in *2010 International Computer Symposium (ICS2010)*, pp. 288–293, Tainan, Taiwan, 2010.
- [15] M. Min and X. Wan, "Learning-based privacy-aware offloading for healthcare IoT with energy harvesting," *Internet of Things Journal*, vol. 6, no. 3, pp. 4307–4316, 2018.
- [16] C. Wang, P. Liu, T. Zhang, and J. Sun, "The adaptive vortex search algorithm of optimal path planning for forest fire rescue UAV," in *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pp. 400–403, Chongqing, China, 2018.
- [17] I. Farris, L. Militano, M. Nitti, L. Atzori, and A. Iera, "Federated edge-assisted mobile clouds for service provisioning in heterogeneous IoT environments," in *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pp. 591–596, Milan, Italy, 2015.
- [18] T. Ouyang, X. Chen, L. Zeng, and Z. Zhou, "Cost-aware edge resource probing for infrastructure-free edge computing: from optimal stopping to layered learning," in *2019 IEEE Real-Time Systems Symposium (RTSS)*, pp. 380–391, Hong Kong, China, 2019.
- [19] J. Xu, L. Chen, and P. Zhou, "Joint service caching and task offloading for mobile edge computing in dense networks," *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, 2018, pp. 207–215, Honolulu, HI, USA, 2018.
- [20] F. Wei, S. Chen, and W. Zou, "A greedy algorithm for task offloading in mobile edge computing system," *China Communications*, vol. 15, no. 11, pp. 149–157, 2018.
- [21] B. Dab, N. Aitsaadi, and R. Langar, "Q-Learning algorithm for joint computation offloading and resource allocation in edge cloud," in *2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pp. 45–52, Arlington, VA, USA, 2019.
- [22] G. Li and J. Cai, "An online incentive mechanism for collaborative task offloading in mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 624–636, 2020.
- [23] X. He, R. Jin, and H. Dai, "Deep PDS-learning for privacy-aware offloading in MEC-enabled IoT," *Internet of Things Journal*, vol. 6, no. 3, pp. 4547–4555, 2019.
- [24] H. Zhang and K. Zeng, "Pairwise Markov chain: a task scheduling strategy for privacy-preserving SIFT on edge," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 1432–1440, Paris, France, 2019.
- [25] P. Zhou, W. Chen, S. Ji, and H. Jiang, "Privacy-preserving online task allocation in edge-computing-enabled massive crowdsensing," *Internet of Things Journal*, vol. 6, no. 5, pp. 7773–7787, 2019.
- [26] X. Yu, M. Guan, M. Liao, and X. Fan, "Pre-migration of vehicle to network services based on priority in mobile edge computing," *IEEE Access*, vol. 7, pp. 3722–3730, 2019.
- [27] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation efficiency in a wireless-powered mobile edge computing network with NOMA," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pp. 1–7, Shanghai, China, 2019.
- [28] W. Zhang, Z. Zhang, S. Zeadally, and H. Chao, "Efficient task scheduling with stochastic delay cost in mobile edge computing," *IEEE Communications Letters*, vol. 23, no. 1, pp. 4–7, 2019.
- [29] T. Ouyang, R. Li, X. Chen, Z. Zhou, and X. Tang, "Adaptive user-managed service placement for mobile edge computing: an online learning approach," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 1468–1476, Paris, France, 2019.
- [30] K. Gi-Soo and P. Myunghee Cho, *Contextual Multi-Armed Bandit Algorithm for Semiparametric Reward Model*, International Conference on Machine Learning, 2019.
- [31] S. Agrawal and N. Goyal, "Thompson sampling for contextual bandits with linear payoffs," in *International Conference on Machine Learning*, pp. 127–135, Atlanta, USA, 2013.

- [32] D. J. Russo, B. Van Roy, A. Kazerouni, I. Osband, Z. Wen et al., “A tutorial on Thompson sampling,” *Foundations and Trends in Machine Learning*, vol. 11, no. 1, pp. 1–96, 2018.
- [33] J. Kwak, Y. Kim, J. Lee, and S. Chong, “DREAM: dynamic resource and task allocation for energy minimization in mobile cloud systems,” *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 12, pp. 2510–2523, 2015.

## Research Article

# Hybrid Precoding Algorithm for Millimeter-Wave Massive MIMO Systems with Subconnection Structures

Xue Zhang<sup>1</sup> and Feng Zhao<sup>2</sup> 

<sup>1</sup>Key Laboratory of Cognitive Radio and Information Processing, Guilin University of Electronic Technology, Guilin 541004, China

<sup>2</sup>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 16 February 2021; Revised 18 March 2021; Accepted 25 May 2021; Published 10 June 2021

Academic Editor: Nathalie Mitton

Copyright © 2021 Xue Zhang and Feng Zhao. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In mmWave massive MIMO systems, traditional digital precoding is difficult to be implemented because of the high cost and energy consumption of RF chains. Fortunately, the hybrid precoding which combines digital precoding and analog precoding not only solves this problem successfully, but also improves the performance of the system effectively. However, due to the constant mode constraint introduced by the phase shifter in the analog domain, it is difficult to solve the hybrid precoding directly. There is a solution which divides the total optimization problem into two stages to solve, that is, first fix the digital precoding matrix, solve the analog precoding matrix, and then optimize the digital precoding matrix according to the obtained analog precoding matrix. In this paper, a high energy-efficient hybrid precoding scheme is proposed for the subconnection structure. In the first stage, the optimization problem can be decomposed into a series of subproblems by means of the independent submatrix structure of the analog precoding matrix. When the optimized analog precoding matrix is obtained, the digital precoding matrix can be solved by the minimum mean error (MMSE). Finally, the digital precoding matrix is normalized to satisfy the constraint conditions. The simulation results demonstrate that the performance of the proposed algorithm is close to that of fully digital precoding based on subconnection structure and better than that of the existing algorithms. In addition, this paper presents the simulation analysis of the algorithm performance under imperfect channel state information. Simulation results show that when the estimation accuracy of channel state information is 0.8, the spectral efficiency of the proposed algorithm can already be maintained at a good level.

## 1. Introduction

With the rapid development of technology, the fifth generation mobile communication (5G) has attracted wide attention due to higher frequency, greater network capacity, and lower latency. It integrates many technologies, among which millimeter-wave (mmWave) communication and massive multiple-input multiple-output (MIMO) technology play important roles. Due to the rich spectrum resources available in the high frequency band, millimeter-wave communication can realize extremely high-speed and short distance communication with high gain, but there are also short transmission distance, poor penetration, and diffraction capability, vulnerability to climatic and environmental impacts [1], while mas-

sive MIMO systems, which use large arrays of antennas to communicate, can rapidly improve wireless data rates and system energy efficiency [2]. The shorter wavelength of mmWave signals enables large antenna arrays to be integrated into a smaller space, which implies that massive MIMO is feasible in wireless communications [3]. In addition, the combination of them can significantly improve user throughput, spectrum, and energy efficiency and increase the potential for mobile network capacity [4].

In millimeter-wave communication system, the cost of traditional digital precoding is too high to realize. Therefore, hybrid precoding, which combines digital precoding with analog precoding, is a better choice. It is usually based on two traditional structures: fully connected and subconnected.

Since the fully connected structure can approach the theoretical optimal spectrum efficiency, it has been extensively studied in the academic community. The hybrid precoding problem was reconstructed by the spatial sparsity of the mmWave channel in [5], and the orthogonal matching pursuit (OMP) algorithm is proposed to solve the problem. Based on the OMP algorithm, a greedy algorithm is proposed in [6]. The algorithm does not need to consider the geometry of the antenna array, which effectively reduces the computational complexity while achieving good performance. A threshold orthogonal matching pursuit (TOMP) algorithm with performance close to the optimal and higher than the OMP algorithm is proposed by setting an appropriate threshold [7]. Although the performance of the OMP algorithm is good, the complexity of the algorithm is relatively high. Therefore, a real-time and high performance precoding strategy based on singular value decomposition is studied in [8]. The performance of this scheme is similar to that of the OMP algorithm, and the complexity is much lower than that of the OMP algorithm. In [9], an alternative minimization (AltMin) algorithm is proposed by using the idea of manifold optimization, and the complexity of the algorithm is further reduced.

The characteristics of subconnected structure make it achieve a better balance between performance and cost, which has attracted the attention of many scholars. Unlike the one-to-one correspondence between the RF chain and the antenna in the fully connected structure, each RF chain in the subconnected structure is only connected with part of the antenna. Therefore, research on hybrid precoding algorithms with subconnection structures has also become the focus of attention in recent years. For the subconnected structure, an AltMin algorithm is also developed with the help of semidefinite relaxation in [9]. Similarly, a new divide-and-conquer precoding scheme is proposed in [10]. The performance of this scheme is close to that of SDR-AltMin, which can save time effectively and is robust to potential saddle points with poor performance. In view of the hybrid precoding problem of energy-saving subconnection structure, a hybrid precoding scheme based on successive interference cancellation (SIC) is proposed in [11]. It does not require singular value decomposition and matrix inversion, and the computational complexity is much lower than traditional sparse reconstruction precoding algorithms. In [12], an optimization model depends on the minimum mean square error (MMSE) is established, and then, a low complexity hybrid precoding algorithm based on particle swarm ant colony optimization (PSACO) is proposed. The high performance hybrid precoding algorithm for subconnection structures is studied in [13], and the performance of the algorithm is better than that of the SIC algorithm when the channel state information is not perfect. A new subconnection structure is introduced in [14] to further reduce the power consumption of the system. According to this structure, an efficient hybrid precoding scheme based on service quality constraint is proposed. A general hybrid precoding algorithm is studied in [15], in which the supplementary matrix is introduced, and the regularization zero-forcing method is used to solve the problem.

Synthesizing the current research situation, most of the literature studies the ideal fully connection structure, but in practice, the structure has high complexity and energy consumption. Although the performance is considerable, the practical application is poor. Therefore, some scholars have gradually focused on the energy-saving and more practical subconnection structure, thus far, also achieved some results, but the performance of the hybrid precoding algorithm and the complexity of the algorithm still need to be further improved. Different from the traditional full-digital precoding, hybrid precoding can only control the phase of the data stream in the analog domain, but cannot adjust its amplitude. The constant modulus constraint of the analog precoding matrix makes the solution of the hybrid precoding more difficult.

Thus, in order to simplify the problem, we consider solving the analog precoding matrix and the digital precoding matrix, respectively, as in most references [11, 13]. The total achievable rate  $R$  of the system is expanded directly, and the optimization problem is decomposed into a series of subrate optimization by recursion in [11, 15]. Different from them, this paper starts with the equivalent problem of maximizing the achievable rate, the Euclidean distance derived in [5] to solve the optimization problem. We utilize the block diagonal form of the analog precoding and decompose the optimization problem into a series of optimization subproblems and solved in turn. The subproblem, which is each column of the analog precoding matrix, is derived from the corresponding Euclidean distance expansion. After the complete analog precoding matrix is obtained, the corresponding digital precoding matrix optimization problem is easy to solve. The simulation results also show that the algorithm can achieve considerable performance regardless of whether the acquired channel state information is perfect or not.

The rest of this paper is arranged as follows. Section 2 introduces the model of mmWave massive MIMO system and the channel model used. Section 3 describes the design process of the proposed algorithm in detail. In Section 4, the simulation results are given, and the performance of the proposed algorithm is analyzed. Finally, we conclude this paper in Section 5.

*Notation:* in this paper,  $\mathbf{A}$ ,  $\mathbf{a}$ , and  $a$  denote a matrix, a vector, and a scalar, respectively;  $(\mathbf{A})^T$ ,  $(\mathbf{A})^H$ , and  $\|\mathbf{A}\|_F$  are the transpose, conjugate transpose, and Frobenius norm of  $\mathbf{A}$ , respectively;  $\mathbf{A}^{-1}$  and  $\mathbf{A}^\dagger$  represent the inverse and Moore-Penrose pseudo inverse of  $\mathbf{A}$ ;  $\text{Tr}(\mathbf{A})$  indicates the trace of matrix  $\mathbf{A}$ ;  $\det(\mathbf{A})$  expresses the determinant of  $\mathbf{A}$ ;  $\text{diag}(\mathbf{A})$  is the diagonal version of  $\mathbf{A}$ ;  $\text{abs}(\mathbf{A})$  is the element-wise absolute value of  $\mathbf{A}$ ;  $\mathbf{A} \oslash \mathbf{B}$  is the Hadamard division between  $\mathbf{A}$  and  $\mathbf{B}$ ;  $\mathcal{CN}(0, \sigma^2)$  is a complex Gaussian vector with mean 0 and covariance  $\sigma^2$ ;  $E(\cdot)$  denotes the expectation;  $\mathbb{C}^{m \times n}$  denotes an  $m \times n$  dimensional complex space;  $\Re(a)$  is the real part of  $a$ .

## 2. System Model and Channel Model

*2.1. System Model.* We focus on subconnection architecture, which can avoid the use of a large number of phase shifters,

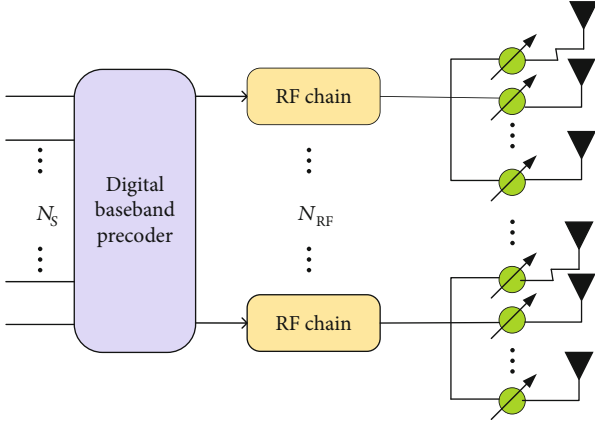


FIGURE 1: Subconnected architecture.

as shown in Figure 1. Consider a single-user mmWave massive MIMO system as shown in Figure 2. The transmitter and the receiver are equipped with  $N_t$  and  $N_r$  antennas, respectively [16].  $N_s$  data streams in the baseband are first precoded by a digital precoder  $\mathbf{D}$  and then precoded by an analog precoder after passing through the corresponding RF chain. After that, each data stream is transmitted by a subantenna array with only  $M$  antennas associated with the corresponding RF chain. The number of RF chains at the transmitter and the receiver is  $N_{RF}^t$  and  $N_{RF}^r$ , respectively. To achieve multiple data transmission, the following constraints must be satisfied:  $N_s \leq N_{RF}^t \leq N_t$  and  $N_s \leq N_{RF}^r \leq N_r$ .

The hybrid precoder is composed of a digital baseband precoder  $\mathbf{D}$  and an analog precoder  $\mathbf{A}$ , and their dimensions are  $N_{RF}^t \times N_s$  and  $N_t \times N_{RF}^t$ . Assuming that the initial signal is  $\mathbf{s}$ , the transmission signal can be written as  $\mathbf{x} = \mathbf{A}\mathbf{D}\mathbf{s}$ , where  $\mathbf{s}$  is the  $N_s \times 1$  symbol vector and satisfies  $E[\mathbf{s}\mathbf{s}^T] = 1/N_s \mathbf{I}_{N_s}$ .  $\|\mathbf{A}\mathbf{D}\|_F^2 = N_s$  is the normalized power constraint of the system [17]. The received signal vector  $\mathbf{y} = [y_1, y_2, \dots, y_K]^T$  of the system can be expressed as

$$\mathbf{y} = \sqrt{\rho}\mathbf{H}\mathbf{x} + \mathbf{n} = \sqrt{\rho}\mathbf{H}\mathbf{A}\mathbf{D}\mathbf{s} + \mathbf{n}, \quad (1)$$

where  $\rho$  is the average received power;  $\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$  denotes the channel matrix, the baseband transmission signal vector can be expressed as  $\mathbf{s} = [s_1, s_2, \dots, s_{N_s}]^T$ , and  $\mathbf{A}$  and  $\mathbf{D}$  represent the analog precoding matrix and the digital precoding matrix, respectively.  $\mathbf{n}$  denotes the noise vector with independently and identically distributed i.i.d.  $\mathcal{CN}(0, \sigma^2)$  entries.  $\mathbf{F} = \mathbf{A}\mathbf{D}$  presents the hybrid precoding matrix of size  $N_t \times N_r$ , which satisfies  $\|\mathbf{F}\|_F \leq N_s$  to meet the total transmit power constraint.

In subconnection architecture, each RF chain is connected to  $M$  ( $M = N_t/N_{RF}$ ) antenna via  $M$  phase shifter [18]. Therefore, the digital precoding matrix  $\mathbf{D}$  in this architecture is the diagonal matrix,  $\mathbf{D} = \text{diag}[\bar{d}_1, \bar{d}_2, \dots, \bar{d}_N]$ , where  $\bar{d}_n \in \mathbb{R}, n = 1, 2, \dots, N$ . The corresponding analog precoding matrix is a block diagonal matrix, which is represented as

$$\mathbf{A} = \begin{bmatrix} \bar{\mathbf{a}}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{a}}_2 & & \mathbf{0} \\ \vdots & & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \bar{\mathbf{a}}_N \end{bmatrix}_{NM \times N} \quad (2)$$

$\bar{\mathbf{a}}_n \in \mathbb{C}^{M \times 1}$  can be regarded as an analog weighting vector. Due to the constant modulus constraint, each element in the  $\bar{\mathbf{a}}_n$  has the same amplitude and different phases [18].

**2.2. Channel Model.** Unlike traditional low frequency channels, the propagation characteristics of millimeter-wave channels no longer subject to Rayleigh fading. Due to the high path loss in free space, the spatial selectivity of millimeter-wave propagation is limited [19]. In addition, large-scale dense antenna arrays will result in obvious antenna correlation in millimeter-wave channels [20]. Therefore, the traditional channel model is not suitable for the millimeter-wave channel in the massive MIMO system. For simplicity, this paper adopts the geometric Saleh-Valenzuela model in most literature [20, 21]. The channel matrix is given by

$$\mathbf{H} = \sqrt{\frac{N_t N_r}{N_{cl} N_{ray}}} \sum_{i=1}^{N_{cl}} \sum_{k=1}^{N_{ray}} \alpha_{ik} \mathbf{a}_r(\phi_{ik}^r, \theta_{ik}^r) \mathbf{a}_t(\phi_{ik}^t, \theta_{ik}^t)^H, \quad (3)$$

where  $N_{cl}$  is the number of scattering clusters and each cluster contributes  $N_{ray}$  propagation paths.  $\alpha_{ik}$  is the gain of the  $k$ th ray in the  $i$ th scattering cluster, and it is subject to i.i.d.  $\mathcal{CN}(0, \sigma_{\alpha,i}^2)$ , where  $\sigma_{\alpha,i}^2$  indicates the power of the  $i$ th clusters [22].  $\mathbf{a}_r(\phi_{ik}^r, \theta_{ik}^r)$  and  $\mathbf{a}_t(\phi_{ik}^t, \theta_{ik}^t)$  represent the normalized antenna array response vectors at the receiver and transmitter, respectively.  $\phi_{ik}^r(\theta_{ik}^r)$  and  $\phi_{ik}^t(\theta_{ik}^t)$  are the azimuth (elevation) angles of the  $k$ th ray in the  $i$ th scattering cluster on the receiving and transmitting sides, respectively. The mean cluster angles  $\phi_{ik}^r(\theta_{ik}^r)$  and  $\phi_{ik}^t(\theta_{ik}^t)$  are subject to uniform-random distribution in the range of  $[0, 2\pi]$  [23]. Within the  $i$ th cluster, the arrival and departure angles of each ray follow Laplace distribution with mean cluster angles  $\phi_{ik}^r(\theta_{ik}^r)$  and  $\phi_{ik}^t(\theta_{ik}^t)$  as the standard deviation.

The antenna array response vector depends on the structure of the antenna array at the transmitter and receiver. There are two types of antenna array structures: uniform linear array (ULA) and uniform planar array (UPA). For millimeter-wave massive MIMO systems, a uniform planar array is conducive to the packaging of antennas at the transceiver, and the realization of beamforming will be more flexible. On the other hand, as the UPA array can bring better energy efficiency and spectrum efficiency, we mainly consider the UPA. In the case of a uniform planar array with horizontal  $W$  antenna elements and vertical  $H$  antenna elements, the array response vector is given by [24]



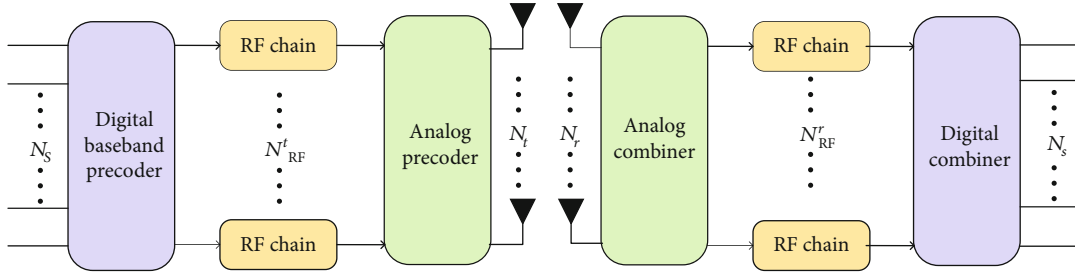


FIGURE 2: Single-user mmWave massive MIMO system.

$$\mathbf{a}_{UPA}(\phi, \theta) = \frac{1}{\sqrt{W \times H}} \cdot \left[ 1, \dots, e^{j(2\pi/\lambda)d(m \sin \phi \sin \theta + n \cos \theta)}, \dots, e^{j(2\pi/\lambda)d((W-1) \sin \phi \sin \theta + (H-1) \cos \theta)} \right]^T, \quad (4)$$

where  $\lambda$  is the wavelength of the carrier signal,  $d$  is the distance between adjacent antennas, and  $m$  and  $n$  satisfy  $0 \leq m < W$  and  $0 \leq n < H$ , respectively.

### 3. Hybrid Precoding Algorithm Design

In this paper, an approximate optimal hybrid precoding algorithm is proposed for the subconnection structure. Because the joint optimization problem can be temporarily decoupled [5], this paper mainly discusses the design of the hybrid precoding algorithm at the transmitter of the system.

The fundamental goal of the hybrid precoding algorithm is to maximize the spectral efficiency  $R$  by designing the optimal hybrid precoder. The spectral efficiency of the system is expressed as [5] follows:

$$R = \log_2 \left( \left| \mathbf{I}_{N_s} + \frac{\rho}{N_s \sigma^2} \mathbf{H} \mathbf{F} \mathbf{F}^H \mathbf{H}^H \right| \right), \quad (5)$$

where  $\mathbf{I}_{N_s}$  denotes a  $N_s \times N_s$  unit matrix and  $\mathbf{F}$  presents the hybrid precoding matrix with block diagonal form. Thus, the hybrid precoder optimization problem in this paper can be expressed as

$$\begin{aligned} (\mathbf{A}^{\text{opt}}, \mathbf{D}^{\text{opt}}) &= \arg \max_{\mathbf{A}, \mathbf{D}} R, \\ \text{s.t. } \mathbf{A} &\in \mathcal{A}, \\ \|\mathbf{A}\mathbf{D}\|_F^2 &= N_s. \end{aligned} \quad (6)$$

Hybrid precoding introduces phase shifters in the analog domain to process the transmitted signal. Since the phase shifter can only adjust the phase of the signal and cannot adjust the amplitude of the signal, the nonzero terms in the analog precoding matrix need to have the same modulus [18].  $\mathcal{A}$  is used to represent the set of  $\mathbf{A}$  matrices that satisfy the constant modulus constraint. Obviously, problem (6) belongs to nonconvex optimization problem due to the existence of constant modulus constraint. Although the constraints make it difficult to obtain the exact solution directly, the approximate optimal mixed precoding matrix

can be obtained by solving the approximation of (6). As described in [5], problem (6) can be further equated to problem (7):

$$\begin{aligned} (\mathbf{A}^{\text{opt}}, \mathbf{D}^{\text{opt}}) &= \arg \min_{\mathbf{A}, \mathbf{D}} \|\mathbf{F}^{\text{opt}} - \mathbf{A}\mathbf{D}\|_F^2, \\ \text{s.t. } \mathbf{A} &\in \mathcal{A}, \\ \|\mathbf{A}\mathbf{D}\|_F^2 &= N_s. \end{aligned} \quad (7)$$

The Euclidean distance is used to make the optimal mixed precoding as close as possible to the unconstrained optimal digital precoding. It is shown in reference [5] that the objective function in the minimization problem (7) can maximize the spectral efficiency. The optimal mixed precoding matrix in the problem can be obtained by singular value decomposition of the channel matrix. The channel matrix singular value decomposition is represented by  $\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$ , and the unconstrained optimal precoding matrix  $\mathbf{F}^{\text{opt}}$  consists of the first  $N_s$  columns of the right singular matrix.

Problem (7) is fundamentally aimed at obtaining approximately optimal analog precoding matrix  $\mathbf{A}$  and digital precoding matrix  $\mathbf{D}$ . The constant mode constraint in (7) makes it difficult to solve the problem directly. Thanks to the special block diagonal form of the hybrid precoding matrix; it can be seen that the different subarrays are not correlated. This allows us to decompose the overall optimization problem into a series of suboptimization problems, each of which can be represented by  $\mathbf{f}_n = \bar{d}_n \bar{\mathbf{a}}_n$ .

$$\begin{aligned} \mathbf{F} = \mathbf{A}\mathbf{D} &= \text{diag} [\bar{\mathbf{a}}_1, \bar{\mathbf{a}}_2, \dots, \bar{\mathbf{a}}_N] \times \text{diag} [\bar{d}_1, \bar{d}_2, \dots, \bar{d}_N] \\ &= \text{diag} [\bar{\mathbf{f}}_1, \bar{\mathbf{f}}_2, \dots, \bar{\mathbf{f}}_N] = \begin{bmatrix} \bar{\mathbf{f}}_1 & 0 & \dots & 0 \\ 0 & \bar{\mathbf{f}}_2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \bar{\mathbf{f}}_N \end{bmatrix}, \end{aligned} \quad (8)$$

where  $\bar{\mathbf{f}}_n$  is the  $n$  column of the hybrid precoding matrix  $\mathbf{F}$  and  $\bar{d}_n$  is the nonzero element in the  $n$  column, which is the nonzero element matrix of column matrix of  $\mathbf{F}$ . The concrete structure of  $\bar{\mathbf{f}}_n$  is shown as

$$\bar{\mathbf{f}}_n = \left[ \mathbf{0}_{1 \times M(n-1)}, \bar{d}_n, \mathbf{0}_{1 \times M(N-n)} \right]^T. \quad (9)$$

It is obvious from formula (8) that the design of hybrid precoder  $\mathbf{F}$  can be converted into optimizing analog precoder  $\mathbf{A}$  and digital precoder  $\mathbf{D}$  separately, or the problem is equivalent to optimizing each independent subantenna array  $\bar{\mathbf{f}}_n$ . Therefore, we assume that the digital precoding matrix  $\mathbf{D}$  is fixed and first optimize the analog precoding matrix  $\mathbf{A}$ . The optimization problem is expressed as

$$\begin{aligned} \mathbf{A}^{\text{opt}} &= \arg \min_{\mathbf{A}} \|\mathbf{F}^{\text{opt}} - \mathbf{A}\mathbf{D}\|_F^2, \\ \text{s.t. } \mathbf{A} &\in \mathcal{A}, \\ \|\mathbf{A}\mathbf{D}\|_F^2 &= N_s. \end{aligned} \quad (10)$$

Since the hybrid precoding matrix of subconnected structure is a special block-diagonal form, the total optimization problem is decomposed into a series of independent subarray optimization problems. The power constraint in (10) can be scaled to meet the constraint after the analog precoder and the digital precoder are optimized. With the help of (8), the subarray optimization problem is described as

$$\begin{aligned} \bar{\mathbf{a}}_n^{\text{opt}} &= \arg \min_{\bar{\mathbf{a}}_n} \|\mathbf{F}_n^{\text{opt}} - \bar{\mathbf{a}}_n \bar{d}_n\|_F^2, \\ \text{s.t. } |(\bar{\mathbf{a}}_n)_m| &= 1. \end{aligned} \quad (11)$$

As can be seen from the objective function (11), the optimization problem of the analog precoding matrix  $\mathbf{A}^{\text{opt}}$  is further decomposed into optimization of nonzero submatrix  $\bar{\mathbf{a}}_n$ . The objective function of the (11) suboptimization problem is expanded as

$$\begin{aligned} \|\mathbf{F}_n^{\text{opt}} - \bar{\mathbf{a}}_n \bar{d}_n\|_F^2 &= \text{Tr} \left( (\bar{d}_n \bar{\mathbf{a}}_n^H - \mathbf{F}_n^{\text{opt}})^H (\bar{d}_n \bar{\mathbf{a}}_n^H - \mathbf{F}_n^{\text{opt}}) \right) \\ &= \text{Tr} \left( (\mathbf{F}_n^{\text{opt}})^H (\mathbf{F}_n^{\text{opt}}) \right) - 2\Re \left( \text{Tr} \left( \bar{d}_n \bar{\mathbf{a}}_n^H \mathbf{F}_n^{\text{opt}} \right) \right) \\ &\quad + \text{Tr} \left( \bar{d}_n \bar{\mathbf{a}}_n^H \bar{\mathbf{a}}_n \bar{d}_n \right) = \text{Tr} \left( (\mathbf{F}_n^{\text{opt}})^H (\mathbf{F}_n^{\text{opt}}) \right) \\ &\quad - 2\Re \left( \text{Tr} \left( \bar{\mathbf{a}}_n^H \mathbf{F}_n^{\text{opt}} \bar{d}_n^H \right) \right) + \text{Tr} \left( \bar{d}_n \bar{\mathbf{a}}_n^H \bar{\mathbf{a}}_n \bar{d}_n \right) \\ &= \text{Tr} \left( (\mathbf{F}_n^{\text{opt}})^H (\mathbf{F}_n^{\text{opt}}) \right) - 2\Re \left( \bar{d}_n \text{Tr} \left( \bar{\mathbf{a}}_n^H \mathbf{F}_n^{\text{opt}} \right) \right) + M \bar{d}_n^2. \end{aligned} \quad (12)$$

$\text{Tr}(AB) = \text{Tr}(BA)$  and  $\bar{\mathbf{a}}_n^H \bar{\mathbf{a}}_n = M$  are used in the expansion of the equation (12). Because all the terms in  $\bar{\mathbf{a}}_n$  have the same amplitude 1,  $\bar{\mathbf{a}}_n^H \bar{\mathbf{a}}_n = M$  is established apparently.

From (12), we can see that the expansion result is made up of three parts, in which the first term  $\text{Tr}((\mathbf{F}_n^{\text{opt}})^H (\mathbf{F}_n^{\text{opt}}))$  and the last term  $M \bar{d}_n^2$  are fixed, and the goal of the problem is converted to maximize the real part of  $\bar{d}_n \text{Tr}(\bar{\mathbf{a}}_n^H \mathbf{F}_n^{\text{opt}})$ , that is, to maximize the real part of  $\bar{\mathbf{a}}_n^H \mathbf{F}_n^{\text{opt}}$ . Under constant modulus constraint, when the phase of each term of the  $\bar{\mathbf{a}}_n$  is equal to the phase of  $\mathbf{F}_n^{\text{opt}}$  corresponding term, the  $\bar{\mathbf{a}}_n^H \mathbf{F}_n^{\text{opt}}$  contains only the real part and reaches the maximum. Therefore, the optimal solution of  $\bar{\mathbf{a}}_n$  can be obtained by preserving the phase of the term in  $\mathbf{F}_n^{\text{opt}}$  and normalizing it.

$$\bar{\mathbf{a}}_n = \mathbf{F}_n^{\text{opt}} \oslash \text{abs}(\mathbf{F}_n^{\text{opt}}), \quad (13)$$

If  $n < N_{\text{RF}}^t$ , then set  $n = n + 1$  and continue to solve the next subproblem in a similar way. By solving  $\bar{\mathbf{a}}_n$  successively, the complete optimized analog precoding matrix  $\mathbf{A} = \text{diag}[\bar{\mathbf{a}}_1, \bar{\mathbf{a}}_2, \dots, \bar{\mathbf{a}}_{N_{\text{RF}}^t}]$  can be obtained. When the analog precoding matrix is determined, the corresponding optimal digital precoding matrix  $\mathbf{D}$  can be easily obtained by the minimum mean square error (MMSE).

$$\mathbf{D} = \mathbf{A}^\dagger \mathbf{F}^{\text{opt}}. \quad (14)$$

Finally, the digital precoding matrix is normalized to meet the previously ignored power constraints.

$$\mathbf{D} = \frac{\sqrt{N_s}}{\|\mathbf{A}\mathbf{D}\|_F} \mathbf{D}. \quad (15)$$

The complete process of hybrid precoding design is shown in Algorithm 1.

We have  $\mathbf{W}_{\text{RF}}^H \mathbf{W}_{\text{RF}} \propto \mathbf{I}$  on the receiving side, while the digital combining matrix satisfies  $\mathbf{W}_{\text{BB}}^H \mathbf{W}_{\text{BB}} \propto \mathbf{I}$ . If  $\mathbf{H}_{\text{eff}} = \mathbf{H}\mathbf{A}\mathbf{D}$  is considered as an equivalent matrix, the spectral efficiency can be given by [13]

$$\begin{aligned} R &= \log_2 \det \left( I_{N_s} + \frac{\rho}{N_s \sigma_n^2} (\mathbf{W}_{\text{RF}} \mathbf{W}_{\text{BB}})^H \mathbf{H}_{\text{eff}} \mathbf{H}_{\text{eff}}^H \mathbf{W}_{\text{RF}} \mathbf{W}_{\text{BB}} \right) \\ &= \log_2 \det \left( I_{N_s} + \frac{\rho}{N_s \sigma_n^2} \mathbf{H}_{\text{eff}}^H \mathbf{W}_{\text{RF}} \mathbf{W}_{\text{BB}} (\mathbf{W}_{\text{RF}} \mathbf{W}_{\text{BB}})^H \mathbf{H}_{\text{eff}} \right). \end{aligned} \quad (16)$$

The spectrum efficiency of the receiver and the mutual information of the transmitter have similar structures, so the analog receiver and the digital receiver can be designed in the same way as the transmitter.

The algorithm proposed in this paper is to iteratively solve a single column vector. As mentioned in [11], the expected performance of the algorithm can be achieved when the number of the iterations is set to 5. The HPD-PS algorithm in [13] requires  $5N_s$  iterations, and the magnitude of the major complexity of each iteration is  $\mathcal{O}(N_t^2/N_{\text{RF}}^2)$ , while the complexity of each iteration of the SIC algorithm in [11] is  $\mathcal{O}(N_t^2/N_s^2)$ . The main complexity of each iteration of the algorithm presented in this paper is about  $\mathcal{O}(2N_t N_s / N_{\text{RF}}^t)$ . It can be seen that compared with other subconnection structure hybrid precoding algorithms, the computational complexity of the algorithm in this paper is not high.

In addition, for mmWave communications, one nature is broadband. Therefore, the case of extending the algorithm to wideband communication is briefly discussed here. The same as in narrowband systems, the extra orthogonality constraint on the digital precoder also has been negligible impact on spectral efficiency in mmWave OFDM systems [9]. That is to say, the main difference between the hybrid precoding schemes of wideband and narrowband systems is the design of analog precoding. In mmWave MIMO-OFDM systems, the signals of all subcarriers usually need to share an analog

```

Input:  $\mathbf{H}$ 
1:  $[\tilde{\cdot}, \tilde{\cdot}, \mathbf{V}_d] = \text{svd}(\mathbf{H})$ ;
2:  $\mathbf{F}^{opt} = (\mathbf{V}_d)_{:,1:N_s}$ ;
3: for  $l \leq L$  do
4:   for  $n \leq N_{RF}$  do
5:      $\bar{\mathbf{a}}_n = \mathbf{F}_n^{opt} \oslash \text{abs}(\mathbf{F}_n^{opt})$ 
6:   end for
7: end for
8:  $\mathbf{A} = \text{diag}[\bar{\mathbf{a}}_1, \bar{\mathbf{a}}_2, \dots, \bar{\mathbf{a}}_N]$ 
9: Compute  $\mathbf{D}$  according to (16);
10: Normalize  $\mathbf{D}$  with (17);
11: return  $\mathbf{A}, \mathbf{D}$ 

```

ALGORITHM 1: Spectrum Efficient Hybrid Precoding Design Based on Subconnected Architecture.

precoder. Thus, the solution of analog precoding can be extended to wideband system by carrier aggregation and other related operations.

#### 4. Simulation Results

In this section, the performance of the proposed algorithm is simulated and compared with the hybrid precoding algorithm based on continuous interference cancellation in [11] and the HPD-PS algorithm in [13]. In the result diagram, for reference, the performance of full-digital precoding and full-analog precoding for subconnection structures is given.

In this simulation, the default parameters are set as follows. The channel matrix is generated from the channel description above. We model the propagation environment as a  $N_{cl} = 8$  cluster environment with  $N_{ray} = 10$  rays per cluster [24]. The angular spread of each cluster is 10 degrees. The carrier frequency is set as 28 GHz [11]. The UPA array is used for the antenna of transmitter and receiver, and the distance between the adjacent antennas is  $d = \lambda/2$ . We assumed that the AODs follow the uniform distribution within  $[-\pi/6, \pi/6]$ , while the AOA follows the uniform distribution within  $[-\pi, \pi]$  [22]. Moreover, the maximum number of iterations is set to 5.

*4.1. Performance under Perfect CSI.* We first consider the ideal state, that is, the millimeter-wave massive MIMO communication scenario in which the CSI can be obtained. Figure 3 shows a comparison of spectral efficiency in a millimeter-wave MIMO system with  $NM \times N_r = 64 \times 16$ , and the number of RF chain is 8. Obviously, the subconnection structure algorithm proposed in this paper is better than the SIC hybrid precoding algorithm in the whole analog signal-to-noise ratio range. In addition, the spectral efficiency of the proposed algorithm is close to that of the optimal unconstrained full-digital precoding algorithm with subconnected structure.

Considering that the continuous interference cancellation algorithm was proposed in 2016, Figure 4 further shows that when the antenna size of the transmitter is  $N_t = 144$  and the receiver is  $N_r = 16$ ,  $N = 4$ , the proposed algorithm is compared with the spectrum efficiency of the hybrid precoding algorithm based on successive interference cancellation in [11] and the HPD-PS algorithm in [13]. It has been shown

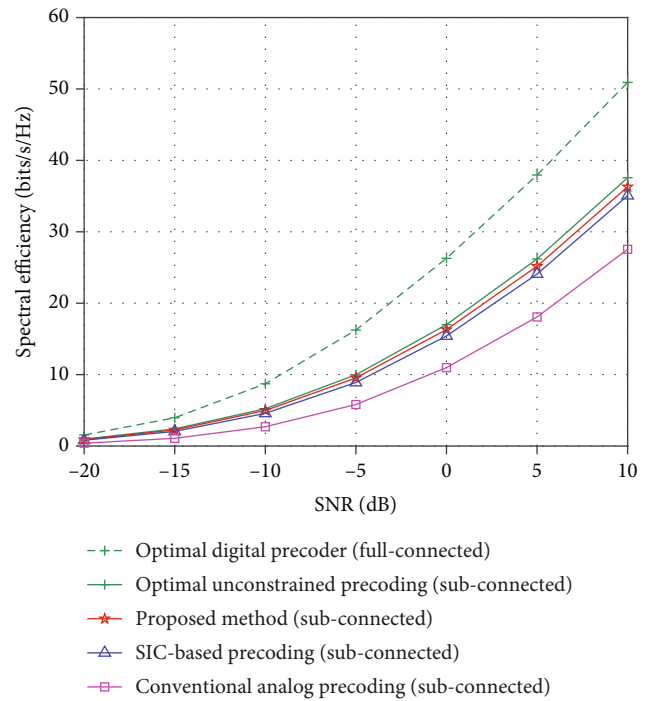


FIGURE 3: Spectral efficiency comparison of different algorithms, when  $NM \times N_r = 64 \times 16$  ( $N = 8$ ).

in [13] that the performance of HPD-PS algorithm is better than that of SIC hybrid precoding algorithm, and the result can be seen in Figure 4. In addition, the hybrid precoding algorithm in this paper is much closer to the unconstrained full-digital precoding algorithm than the HPD-PS algorithm.

There is a gap of 5 dB between the optimal fully connected method and the optimal unconstrained precoding for subconnected structure. However, the subconnection structure is much lower in cost and hardware complexity than the full-connection structure. Taking the system model of this article as an example, the fully connected structure requires at least  $N^2M$  phase shifters, while the subconnected structure only needs  $NM$  phase shifters.

*4.2. Impact of Imperfect CSI.* However, in the actual communication process, due to the lack of cooperation between the

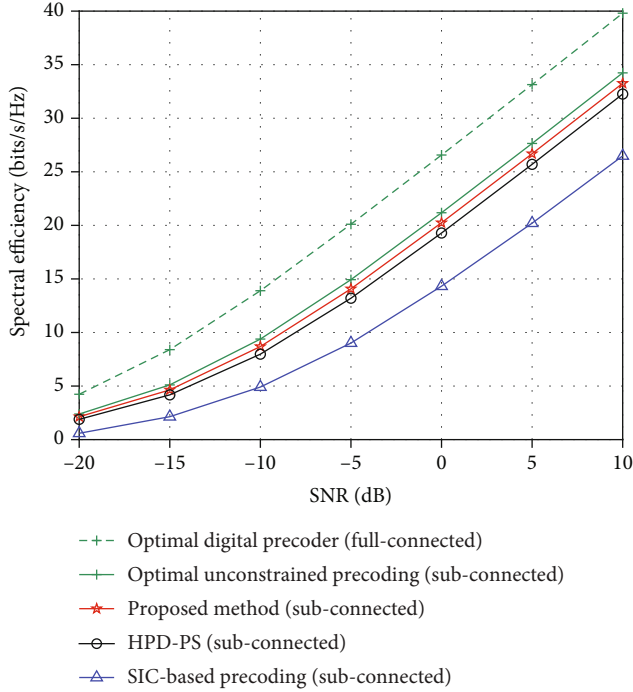


FIGURE 4: Spectral efficiency comparison of different algorithms, when  $NM \times N_r = 144 \times 16 (N = 4)$ .

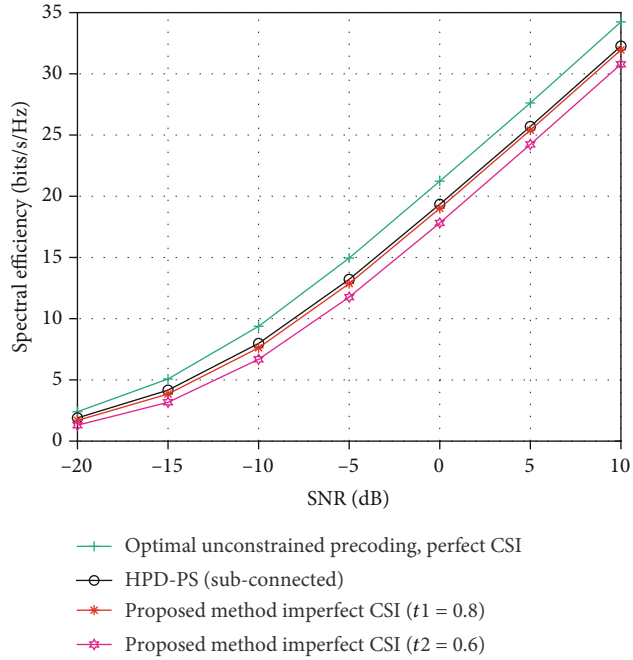


FIGURE 5: Spectral efficiency by different algorithms under the perfect and the imperfect.

base station and the user, and some uncertain factors, such as uncertain channel estimation and limited feedback, perfect channel state information is difficult to obtain. Thus, the preprocessing of the transmit signal of the base station cannot completely eliminate the interference between the users as under the accurate channel state information, and the users cannot be used as the interference elimination strategy at

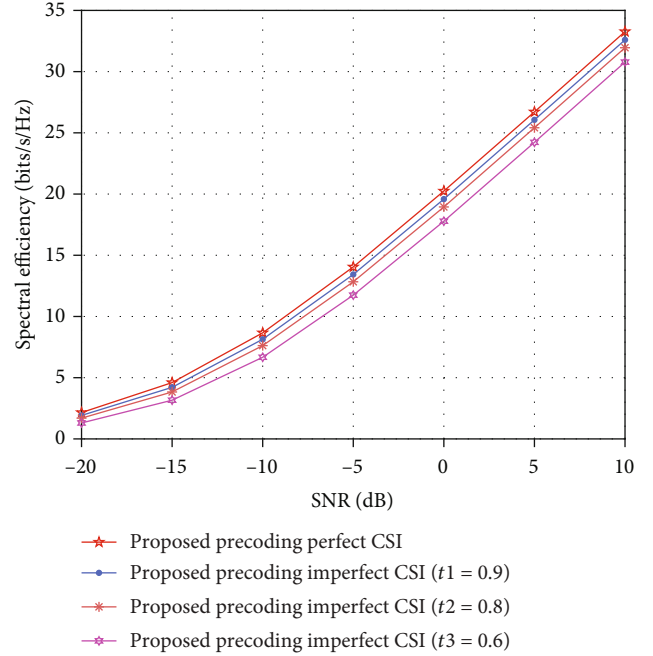


FIGURE 6: Impact of imperfect CSI on the proposed hybrid precoding algorithm.

the receiving end due to the limitation of the number of receiving antennas. No matter how accurate the channel estimation algorithm is, there will always be a fixed error bound, which cannot meet the specified arbitrary accuracy requirements [25, 26]. Therefore, it is of great research and practical value to analyze the effect of imperfect channel state information on downlink multiuser MIMO system performance and the method of obtaining and utilizing channel state information effectively.

The perfect result cannot be guaranteed by data acquisition and processing in communication network [27–30]. Although there are a lot of channel estimation algorithms, still with the ideal state, the perfect channel state information estimation has a certain gap. In the case of imperfect channel state information, the performance of the proposed hybrid precoding algorithm is compared. For the estimation of channel state information, the minimum mean square error method is used, and the expression is as [31] follows:

$$\hat{\mathbf{H}} = t\mathbf{H} + \sqrt{1-t^2}\mathbf{E}, \quad (17)$$

where  $\hat{\mathbf{H}}$  is the estimated channel matrix,  $0 \leq t \leq 1$  presents the CSI accuracy, and  $\mathbf{E}$  is the error matrix with entries following the distribution i.i.d.  $\mathcal{E}\mathcal{N}(0, 1)$ .

Figure 5 shows the performance comparison of the proposed algorithm with HPD-PS and optimal unconstrained full-digital precoding under imperfect channel state information and perfect channel state information. Obviously, the performance of this algorithm is not as good as that of HPD-PS when  $t = 0.6$ , but it is close to that of HPD-PS when  $t = 0.8$ .

Other than that, Figure 6 shows the detailed performance simulation results of the proposed algorithm under imperfect

channel state information. As can be seen from the graph, the spectral efficiency of the algorithm is relatively stable from the perfect channel state information to  $t = 0.8$ , and even if  $t = 0.6$ , the performance of the algorithm does not decrease greatly.

## 5. Conclusions

In this paper, we mainly focus on the hybrid precoding of millimeter-wave large-scale MIMO systems based on sub-connection structures. Firstly, the hybrid precoding optimization problem is decoupled into analog precoding and digital precoding. In the optimization process, the digital precoding matrix is first fixed to solve the optimal analog precoder. Thanks to the block diagonal form of the analog precoding matrix; the optimization problem can be decomposed into a series of optimization subproblems and solved in turn. Finally, the optimal digital precoding matrix can be obtained by MMSE. The simulation results show that the spectrum efficiency of the proposed algorithm is better than that of SIC and HPD-PS. Considering the fact that the channel state information cannot be obtained completely, the performance of the algorithm under imperfect channel state information is simulated and analyzed. It can be seen that the performance degradation of the algorithm is acceptable even under imperfect CSI and is similar to the performance of full-digital precoding under perfect channel state information.

## 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 known competing financial interests or personal relationships that could have appeared to influence the work reported in 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. K. Yong and C.-C. Chong, "An overview of multigigabit wireless through millimeter wave technology: potentials and technical challenges," *EURASIP Journal on Wireless Communications and Networking*, vol. 2007, no. 1, 10 pages, 2006.
- [2] W. H. Chin, Z. Fan, and R. Haines, "Emerging technologies and research challenges for 5G wireless networks," *IEEE Wireless Communications*, vol. 21, no. 2, pp. 106–112, 2014.
- [3] F. Al-Ogaili and R. M. Shubair, "Millimeter-wave mobile communications for 5G: challenges and opportunities," in *2016 IEEE International Symposium on Antennas and Propagation (APSURSI)*, pp. 1003–1004, Fajardo, PR, USA, 2016.
- [4] S. A. Busari, K. M. S. Huq, S. Mumtaz, L. Dai, and J. Rodriguez, "Millimeter-wave massive MIMO communication for future wireless systems: a survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 836–869, 2018.
- [5] O. E. Ayach, S. Rajagopal, S. Abu-Surra, Z. Pi, and R. W. Heath, "Spatially sparse precoding in millimeter wave MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 13, no. 3, pp. 1499–1513, 2014.
- [6] R. Méndez-Rial, C. Rusu, N. González-Prelcic, and R. W. Heath, "Dictionary-free hybrid precoders and combiners for mmWave MIMO systems," in *2015 IEEE 16th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 151–155, Stockholm, Sweden, 2015.
- [7] F. Zhang and M. Wu, "Hybrid analog-digital precoding for millimeter wave MIMO Systems," in *2017 IEEE 17th International Conference on Communication Technology (ICCT)*, Chengdu, China, 2017.
- [8] X. Liu, X. Li, S. Cao et al., "Hybrid precoding for massive mmWave MIMO systems," *IEEE Access*, vol. 7, pp. 33577–33586, 2019.
- [9] X. Yu, J. Shen, J. Zhang, and K. B. Letaief, "Alternating minimization algorithms for hybrid precoding in millimeter wave MIMO systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 485–500, 2016.
- [10] K. Xu, F. Zheng, P. Cao, H. Xu, and X. Zhu, "A divide-and-conquer precoding scheme for sub-connected massive MIMO systems," in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, pp. 1–5, Honolulu, HI, USA, 2019.
- [11] X. Gao, L. Dai, S. Han, C. L. I, and R. W. Heath, "Energy-efficient hybrid analog and digital precoding for MmWave MIMO systems with large antenna arrays," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 4, pp. 998–1009, 2016.
- [12] Y. Chen, Y. Xia, Y. Xing, and L. Yang, "Low complexity hybrid precoding for mmWave massive MIMO systems," in *2017 26th Wireless and Optical Communication Conference (WOCC)*, pp. 1–5, Newark, NJ, USA, 2017.
- [13] Y. Wang and W. Zou, "Hybrid precoding design for millimeter wave systems with the partially-connected structure," *IET Communications*, vol. 14, no. 4, pp. 561–567, 2020.
- [14] W. Dong, T. Zhang, Z. Hu, Y. Liu, and X. Han, "Energy-efficient hybrid precoding for mmWave massive MIMO systems," in *2018 IEEE/CIC International Conference on Communications in China (ICCC Workshops)*, pp. 6–10, Beijing, China, 2018.
- [15] Y. Xie, B. Li, Z. Yan, J. Fan, and M. Yang, "A general hybrid precoding method for mmWave massive MIMO systems," *Radioengineering*, vol. 27, no. 2, pp. 439–446, 2019.
- [16] P. Xia, S.-K. Yong, J. Oh, and C. Ngo, "A practical SDMA protocol for 60 GHz millimeter wave communications," in *2008 42nd Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, USA, 2008.
- [17] S. Han, C.-I. I, Z. Xu, and C. Rowell, "Large-scale antenna systems with hybrid analog and digital beamforming for millimeter wave 5G," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 186–194, 2015.
- [18] S. Hur, T. Kim, D. J. Love, J. V. Krogmeier, T. A. Thomas, and A. Ghosh, "Millimeter wave beamforming for wireless backhaul and access in small cell networks," *IEEE Transactions on Communications*, vol. 61, no. 10, pp. 4391–4403, 2013.

- [19] H. Xu, V. Kukshya, and T. S. Rappaport, "Spatial and temporal characteristics of 60-GHz indoor channels," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 3, pp. 620–630, 2002.
- [20] V. Raghavan and A. M. Sayeed, "Sublinear capacity scaling laws for sparse MIMO channels," *IEEE Transactions on Information Theory*, vol. 57, no. 1, pp. 345–364, 2011.
- [21] L. M. Correia and P. F. M. Smulders, "Characterisation of propagation in 60 GHz radio channels," *Electronics & Communication Engineering Journal*, vol. 9, no. 2, pp. 73–80, 1997.
- [22] D. Fan, F. Gao, Y. Liu, Y. Deng, and A. Nallanathan, "Angle domain channel estimation in hybrid mmWave massive MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 12, pp. 8165–8179, 2018.
- [23] J. Zhao, F. Gao, W. Jia, S. Zhang, S. Jin, and H. Lin, "Angle domain hybrid precoding and channel tracking for mmWave massive MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 16, no. 10, pp. 6868–6880, 2017.
- [24] R. Bansal, "Antenna theory; analysis and design," *Proceedings of the IEEE*, vol. 72, no. 7, pp. 989–990, 1984.
- [25] J. Li, S. Cheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Approximate holistic aggregation in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 13, no. 2, pp. 1–24, 2017.
- [26] 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.
- [27] Z. Cai, S. Ji, J. He, A. G. Bourgeois, and L. Wei, "Distributed and asynchronous data collection in cognitive radio networks with fairness consideration," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2020–2029, 2014.
- [28] Z. Cai, Y. Duan, and A. G. Bourgeois, "Delay efficient opportunistic routing in asynchronous multi-channel cognitive radio networks," *Journal of Combinatorial Optimization*, vol. 29, no. 4, pp. 815–835, 2015.
- [29] Z. Cai, S. Ji, J. He, and A. G. Bourgeois, "Optimal distributed data collection for asynchronous cognitive radio networks," in *2012 IEEE 32nd International Conference on Distributed Computing Systems*, pp. 245–254, Macau, China, 2012.
- [30] Q. Chen, Z. Cai, L. Cheng, and H. Gao, "Low-latency data aggregation scheduling for cognitive radio networks with non-predetermined structure," *IEEE Transactions on Mobile Computing*, 2020.
- [31] F. Rusek, D. Persson, Buon Kiong Lau, E. G. Larsson, T. L. Marzetta, and F. Tufvesson, "Scaling up MIMO: opportunities and challenges with very large arrays," *Signal Processing Magazine*, vol. 30, no. 1, pp. 40–60, 2013.

## Research Article

# Dual-Level Attention Based on a Heterogeneous Graph Convolution Network for Aspect-Based Sentiment Classification

Peng Yuan , Lei Jiang , Jianxun Liu , Dong Zhou , Pei Li , and Yang Gao 

School of Computer Science and Engineering, Hunan University of Science and Technology, 411201, China

Correspondence should be addressed to Lei Jiang; [jleihn@hotmail.com](mailto:jleihn@hotmail.com)

Received 28 December 2020; Accepted 26 April 2021; Published 27 May 2021

Academic Editor: Yan Huang

Copyright © 2021 Peng Yuan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the development of 5G, the advancement of basic infrastructure has led to considerable development in related research and technology. It also promotes the development of various smart devices and social platforms. More and more people are now using smart devices to post their reviews right after something happens. In order to keep pace with this trend, we propose a method to analyze users' sentiment by using their text data. When analyzing users' text data, it is noted that a user's review may contain many aspects. Traditional text classification methods used by smart devices, however, usually ignore the importance of multiple aspects of a review. Additionally, most algorithms usually ignore the network structure information between the words in a sentence and the sentence itself. To address these issues, we propose a novel dual-level attention-based heterogeneous graph convolutional network for aspect-based sentiment classification which minds more context information through information propagation along with graphs. Particularly, we first propose a flexible HIN (heterogeneous information network) framework to model the user-generated reviews. This framework can integrate various types of additional information and capture their relationships to alleviate semantic sparsity of some labeled data. This framework can also leverage the full advantage of the hidden network structure information through information propagation along with graphs. Then, we propose a dual-level attention-based heterogeneous graph convolutional network (DAHGCN), which includes node-level and type-level attentions. The attention mechanisms can analyze the importance of different adjacent nodes and the importance of different types of nodes for the current node. The experimental results on three real-world datasets demonstrated the effectiveness and reliability of our model.

## 1. Introduction

With the introduction of 5G, the combination of 5G and artificial intelligence promotes information integration and improves people's smart device experience [1]. For example, China Mobile proposed the "5G+ACIDE" plan. This also makes it more convenient for people to use smart devices to express their reviews. For instance, on Yelp and Meituan [2, 3], the number of user-generated reviews has increased dramatically. When analyzing users' text data, it is noted that a user's review may contain many aspects. Traditional text classification methods used by smart devices usually ignore the importance of multiple aspects of the information. The addition of aspect-based information can increase the accuracy of text data analysis. For example, by collecting Yelp restaurant review data to analyze what the user's attitudes

implied in it are, we can infer whether the restaurant is worth going to and which aspect of the restaurant can be improved. In recent years, people have proposed an aspect-based sentiment analysis task, which is a subtask of sentiment analysis [4, 5]. It is aimed at identifying the sentiment polarity towards a given aspect and providing more detailed feedback information than traditional sentiment analysis jobs [6–9]. An example of a user-generated review is shown in Figure 1. "Great food, but the service is dreadful." We can find that "great" is a specific sentiment word towards foods and "service" is the sentiment word corresponding to the given aspect service. It is helpful for the restaurant to attract customers by improving their dining experience through sentiment analysis by extracting their evaluation of all aspects of the restaurant. With the rapid development of the Internet, the Internet of Things, and cloud computing, data in each

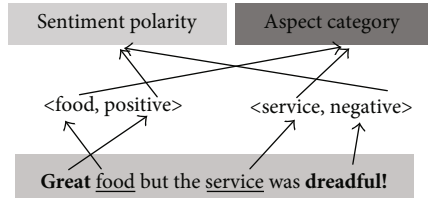


FIGURE 1: An example of aspect-level sentiment classification. The underlined words are aspect terms. The bold-faced words are aspect-related sentiment terms and sentiment words.

field has increased tremendously, e.g., science, technology, software industry, and business. The Internet of things here means machines embedded with iBeacon or sensors that collect and store data for analysis [10–13]. The wave of big data generated by the Internet of things will drive the growing demand for data analysis. Big Data Sentiment Analysis (BDSA) has developed into a very hot topic, attracting wide attention from all investors and stakeholders from the global academic community, industry, and government. All of these also apply to IoT datasets [9, 14–17]. The Internet of things (IoT) enables massive terminal connections and generates massive amounts of data through high-speed Internet and sensor technologies [18]. As shown in Figure 2, we can see the relationship between IoT/sensor network and sentiment analysis. We have also found that with the development of the blockchain which is a pioneer cryptocurrency [15, 19–22], media and public opinions show increased usage of text analysis [23, 24]. Blockchain is considered to be an innovative technology with epoch-making significance in recent years. Blockchain is a peer-to-peer distributed ledger based on cryptography and network sharing system, which is decentralized, transparent, and open [25]. The research and application of sentiment analysis in these different emerging research fields show that the field of sentiment analysis is also expanding and has a positive impact on practical applications.

The overall flowchart of sentiment analysis is shown in Figure 3. We can know that sentiment classification is performed by sentiment identification and feature selection of the user comments. Previously, most existing methods leverage the neural networks to extract contextual information and the representations of aspect categories, such as long-short-term memory network (LSTM) [26, 27] and recurrent neural network (RNN) [28, 29], which usually lead to mismatching of the sentiment polarity with the target aspect. Recently, with widespread application of attention mechanisms in NLP, we noticed that lots of ASC (aspect-level sentiment classification) models combine the attention mechanisms with RNN and achieve great performance [18, 21, 30–34]. Chen et al. [35] applied multiattention mechanisms to capture the correlation between sentiment word and the target aspect separated by a long distance. Tang et al. [36] incorporated the attention mechanisms to focus on different parts of a sentence when different aspects are taken as input. These models use attention mechanisms to capture semantic information between aspect features and context. The method of combining attention models can fur-

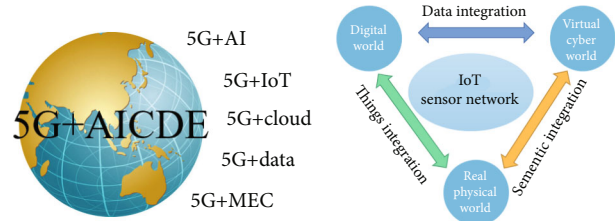


FIGURE 2: The relationship between 5G and sentiment analysis.

ther improve the performance of the model. However, it ignores the network structure. In addition, it may end up with suboptimal performance if inherent noises are introduced in the attention mechanisms.

Recently, a hierarchical attention network proposed by Cheng et al. [37] and Gao et al. [38] introduced the location relation and uses the aspect features to extract the sentiment features. However, there is no explanation for the relationship between syntactic constraints and long-distance dependency. Due to few labeled data and a large amount of unlabeled data, existing aspect-based sentiment classification methods still have the problem of semantic sparsity in modeling. In addition, the previous studies did not consider how to add additional information to mine and enrich semantic information.

To address the above issues, we propose a novel dual-level attention based on a heterogeneous graph convolutional network for aspect-based sentiment analysis. Firstly, we propose a flexible HIN (heterogeneous information network) framework to model the review to alleviate semantic sparsity and enrich semantic information. It can integrate several amounts of additional information and learn the hidden network structure information. In this paper, we introduce two types of additional information (e.g., aspect term and sentiment term) which greatly enriches the semantic information. Then, we propose a dual-level attention based on a heterogeneous graph convolutional network for aspect-based sentiment analysis. Due to the heterogeneity between the different types of information, HIN cannot be directly embedded into GCN. So we adopt GCN’s improved model HGCN and embed HIN into HGCN. And we apply a dual-level attention to learn the importance of different adjacent nodes and the importance of different types of nodes to the current node. In this way, by adjusting the weights of different types of nodes to learn the relationship between aspect items and sentiment items, it can alleviate the long-distance dependence between aspect items and sentiment items. By increasing the weights of different types nodes and learning the relationship between aspect item and sentiment item, it can alleviate long-distance dependency between aspect item and sentiment item. Experimental results on SemEval competition datasets demonstrate the effectiveness and reliability of our approach.

The remainder of this article is organized as follows. Section 2 introduces related works. Section 3 introduces the framework and details of the DAHGCN model. Section 4 presents the experimental evaluation results and discussion. Finally, the main conclusions are summarized in Section 5.



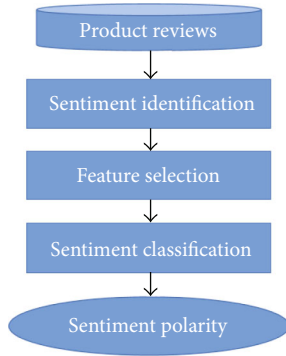


FIGURE 3: Sentiment analysis process on product reviews.

## 2. Related Work

Aspect-based sentiment analysis is aimed at identifying the specific sentiment word towards the target aspect, which is a fine-grained task in the ABSA task. We will introduce the development of aspect-based sentiment analysis in the following three periods.

Early studies mainly focus on training a classifier including SVM (support vector machine), and some improved models of SVM. They feed the feature vector of text into the classifier and then obtain the classification results. For example, Wagner et al. [39] introduced the relationship between a sentiment word and target aspect to assist in training an improved SVM classifier.

Later, the recurrent neural network (RNN) abstracts more attention with widespread application in NLP [40–42]. Lots of studies introduce RNN into aspect-based sentiment analysis and achieve better performance [27, 36, 43–45]. For instance, Tang et al. [36] used two LSTMs to model the bidirectional semantic context of a given aspect. Ruder et al. [35] implemented a hierarchical bidirectional LSTM model to learn the sentences’ contextual information. These RNN-based models achieve better classification results because RNN has many advantages; i.e., the LSTM is better at extracting short-range dependencies among words in sentences [29, 46, 47]. However, these RNN-based methods cannot extract potential correlations between sentiment words and aspect words that are relatively far away in complex sentences.

Recently, lots of studies indicate that the introduction of attention mechanisms can alleviate the aforementioned problem [24, 30–34]. A complex sentence may contain several aspects, each word in a sentence may be associated with one or more aspect terms, and a phrase in a sentence may convey sentiment information about a particular aspect term. By introducing the attention mechanism, we can capture the detailed sentiment features towards the specific aspect in the complex sentences. In particular, Wang et al. [31] proposed the ATAE-LSTM model which combines LSTM and attention mechanisms. It takes the aspect of embedding to calculate the attention weights. Ma et al. [44] proposed a model with bidirectional attention mechanisms for effective learning context and attention weights of aspect terms, respectively. Obviously, these models further improve the

accuracy of sentiment analysis. But, while introducing attention mechanisms, they also introduce its inherent noise. In particular, they ignore the network structure information between the words in the comment text and the comment text itself.

Moreover, all these models lack an explanation of syntactic constraints and long-distance dependence. Due to the short review text and the few labeled data, most existing methods have the problem of semantic sparsity while modeling text. And these studies do not consider integrating additional information for enriching semantic information. Thus, we propose a novel dual-level attention based on a heterogeneous graph convolutional network for aspect-based sentiment analysis.

## 3. DAHGCN Model

In this paper, we propose a novel dual-level attention based on a heterogeneous graph convolutional network for aspect-based sentiment analysis which leverages the full advantage of the few labeled data by allowing the information propagation along with the graph. Our method involves two steps. First, in order to address the semantic sparsity and mind the hidden network structure information, we propose a flexible HIN (heterogeneous information network) [13, 48, 49] framework. It also can integrate several amounts of additional information to greatly enrich the semantic information. Then, we propose the DAHGCN model which uses both type-level and node-level attentions to learn the relationship between the aspect term and sentiment term. It can not only improve the ability to learn the importance of different nodes by reducing the weights of noisy information but also learn the importance of different node types.

*3.1. HIN for a User-Generated Review.* We first present the HIN framework for modeling the review texts which alleviates the semantic sparsity by integrating some additional information and greatly enriches the semantic information among the review text and additional information. It is also helpful for smart devices to conduct the analysis of users’ text data by reviewing multiple aspects of the information.

Previous studies usually adopt the topic of the text and external knowledge to enrich semantic information. However, it does not apply to the ABSA task. In this paper, we introduce the relationship between aspect features and sentiment features to enrich the semantic information. Thus, we consider two types of additional information, i.e., aspect term and sentiment term. As shown in Figure 4, the HIN is constructed as  $G = (V, \epsilon)$ , which contains the review texts  $T = \{t_1, t_2, \dots, t_n\}$ , aspect term  $A = \{\alpha_1, \alpha_2, \dots, \alpha_k\}$ , and sentiment term  $S = \{s_1, s_2, \dots, s_n\}$  as nodes, where  $V = T \cup A \cup S$  is a set of edges representing the structural relationships between two nodes. We will show the details of HIN as follows.

The construction of the HIN mainly includes two steps. First, we recognize the aspect term in the review text and map them to Wikipedia with the linking tool TAGME. And we take the aspect term as a whole word and learn the embeddings of the aspect term. In order to enrich the

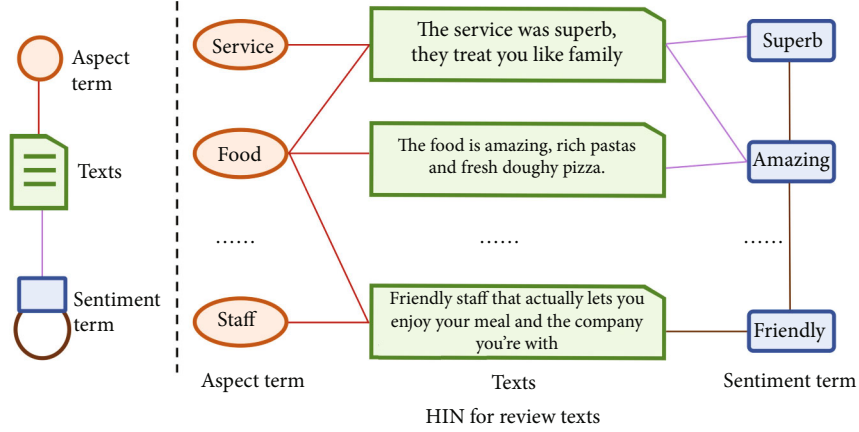


FIGURE 4: An example of HIN for short texts on Rest14.

semantic information, we introduce the similarity among the aspect terms; if the score that is computed by their embeddings between two aspect terms is above a predefined value  $\delta_a$ , we build a link for them. Second, in the same way, we recognize the sentiment term in the review text and map them to Wikipedia. And we also take the sentiment term as a whole word, learn the embeddings of the sentiment term, define a value  $\delta_b$ , and compute the similarity between two sentiment term embeddings. If the score is above  $\delta_b$ , we link them through an edge.

We can learn the relationship among the aspect terms, texts, and sentiment terms, relationships between sentiment terms, and relationship between aspect terms by integrating both additional information of the aspect term and sentiment term, as shown in Figure 4. It greatly enriches the information of the review texts and improves the performance of the ABSA task.

**3.2. DAHGCN.** As shown in Figure 5, we propose a novel dual-level attention based on a heterogeneous graph convolutional network for aspect-based sentiment analysis. It contains node-level attention and type-level attention. We first embed the HIN into HGCN. Then, we introduce the dual-level attention mechanisms to compute the attention weights of different adjacent nodes and different types of nodes. Finally, we use a softmax layer to give prediction results of sentiment classification.

**3.3. HGCN (Heterogeneous Graph Convolution Network).** In this paper, the HIN framework integrates two kinds of additional information. Due to the heterogeneity among the different types of nodes, the HIN framework cannot directly apply to traditional GCN (graph convolutional network) [50]. To address this issue, we introduce HGCN (heterogeneous graph convolutional network) which is an improved model of GCN.

Hu et al. [51] first propose a multilayer neural network which can operate in a homogeneous graph. Usually, for a graph  $G = (V, \epsilon)$ , where  $V$  and  $\epsilon$  represent the set of nodes and edges, respectively, it makes  $X \in R^{|V| \times d}$  represent the matrix of all nodes with their nodes, where  $|V|$  is the number of nodes and  $d$  is the dimension of the feature vectors.

Each row of the matrix represents a feature vector  $x_v$  of a node. In a graph, due to the self-connection of each node, we set the adjacent matrix  $\tilde{A} = A + I$ , where  $M_i = \sum_j A_{ij}$ . The layer-wise propagation of GCN is defined as follows:

$$H^{(l+1)} = \sigma(\tilde{A} \cdot H^{(l)} \cdot W^{(l)}), \quad (1)$$

where  $\sigma$  is an active function,  $\tilde{A}$  is the normalized adjacency matrix of  $A$ , and  $H^{(l)} \in R^{|V| \times d}$  is the hidden state of all nodes in the  $l^{\text{th}}$  layer. Initially,  $H^{(0)} = X$  and  $W^{(l)}$  are layer-specific trainable transformation matrices. HGCN considers the heterogeneity of all types of nodes and projects them into an implicit common space with their respective transformation matrices. The layer-wise propagation of HGCN is defined as follows:

$$H^{(l+1)} = \sigma\left(\sum_{\tau} \tilde{A}_{\tau} \cdot H_{\tau}^{(l)} \cdot W_{\tau}^{(l)}\right), \quad (2)$$

where  $\tilde{A}_{\tau} \in R^{|V| \times d}$  is the submatrix of  $\tilde{A}$  and rows and columns represent all nodes and their neighboring nodes with type  $\tau$ , respectively. The node representations  $H^{(l+1)}$  are obtained by aggregating information from the features of their neighboring nodes  $H_{\tau}^{(l)}$  with different types  $\tau$  using different transformation matrices  $W_{\tau}^{(l)} \in R^{d^{(l)} \times d^{(l+1)}}$ . The transformation matrix  $W_{\tau}^{(l)}$  considers the difference of different feature spaces and projects them into an implicit common space  $R^{d^{(l+1)}}$ . Initially,  $H_{\tau}^{(0)} = X$ .

**3.4. Dual-Level Attention Mechanism.** Particularly, given a target node, it will bring different impacts on the target node by different adjacent nodes with different types. For instance, adjacent nodes with different types may supply more valuable information. Moreover, different adjacent nodes with the same type may have different information. In order to characterize relationships between terms at the node level and type level, we propose novel dual-level attention mechanisms.

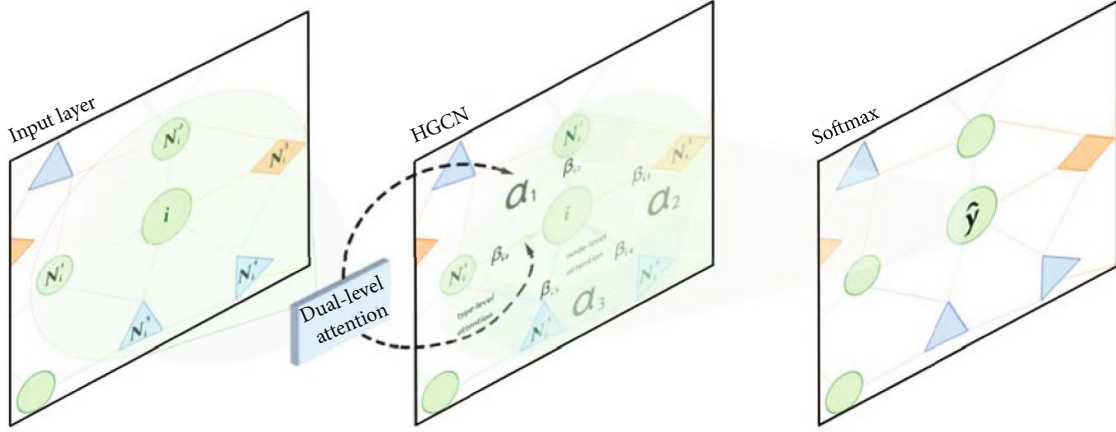


FIGURE 5: Illustration of our model DAHGCN.

**3.4.1. Type-Level Attention.** For a target node  $v$ , type-level attention can learn the weights of adjacent nodes of different types. Particularly, we first use the embedding vector  $h_\tau = \sum_{v'} \widetilde{A}_{vv'} h_{vv'}$  to represent the type  $\tau$ ; it represents the sum of adjacent node features with type  $\tau$ . Then, we leverage the target node embedding vector  $h_v$  and type embedding vector  $h_\tau$  to calculate the type-level attention score.

$$\alpha_\tau = \sigma(\mu^T \cdot [h_v | h_\tau]), \quad (3)$$

where  $\mu^T$  is the attention vector of type  $\tau$  and  $|$  is the “concatenate.” After normalizing the attention weights of all types, the attention weights are defined as

$$\alpha_\tau = \frac{\exp(\alpha_\tau)}{\sum_{\tau' \in \tau} \exp(\alpha_{\tau'})}. \quad (4)$$

**3.4.2. Node-Level Attention.** We introduce node-level attention to learn the importance of different adjacent nodes and reduce the weights of the noise information. Particularly, given a target node  $v$  and its neighboring nodes  $v'$ , we use the embedding vector  $h_v$  of node  $v$ , the embedding vector  $h_{v'}$  of node  $v'$ , and the type-level attention score of a node as the input to calculate the node-level attention score. It is defined as

$$b_{vv'} = \sigma(v^T \cdot \alpha_\tau \cdot [h_v | h_{v'}]), \quad (5)$$

where  $v^T$  is the attention vector of node  $v$ . After normalizing the attention weights of all neighboring nodes, the attention weights are defined as

$$\beta_{vv'} = \frac{\exp(b_{vv'})}{\sum_{i \in N_v} \exp(b_{vi})}. \quad (6)$$

TABLE 1: Datasets with the number of sentences (S) and &lt;aspect, sentiment&gt; tuples (T).

Dataset	Train		Test		Total	
	#S	#T	#S	#T	#S	#T
Rest14	3401	3713	800	1025	4201	4738
Rest15	1315	1476	685	730	2000	2206
Rest16	2000	2107	676	703	2676	2810

Finally, we add the dual-level attention into the HGCN.

$$H^{(l+1)} = \sigma \left( \sum_{\tau \in \Gamma} B_\tau \cdot H_\tau^{(l)} \cdot W_\tau^{(l)} \right), \quad (7)$$

where  $B_\tau$  is the attention weight matrix of all nodes; we use  $\beta_{vv'}$  to represent the element in the  $v^{\text{th}}$  row  $v'$  column of the matrix.

### 3.5. Prediction of Aspect-Based Sentiment Analysis and Loss Function

**3.5.1. Aspect-Based Sentiment Classification Prediction.** Based on DAHGCN, we can obtain the text embeddings  $H^{(L)}$  of review text in HIN. Then, we train the embedding into softmax for classification.

$$Z = \text{softmax}(H^{(L)}). \quad (8)$$

We use the  $L_2$ -norm as the loss function in our model training, and it is defined as

$$L = - \sum_{i \in D_{\text{train}}} \sum_{j=1}^C Y_{ij} \log Z_{ij} + \eta \|\Theta\|_2, \quad (9)$$

where  $C$  is the number of sentiment polarity categories,  $D_{\text{train}}$  is the set of review numbers for training,  $\eta$  is the  $L_2$ -regularization term, and  $\Theta$  is the parameter set.

TABLE 2: The entity and attribute of datasets.

Entity labels
Restaurant, food, drinks, ambience, service, location
Attribute labels
General, prices, quality, style_options, miscellaneous

TABLE 3: The output of classification.

	1	0
1	True positive	False negative
0	False positive	True negative

TABLE 4: Accuracy on aspect-level sentiment classification. Bin. indicates binary prediction. Thr. stands for 3-class prediction. Best scores are in bold.

Model	Rest14		Rest15		Rest16	
	Bin.	Thr.	Bin.	Thr.	Bin.	Thr.
AT-LSTM	89.6	83.1	81.0	77.2	87.6	83.0
ATAE-LSTM	89.9	84.0	80.9	77.4	87.2	82.7
HEAT-BiGRU	90.1	85.3	82.6	80.2	89.8	86.7
ASGCN	89.3	85.2	83.3	80.4	89.3	86.3
HGCN	89.6	85.5	83.7	80.6	89.8	86.4
TD-GAT	89.5	85.4	83.6	80.7	89.7	86.7
DHGCN	<b>90.3</b>	<b>86.2</b>	<b>84.2</b>	<b>80.9</b>	<b>90.5</b>	<b>87.3</b>

TABLE 5: Recall on aspect-level sentiment classification. Bin. indicates binary prediction. Thr. stands for 3-class prediction. Best scores are in bold.

Model	Rest14		Rest15		Rest16	
	Bin.	Thr.	Bin.	Thr.	Bin.	Thr.
AT-LSTM	81.5	70.8	74.6	53.4	78.8	58.6
ATAE-LSTM	81.3	71.3	74.8	53.8	78.5	57.6
HEAT-BiGRU	81.8	71.7	75.5	54.3	79.7	59.3
ASGCN	81.2	71.6	75.2	54.5	79.5	59.1
HGCN	81.5	72.3	75.4	54.7	79.8	59.4
TD-GAT	82.1	72.6	76.2	54.9	79.7	59.6
DHGCN	<b>82.5</b>	<b>73.2</b>	<b>76.6</b>	<b>55.3</b>	<b>80.4</b>	<b>60.2</b>

## 4. Experiments

**4.1. Datasets and Experiment Settings.** We use the SemEval competition dataset to evaluate the DAHGCN model at aspect-level sentiment classification. It includes task 4 of the SemEval 2014 competition, task 12 of SemEval 2015 competition, and task 5 of the SemEval 2015 competition. The dataset contains user-generated reviews of restaurants. Each dataset contains a target aspect, an aspect term, and the aspect-specific sentiment polarity. They are labeled as {positive, negative, neutral}. The datasets are detailed in Table 1, where Rest represents restaurant reviews and 14, 15, and 16 [52–54] indicate the year the datasets came from.

TABLE 6: Macro-F1 on aspect-level sentiment classification. Bin. indicates binary prediction. Thr. stands for 3-class prediction. Best scores are in bold.

Model	Rest14		Rest15		Rest16	
	Bin.	Thr.	Bin.	Thr.	Bin.	Thr.
AT-LSTM	83.7	74.4	78.7	58.7	82.4	60.7
ATAE-LSTM	83.6	75.1	78.9	59.4	82.1	60.2
HEAT-BiGRU	84.1	75.4	80.2	60.2	83.8	61.2
ASGCN	83.8	74.3	79.7	59.8	83.5	61.3
HGCN	84.0	74.8	81.2	60.4	84.2	61.7
TD-GAT	84.3	75.2	81.6	60.6	84.4	62.5
DHGCN	<b>84.7</b>	<b>75.6</b>	<b>82.3</b>	<b>60.7</b>	<b>84.9</b>	<b>63.1</b>

And Table 2 shows the entity labels and attribute labels of datasets. The baseline dataset of each classification is divided into a training set (70%) and a test set (30%) by using the random segmentation tool in Sklearn.

We selected accuracy, recall, and macro-F1 to compare and evaluate each method [55]. They are defined with reference to Table 3, and their formulas are as follows:

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}}, \quad (10)$$

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}}, \quad (11)$$

$$\text{F1\_score} = 2 \times \frac{\text{recall} \times \text{precision}}{\text{recall} + \text{precision}}, \quad (12)$$

$$\text{macro\_F1} = \frac{\sum_i^n \text{F1\_score}_i}{n}. \quad (13)$$

### 4.2. Model for Comparison

- (1) AT-LSTM: it adopts the attention mechanism in LSTM to generate a weighted representation of a sentence [31].
- (2) ATAE-LSTM: this method is an extension of AT-LSTM. This model appends input aspect embedding into each word input vector to model the interdependence between words and the input aspect [31].
- (3) HEAT-BiGRU: the model captures the aspect information of a text and uses the aspect information to capture the aspect-specific sentiment information [37].
- (4) ASGCN: it builds a graph convolutional network (GCN) on the sentence’s dependency tree. It uses syntactic information and word dependencies to resolve errors and identify grammatically unrelated context words as clues for judgment emotion [56].
- (5) TD-GAT: it applies a graph to make explicit use of the dependency relationships between words to directly spread emotional features from the syntactic

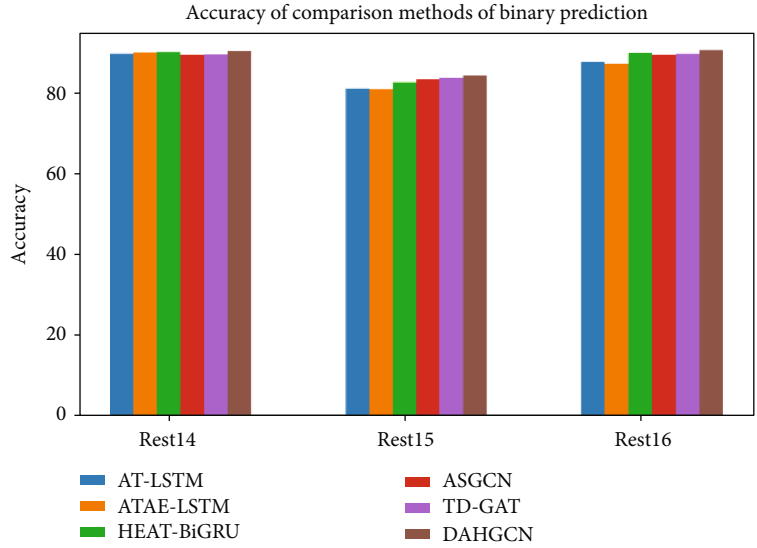


FIGURE 6: Accuracy of comparison methods of binary prediction.

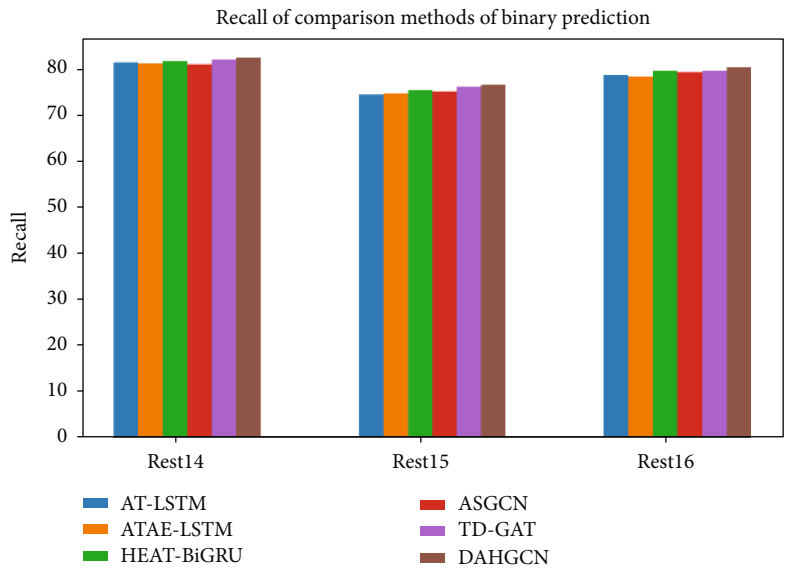


FIGURE 7: Recall of comparison methods of binary prediction.

context of aspect targets and to classify aspect-level emotion [57].

- (6) DAHGCN: an aspect-level sentiment classification model of a heterogeneous graph convolutional network (HGCN) based on a two-level attention mechanism, which includes node-level attention and type-level attention, embeds HIN into aspect-level sentiment classification

4.3. *Evaluation and Analysis.* As shown in Tables 4, 5, and 6, the accuracy, recall, and macro-F1 of our method DAHGCN are better than those of other models. It is also easy for us to draw this conclusion from Figures 6, 7, 8, 9, 10, and 11. We argue that it benefited from HIN, which integrates two types

of additional information: aspect and emotion. The additional information greatly enriches the semantic information of the nodes, which effectively reduces the semantic sparseness. In addition, the node-level and type-level attention mechanisms we adopted enable the model to fully learn the importance of different neighbor nodes and different types of nodes to the current node. At the same time, these attention mechanisms also alleviate noise. These may be the reason why DAHGCN’s results are the best.

Firstly, we analyze these algorithms from the attention mechanism. AT-LSTM, ATAE-LSTM, and HEAT-BiGRU all used attention mechanisms to learn the relationship between aspects and emotions, but they do not consider the relationship between sentences. DAHGCN adds node-level attention, and it can effectively learn the importance of

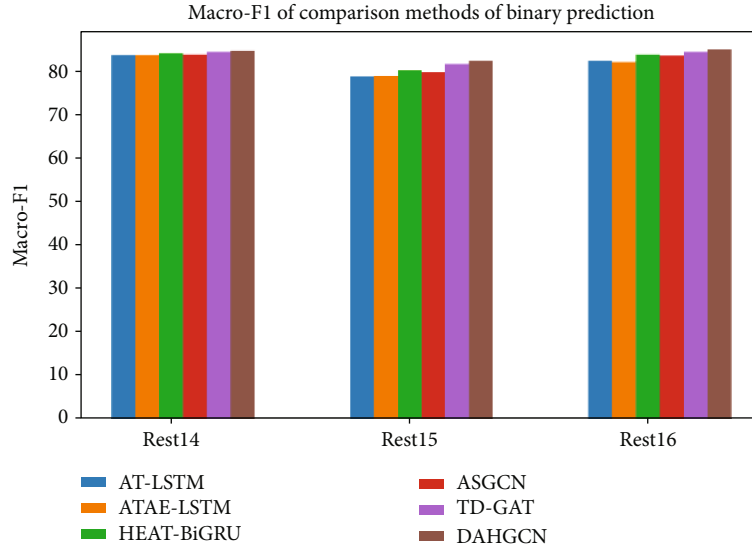


FIGURE 8: Macro-F1 of comparison methods of binary prediction.

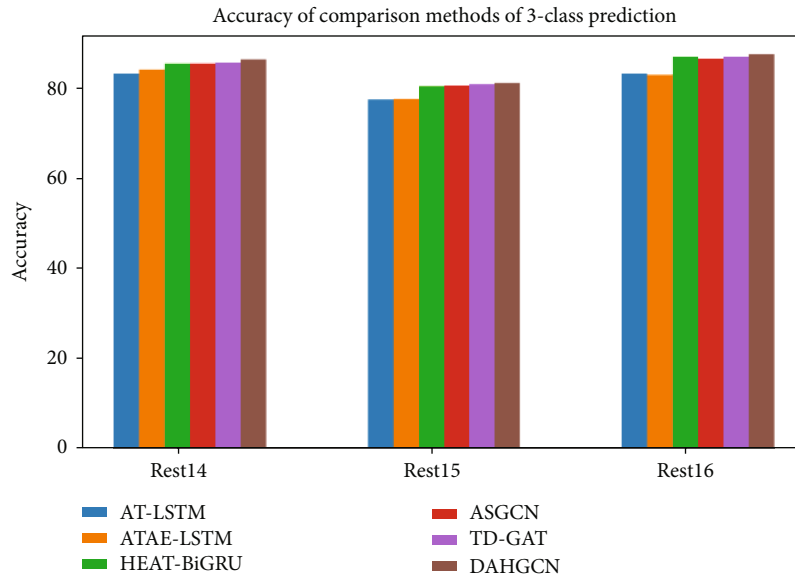


FIGURE 9: Accuracy of comparison methods of 3-class prediction.

adjacent nodes to the current node so it is able to capture the dependency relationships between sentences. Therefore, DAHGCN can obtain better classification results. It is shown in Table 4 that the accuracy of DAHGCN is better than that of the above algorithm, and the results in Tables 5 and 6 also support this conclusion. We can see that accuracy results of DAHGCN are 0.7%, 0.4%, and 0.2% higher than those of AT-LSTM, ATAE-LSTM, and HEAT-BiGRU, respectively. Recall results of DAHGCN are 1.0%, 1.2%, and 0.7% higher than those of AT-LSTM, ATAE-LSTM, and HEAT-BiGRU, respectively. Macro-F1 results of DAHGCN are 1.0%, 1.1%, and 0.6% higher than those of AT-LSTM, ATAE-LSTM, and HEAT-BiGRU, respectively. Secondly, we discuss DHGCN from the perspective of HIN. It can be seen from

the classification results of each dataset in Table 4 that the accuracy of HGCN is generally better than that of ASGCN, AT-LSTM, ATAE-LSTM, and HEAT-BiGRU. It is also shown in recall and macro-F1 (Tables 5 and 6) that HGCN results are better than those of ASGCN, AT-LSTM, ATAE-LSTM, and HEAT-BiGRU. Specifically, as seen intuitively from Figure 12, HEAT-BiGRU performs slightly better than HGCN in binary prediction and HGCN is better than AT-LSTM and ATAE-LSTM, but we can see that HGCN performs better in 3-class prediction. Adding heterogeneous information and supplementing classification judgment information enable HGCN to perform more stably. Compared with other algorithms, we argue that the main reason is that HGCN uses HIN to further mine the semantic

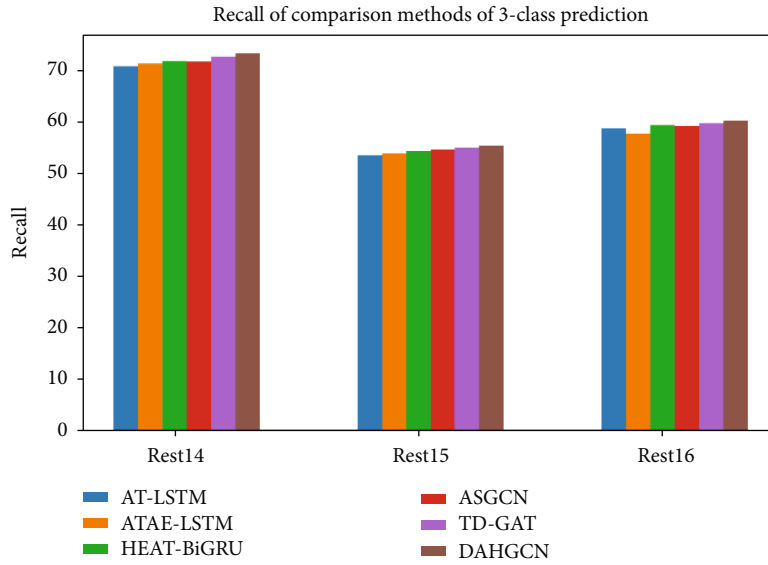


FIGURE 10: Recall of comparison methods of 3-class prediction.

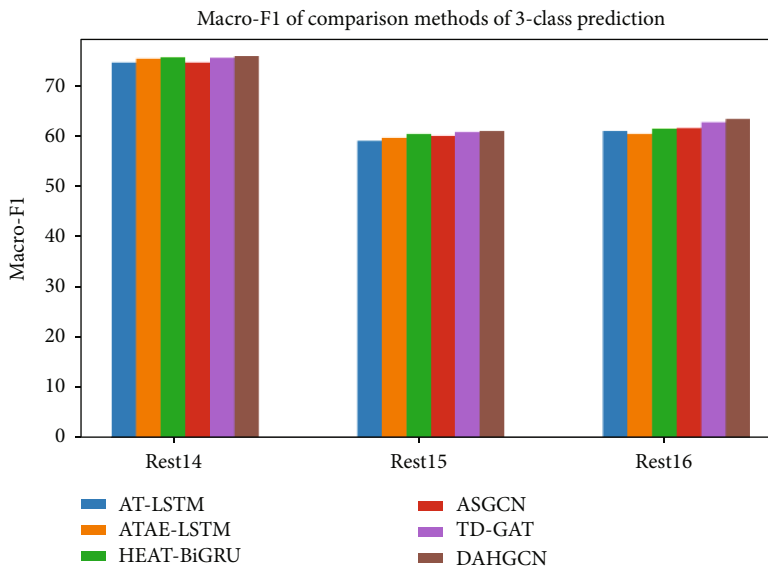


FIGURE 11: Macro-F1 of comparison methods of 3-class prediction.

information of the text. In this way, its node feature representation is more accurate, so its results are better. In summary, DAHGCN adopts the HIN mechanism and improves the attention mechanism, so its result is the best among all algorithms.

From the perspective of the attention mechanism, AT-LSTM, ATAELSTM, and HEAT-BiGRU all use attention mechanisms to learn the relationships between aspects and sentiments, but they do not consider the relationship between sentences. However, DAHGCN uses node-level attention to effectively learn the importance of neighboring nodes to the current node, so as to learn the relationship between sentences, so as to get better classification results, as shown in Table 4, from Rest14 data. The results of the sec-

ond classification of the set can be seen that the accuracy of DAHGCN is better than that of ASGCN (1.0%), AT-LSTM (0.6%), ATAELSTM (0.4%), HEAT-BiGRU (0.2%), and TD-GAT (0.8%). The results in Tables 5 and 6 also show the excellence of DAHGCN. As we know, the attention mechanism can extract information about keywords in a sentence. Extracting attention weights for classification can improve the classification effect. Experiments have proven that using the attention mechanism can achieve good results. However, as the amount of information contained in the data and the complexity increase, more information needs to be considered. In this paper, by establishing a graph network, supplementing information, and filtering information by designing a dual-level attention mechanism, a better

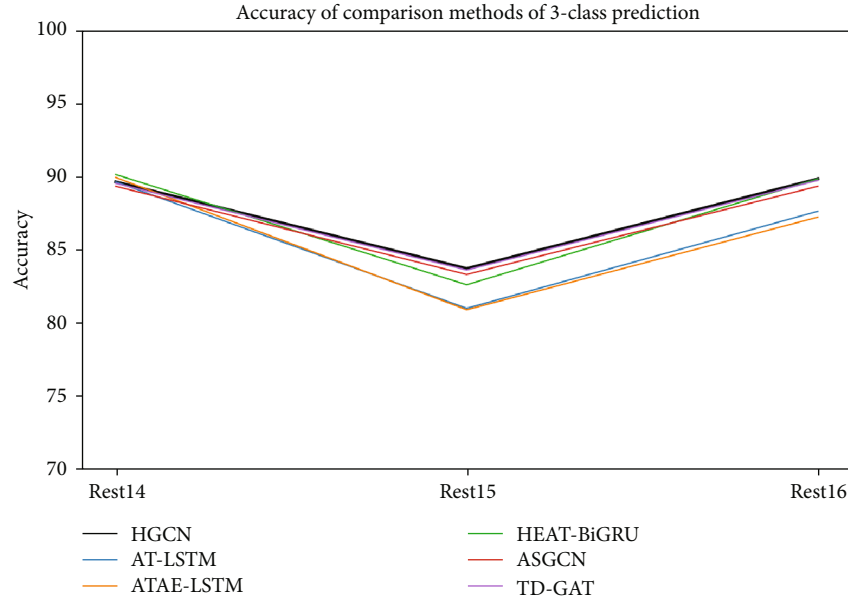


FIGURE 12: HGCN and other comparison methods.

experimental effect was obtained. For some complex situations, the attention mechanism will also have difficulties. This difficulty will be mentioned below.

In order to have an intuitive understanding on how DAHGCN works, we present a case study to show it. For instance, the sentence “Great food but the service was dreadful” contains two aspects which may result in sentiment mismatch by using the model combining attention and recurrent neural network. The other sentence “If the service is nice, I will go back again” uses a word “if,” bringing extra difficulty in detecting implicit semantics.

For the first sentence, our model can easily supervise the sentiment words and enable the model to allocate more attentions to the aspect-specific sentiment words. In particular, “Great” is the sentiment term towards food, and “dreadful” is the sentiment term corresponding to service. But our model has difficulty learning the logical information in a sentence. For the second sentence “if the service is nice, I will go back again,” it expresses negative feelings towards the service, but there are no obvious negative words; it is difficult to make a correct prediction. Our approach can easily extract the sentiment information “nice” but cannot learn the word “if.” In this case, the attention mechanism is difficult to learn. In the process of learning, it is difficult for attention to associate sentiment words with the label; i.e., it is difficult to learn attention weights. This is also the direction we need to study in the future.

As can be seen from Figures 6, 7, and 8, the comparison method performs worse on the dataset Rest15 than on Rest14 and Rest16. This is because the data of Rest15 is relatively large, and you can see from Table 1 that the sentiment information of the Rest15 dataset is less. So the result of the comparison method is relatively poor on this dataset. The same results can be seen from Figures 9, 10, and 11. Of course, there are differences in the performance of the comparison

method between binary prediction and 3-class prediction. 3-class prediction requires more integrated information for judgment, and at the same time, sentences like “if the service is nice, I will go back again” will affect the classification judgment more.

From the results, it is evident that the proposed model has a comparable performance that is superior to that of the other models. By combining with the previous experiments, it can be proven that the proposed model can achieve relatively high classification performance.

## 5. Conclusion and Future Work

In this paper, we propose a novel dual-level attention model based on a heterogeneous graph convolutional network for aspect-based sentiment analysis. We first propose a HIN framework that integrates two kinds of additional information (aspect term and sentiment term). To mine the hidden network information and semantic information, we construct a heterogeneous graph network which can also address the semantic sparsity. By leveraging the HGCN, the embedding information of a word and review text can be learned. And we introduce a dual-level attention mechanism to learn the importance of different adjacent nodes and different types of nodes. Finally, the aspect sentiment classification is transformed into node classification. The experimental results show that the results after we add aspect information to HGCN are better than those of other methods in terms of accuracy, recall, and macro-F1 measures. It facilitates analysis of multiple aspects of user text by smart devices. In the future work, we will make use of the grammatical structure and the dependence between different sentiments in the review text to improve the effect of sentiment analysis at the aspect level.



## Data Availability

In this paper, we used the open-source dataset and cited it. Data of these datasets is mainly used for sentiment analysis. The data file of datasets uses XML format. There are no restrictions on data download. Download links are as follows: Rest14 (<https://alt.qcri.org/semeval2014/task4/index.php?id=data-and-tools>), Rest15 (<https://alt.qcri.org/semeval2015/task12/index.php?id=data-and-tools>), and Rest16 (<https://alt.qcri.org/semeval2016/task5/index.php?id=data-and-tools>).

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

We thank all researchers who participated in this paper. We thank the conference “2020 IEEE International Conference on Smart Cloud (SmartCloud)” for recommending our research to *Wireless Communications and Mobile Computing*. This work was supported in part by the Ministry of education of Humanities and Social Science Project, China (Grant No. 17YJAZH032); the Provincial and Municipal Joint Fund of Hunan Provincial Natural Science Foundation of China (2018JJ4052); the Hunan Provincial Department of Education Innovation Platform Open Fund (no. 20K050); and the Key Project of Hunan Provincial Department of Education (19A172).

## References

- [1] 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*, p. 1, 2021.
- [2] M. Luca, *Reviews, Reputation, and Revenue: The Case of Yelp.com*, no. 12-016, 2016Com (March 15, 2016). Harvard Business School NOM Unit Working Paper, 2016.
- [3] B. R. Mukhopadhyay and C. R. Chatwin, “The significance of Herzberg and Taylor for the gig economy of China,” *International Journal of Applied Behavioral Economics (IJABE)*, vol. 9, no. 4, pp. 1–17, 2020.
- [4] B. Agarwal, N. Mittal, P. Bansal, and S. Garg, “Sentiment analysis using common-sense and context information,” *Computational intelligence and neuroscience*, vol. 2015, 9 pages, 2015.
- [5] R. Yan, Z. Xia, Y. Xie, X. Wang, and Z. Song, “Research on sentiment classification algorithms on online review,” *Complexity*, vol. 2020, 6 pages, 2020.
- [6] R. K. Bakshi, N. Kaur, R. Kaur, and G. Kaur, “Opinion mining and sentiment analysis,” in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIA-Com)*, pp. 452–455, New Delhi, India, 2016.
- [7] 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.
- [8] 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, article e5556, 2020.
- [9] X. Zhou, W. Liang, I. Kevin, 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, 2018.
- [10] K. Li, G. Lu, G. Luo, and Z. Cai, “Seed-free graph de-anonymization with adversarial learning,” in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 745–754, 2020.
- [11] Y. Liang, Z. Cai, J. Yu, Q. Han, and Y. Li, “Deep learning based inference of private information using embedded sensors in smart devices,” *IEEE Network*, vol. 32, no. 4, pp. 8–14, 2018.
- [12] X. Zheng, Z. Cai, and Y. Li, “Data linkage in smart Internet of things systems: a consideration from a privacy perspective,” *IEEE Communications Magazine*, vol. 56, no. 9, pp. 55–61, 2018.
- [13] X. Zhou, W. Liang, I. Kevin, K. Wang, and L. T. Yang, “Deep correlation mining based on hierarchical hybrid networks for heterogeneous big data recommendations,” *IEEE Transactions on Computational Social Systems*, vol. 8, no. 1, pp. 171–178, 2021.
- [14] G. Xiao, G. Tu, L. Zheng et al., “Multi-modality sentiment analysis in social Internet of things based on hierarchical attentions and CSATTCN with MBM Network,” *IEEE Internet of Things Journal*, 2021.
- [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, 2016.
- [16] 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.
- [17] C. Huang, G. Huang, W. Liu, R. Wang, and M. Xie, “A parallel joint optimized relay selection protocol for wake-up radio enabled WSNs,” *Physical Communication*, vol. 47, p. 101320, 2021.
- [18] W. Liang, S. Xie, D. Zhang, X. Li, and K. Li, “A mutual security authentication method for RFID-PUF circuit based on deep learning,” *ACM Transactions on Internet Technology*, pp. 1–20, 2020.
- [19] 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*, p. 1, 2021.
- [20] 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*, 2020.
- [21] W. Liang, D. Zhang, X. Lei, M. Tang, K.-C. Li, and A. Zomaya, “Circuit copyright blockchain: blockchain-based homomorphic encryption for IP circuit protection,” *IEEE Transactions on Emerging Topics in Computing*, 2020.
- [22] 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.
- [23] W. Liang, J. Long, K.-C. Li, J. Xu, N. Ma, and X. Lei, “A fast defogging image recognition algorithm based on bilateral hybrid filtering,” *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 17, no. 2, pp. 1–16, 2020.
- [24] M. Yang, W. Tu, J. Wang, F. Xu, and X. Chen, “Attention based LSTM for target dependent sentiment classification,”

- in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
- [25] W. Liang, L. Xiao, K. Zhang, M. Tang, D. He, and K.-C. Li, "Data fusion approach for collaborative anomaly intrusion detection in blockchain-based systems," *IEEE Internet of Things Journal*, 2021.
- [26] M. Sundermeyer, R. Schlüter, and H. Ney, "LSTM neural networks for language modeling," in *Thirteenth annual conference of the international speech communication association*, 2012.
- [27] X. Zhou, Y. Hu, W. Liang, J. Ma, and Q. Jin, "Variational LSTM enhanced anomaly detection for industrial big data," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3469–3477, 2021.
- [28] S. Kombrink, T. Mikolov, M. Karafiát, and L. Burget, "Recurrent neural network based language modeling in meeting recognition," in *Twelfth annual conference of the international speech communication association*, 2011.
- [29] 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*, p. 1, 2020.
- [30] Z. Li, Y. Wei, Y. Zhang, X. Zhang, and X. Li, "Exploiting coarse-to-fine task transfer for aspect-level sentiment classification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 4253–4260, 2019.
- [31] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 606–615, 2016.
- [32] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "Effective attention modeling for aspect-level sentiment classification," in *Proceedings of the 27th international conference on computational linguistics*, pp. 1121–1131, 2018.
- [33] J. Wang, J. Li, S. Li et al., "Aspect sentiment classification with both word-level and clause-level attention networks," *IJCAI*, vol. 2018, pp. 4439–4445, 2018.
- [34] Y. Tay, L. A. Tuan, and S. C. Hui, "Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [35] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent attention network on memory for aspect sentiment analysis," in *Proceedings of the 2017 conference on empirical methods in natural language processing*, pp. 452–461, 2017.
- [36] D. Tang, B. Qin, X. Feng, and T. Liu, "Effective LSTMs for target-dependent sentiment classification," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 3298–3307, 2016.
- [37] J. Cheng, S. Zhao, J. Zhang, I. King, X. Zhang, and H. Wang, "Aspect-level sentiment classification with heat (hierarchical attention) network," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 97–106, 2017.
- [38] Y. Gao, J. Liu, P. Li, and D. Zhou, "CE-HEAT: an aspect-level sentiment classification approach with collaborative extraction hierarchical attention network," *IEEE Access*, vol. 7, pp. 168548–168556, 2019.
- [39] J. Wagner, P. Arora, S. Cortes et al., *DCU: Aspect-Based Polarity Classification for Semeval Task 4*, Association for Computational Linguistics, 2014.
- [40] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing [review article]," *IEEE Computational Intelligence Magazine*, vol. 13, no. 3, pp. 55–75, 2018.
- [41] 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*, 2021.
- [42] 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.
- [43] M. Zhang, Y. Zhang, and D.-T. Vo, "Gated neural networks for targeted sentiment analysis," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.
- [44] Y. Ma, H. Peng, and E. Cambria, "Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [45] S. Ruder, P. Ghaffari, and J. G. Breslin, "A hierarchical model of reviews for aspect-based sentiment analysis," 2016, <https://arxiv.org/1609.02745>.
- [46] 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*, 2020.
- [47] 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.
- [48] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "PathSim: meta path-based top-k similarity search in heterogeneous information networks," *Proceedings of the VLDB Endowment*, vol. 4, no. 11, pp. 992–1003, 2011.
- [49] 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, 2019.
- [50] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *European semantic web conference*, pp. 593–607, Cham, 2018.
- [51] L. Hu, T. Yang, C. Shi, H. Ji, and X. Li, "Heterogeneous graph attention networks for semi-supervised short text classification," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4823–4832, 2019.
- [52] M. Pontiki, D. Galanis, H. Papageorgiou et al., "Semeval-2016 task 5: aspect based sentiment analysis," *10th International Workshop on Semantic Evaluation (SemEval 2016)*, 2016.
- [53] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, "Semeval-2015 task 12: aspect based sentiment analysis," *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pp. , 2015486–495, 2015.
- [54] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, "SemEval-2014 task 4: aspect based sentiment analysis," *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, 2014, pp. 27–35, Association for Computational Linguistics, Dublin, Ireland, 2014.

- [55] Y. J. Huang, R. Powers, and G. T. Montelione, "Protein NMR recall, precision, and F-measure scores (RPF scores): structure quality assessment measures based on information retrieval statistics," *Journal of the American Chemical Society*, vol. 127, no. 6, pp. 1665–1674, 2005.
- [56] C. Zhang, Q. Li, and D. Song, "Aspect-based sentiment classification with aspect-specific graph convolutional networks," 2019, <https://arxiv.org/abs/1909.03477>.
- [57] B. Huang and K. M. Carley, "Syntax-aware aspect level sentiment classification with graph attention networks," 2019, <https://arxiv.org/abs/1909.02606>.

## Research Article

# Optimize the Communication Cost of 5G Internet of Vehicles through Coherent Beamforming Technology

Lan Wu,<sup>1</sup> Juan Xu ,<sup>1</sup> Lei Shi ,<sup>1</sup> Yi Shi,<sup>2</sup> and Wenwen Zhou<sup>1</sup>

<sup>1</sup>*School of Computer Science and Information Engineering, Intelligent Interconnected Systems Laboratory of Anhui Province, Hefei University of Technology, Hefei 230009, China*

<sup>2</sup>*Department of ECE, Virginia Tech, Blacksburg, VA 24061, USA*

Correspondence should be addressed to Lei Shi; [shilei@hfut.edu.cn](mailto:shilei@hfut.edu.cn)

Received 26 November 2020; Accepted 29 April 2021; Published 17 May 2021

Academic Editor: Yan Huang

Copyright © 2021 Lan Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Edge computing, which sinks a large number of complex calculations into edge servers, can effectively meet the requirement of low latency and bandwidth efficiency and can be conducive to the development of the Internet of Vehicles (IoV). However, a large number of edge servers mean a big cost, especially for the 5G scenario in IoV, because of the small coverage of 5G base stations. Fortunately, coherent beamforming (CB) technology enables fast and long-distance transmission, which gives us a possibility to reduce the number of 5G base stations without losing the whole network performance. In this paper, we try to adopt the CB technology on the IoV 5G scenario. We suppose we can arrange roadside nodes for helping transferring tasks of vehicles to the base station based on the CB technology. We first give the mathematical model and prove that it is a NP-hard model that cannot be solved directly. Therefore, we design a heuristic algorithm for an Iterative Coherent Beamforming Node Design (ICBND) algorithm to obtain the approximate optimal solution. Simulation results show that this algorithm can greatly reduce the cost of communication network infrastructure.

## 1. Introduction

The concept of Internet of Vehicles (IoV) has been proposed and studied for many years. However, with the continuous improvement of people's pursuit of comfort and safety, Internet of Vehicles (IoV) has been paid more and more attention and research [1–3]. Especially in recent years, automatic driving technology has been of great concern and widely studied with the development of artificial intelligence technology [4]. Autonomous driving technology must rely on the full development of the vehicles' ability to perceive the surrounding environment and communicate. Edge computing is a hot research field in recent years, which sinks a large number of complex calculations into the edge server environment to reduce cloud burden and delay [5], thus making autonomous driving technology possible [6]. Edge computing has played an important role in autonomous driving [7–9], Internet of Things (IoT) [10, 11], data privacy [12, 13], and other research fields [14, 15].

The basic communication of the edge computing framework is built on the 5G network. The 5G network has the characteristics of high data volume and low latency, which is also a key factor to ensure the widespread application of autonomous driving technology in the future [16]. In addition to bringing more extreme experience and larger capacity, 5G will also open the era of the Internet of Things (IoT) and penetrate into various industries [17, 18]. Moreover, 5G is being applied in more and more fields, such as Internet of Things, smart city, traffic driving, and surgery [19]. However, compared with current commercial 4G networks, 5G networks also have disadvantages. The coverage of 5G base stations (BS) is small and the cost is high [20, 21]. Due to the wide range of an automobile's work scenarios and strict requirements on delay, it is difficult for a 5G network to directly replace the current commercial 4G based on vehicle networking systems.

Fortunately, in recent years, the communication technology based on beamforming has provided the possibility of large-scale data communication under the 5G network.

Beamforming is a combination of antenna technology and digital signal processing technology, which is used to transmit or receive directional signals. There are many branches of beamforming, and we will briefly introduce cooperative beamforming and coherent beamforming. Collaborative beamforming technology is a part of collaborative communication science [22, 23]. In collaborative beamforming, nodes are not uniformly distributed and, through calculating the number of optimal array nodes and selecting the optimal array node to establish a virtual antenna array, obtain a high gain beam for transmission and reduce the energy consumption and communication delay of nodes [24]. Even if the destination node is not within the transmission range of the transmission node, as long as there are enough idle nodes around the transmission node, collaborative beamforming can effectively improve the transmission range of a single node [25]. It changes the status quo of long-distance multi-hop wireless transmission, and this technology also improves the reliability and security of data transmission.

In this paper, we use coherent beamforming (see Figure 1; the red node is the transmitting node and the green node is the receiving node). Different from collaborative beamforming technology, CB (coherent beamforming) technology does not have strict requirements on node position and does not generate a high directional beam, and its transmission range is approximately a regular circle. CB technology greatly improves the transmission range of nodes through the cooperation of multiple nodes. When CB technology is used to transmit data, each node uses only one omnidirectional antenna. The transmitting node needs multiple other nodes to help its transmission, thereby improving the transmission distance by increasing the power gain. We aim to use coherent beamforming to give an optimal scheme for deploying the roadside CB-nodes so that we can transmit data to the edge server with a low cost on the 5G IoV scenario. The main contributions are summarized as follows:

- (1) We use the coherent beamforming technology to reduce the cost of network communication in the scenario of 5G Internet of Vehicles. CB-nodes should be arranged on both sides of the road to assist the vehicle to transmit data to the base station, thus saving the huge cost brought by the arrangement of multiple 5G base stations
- (2) We use CB technology to optimize the cost of communication infrastructure in the Internet of Vehicles scenario and set up a mathematical model; this model is a NP-hard problem, and it is difficult to find the optimal solution directly. For this reason, we design a heuristic algorithm for the Iterative Coherent Beamforming Node Design (ICBND) based on the greedy strategy to find the optimal number of CB-nodes in each subpart and obtain the approximate optimal solution by combining the optimal value of each subpart
- (3) We adopt the layout plan without cross-bit CB-nodes and the 5G base station as a comparative experiment, and we evaluate our algorithm through extensive

simulations. Simulation results demonstrate that the algorithm can achieve superior performance

The rest of the article is organized as follows. Section 2 discusses the related work. Section 3 gives the system model and the problem setting of the total node arrangement on a road and presents an Iterative Coherent Beamforming Node Design algorithm. In Section 4, two control variable methods are proposed and numerical results are given. Section 5 gives the conclusion.

## 2. Related Work

Internet of Vehicles is the development direction of intelligent transportation systems, which is of great significance for solving urban traffic problems. In recent years, the research on Internet of Vehicles has become increasingly hot, focusing on route selection, task transfer, unloading, and so on. In [26], in this paper, they propose a learning method for predicting quality of service (QoS), which achieves an automatic balance between exploration and utilization through automatic adjustment of super parameters based on maximum entropy enhanced learning. In [27], they proposed a multicast data transmission scheme with random delay and minimum cost constraints to optimize congestion of bottleneck vehicle node problem. In [28], the authors combined the vehicle position probability matrix, the vehicle position correlation matrix, and the recessive factor to study the potential function of the influence of the vehicle position; proposed a routing algorithm analysis based on the vehicle position (RAVP); and obtained more accurate vehicle trajectory prediction. In [29], aiming at the security problem of intelligent terminals in IoV, the authors propose two kinds of multimodal implicit authentication protocols based on intelligent terminal privacy protection. The security of the protocol is compared with other related protocols in terms of computing and communication overhead. The results show that the protocol has better security and efficiency. In [30], they propose a two-layer (sensing layer and data processing layer) sensing scheme to optimize link utilization rate and reduce resource consumption in high-speed mobile network IoV. In [31], the authors study the vehicle content cache decision method based on vehicle-to-vehicle collaboration in order to minimize the delay of vehicle content acquisition. In this paper, they propose an on-board content cache algorithm with perceptive delay to optimize the content cache obtained by the vehicle and to optimize the precache decision.

With the in-depth development of 5G, there are more and more articles on the combination of Internet of Vehicles and 5G. 5G has the characteristics of low latency and high data volume and has penetrated into various industries. In [32], the authors analyze and combine blockchain and SDN to effectively operate in the scenario of 5G and fog computing. This paper proposes a trust-based model for controlling network malicious behavior, which helps to relieve the pressure of the controller due to the ubiquitous processing. In [33], in order to solve the problem for distinguishing unloading targets of connected vehicle (IoCV) computing tasks, the

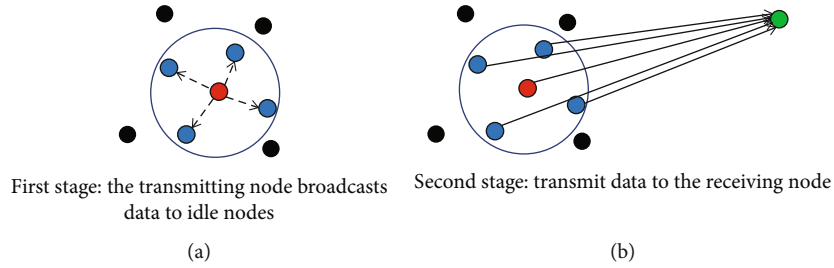


FIGURE 1: Two stages of data transmission in collaborative beamforming technology.

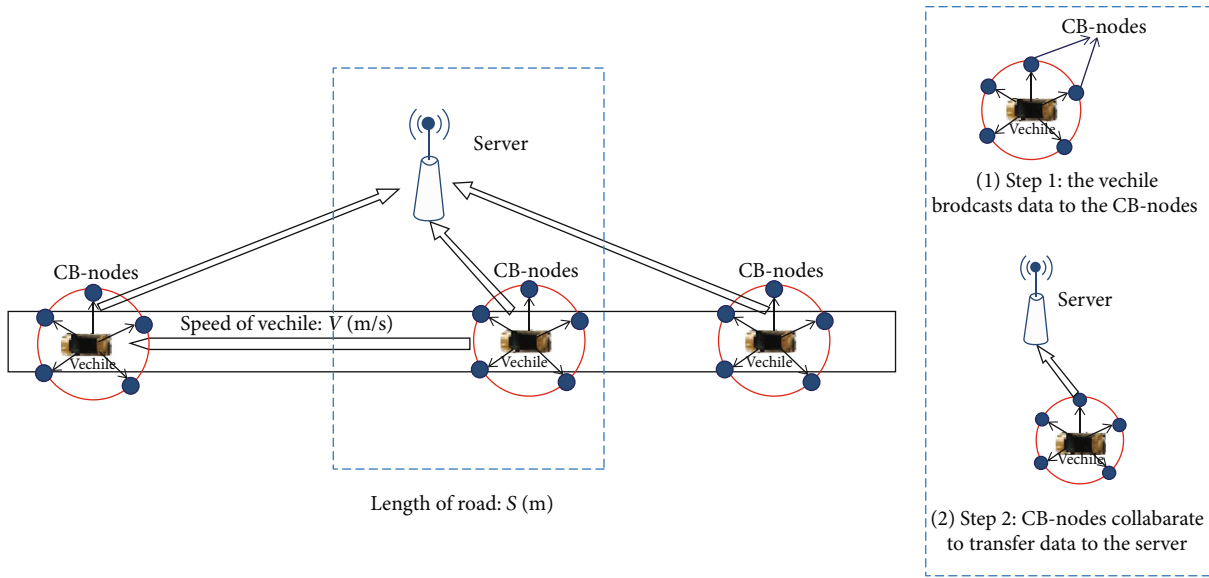


FIGURE 2: CB-nodes collaborate with the vehicle to transmit data.

authors design an adaptive calculating unloading method in 5G IoCV to optimize the unloading delays of tasks and the resource utilization of the edge system. In [34], the authors establish a novel architecture combining satellite networks, vehicle internet, and 5G cloud, which supports seamless and resource management effectively. With the rapid development of the 5G network, opportunities have been brought to the development of the Internet of Vehicles under the scenario of edge computing. In the edge computing IoV system supported by 5G, highly dense 5G base stations can provide rapidity and real-time calculation but, at the same time, increase the cost of network infrastructure [35]. Fortunately, beamforming can alleviate this problem, which improves the receiving power, expands the transmission range, and has low cost. There are several techniques for beamforming. In this paper, we will use coherent beamforming.

Coherent communications were studied mainly for applications in sensor networks or gain of transmitted power. In [36], the authors proved that the power gain of  $N$  senders and  $M$  receivers in coherent beamforming communication can reach  $N^2M$ , and the power and SIR gains obtained are higher than point-to-point transmission; thus, the transmission range of the node is greatly improved. In [37], they designed an open-loop coherent beamforming scheme for

MISO communications and analyzed the basic parameters and common parameters encountered in most open-loop coherent distributed array implementations. In [38], this paper proposed two plans of beamforming to improve the contrast-to-noise ratio; they are the fast minimum variance (FMV) and the fast coherent time delay and correlated pixel-based (FCCP) beamforming. In [39], the authors apply the coherent beamforming technique to MISO to achieve a similar range to minimize transmission power. In [40], an innovative spatiotemporal MIMO radar waveform design method is proposed, and coherent beamforming is applied to the radar system, so that the MIMO radar can meet the spatial domain transmit beamforming constraints and the time domain waveform orthogonality requirements. In [41], this article introduced a technique that can transmit beamforming signals from a node in a distributed radio network to a distant target node through a frequency selective channel to avoid the need for explicit channel status feedback from the destination and solved the problem of the variability of the irreversible effects caused by electronic interference. In [42], they developed the optimal adaptive transmission strategy and the optimal distributed beamforming and resource allocation strategy, and the numerical results proved the superiority of these strategies.

TABLE 1: Notations.

Notation	Description
$s_i$	One of the CB-nodes
$S_j$	One group of CB-nodes that collaborate for a transmission
$R_v$	The radius of the vehicle broadcast
$R_s$	The radius of the CB-node broadcast
$r_{v \rightarrow s_i}^{S_j}(t)$	The vehicle's transmitting data rate to $s_i$ for the set $S_j$ at time $t$
$r_{S_j \rightarrow b}(t)$	The transmitting data rate from the set $S_j$ to the edge server at time $t$
$v(t)$	The speed of the vehicle at time $t$
$d_{v \rightarrow s_i}(t)$	The distance between the vehicle and $s_i$ at time $t$
$d_{s_i \rightarrow b}$	The distance between $s_i$ and the edge server
$n(t)$	The number of CB-nodes that is needed in transmission at time $t$
$\sigma_{v \rightarrow s_i}^{S_j}(t)$	The signal-to-noise ratio (SINR) from the vehicle to $s_i$ for the set $S_j$ at time $t$
$\sigma_{S_j \rightarrow b}(t)$	The sum of SINR from all $s_i$ in $S_j$ to send data to the edge server at time $t$
$\lambda$	The pass loss index
$P_s$	Transmitting power of CB-node sending data to edge server
$P_v$	The power of the vehicle broadcasting data to the roadside CB-node
$W$	Bandwidth
$N_0$	White Gaussian noise power
$D$	The amount of data to be unloaded when the vehicle passes through this section of road

To our knowledge, most predecessors use CB technology to study network throughput and power problems. This is the first time that we try to use coherent beamforming technology to build a communication scheme for the Internet of Vehicles (IoV) under the 5G network. We try to expand the communication range and reduce the number of base stations so as to reduce the cost without reducing the communication requirements of vehicles. This paper analyzes the working model of coherent beamforming in a 5G vehicle network. We place coherent beamforming nodes reasonably and effectively on both sides of the road. By using coherent beamforming, these CB-nodes can collaborate to help the vehicle transfer data to the edge server. Compared with a 5G base station (BS) layout, the delay of this method is similar to that of the 5G base station layout, which greatly reduces the cost of infrastructure. Then, firstly, we design this problem as a mathematical model and reorganize the mathematical model. Secondly, we design an algorithm that is an Iterative Coherent Beamforming Node Design to solve the approximate optimal solution. Finally, we select two groups of comparative experiments and obtain that our algorithm is superior through the results of multiple experimental data.

### 3. System Model and Problem Definition

We first describe the system model (see Figure 2). Consider a part of a straight road with the length  $L$ . An edge server is located on the road side, near the center point of the road. A number of wireless nodes are placed on both sides of the road for helping communication. We consider when a vehi-

cle is passing the road; it will communicate with the edge server with the help of these nodes. The vehicle will first broadcast data to its nearby nodes, and then, these nodes will collaborate to send data to the edge server by using the coherent beamforming technique. We call these nodes as CB-nodes. Suppose the vehicle needs to transmit  $D$  data to the edge server, and suppose the vehicle will pass the road by  $T$  time. We want to give an optimal scheme for deploying the CB-nodes, so that we can use a minimum number of CB-nodes while guaranteeing that the vehicle can finish its transmission job in  $T$  time. Notice that when different vehicles pass the road, they may have different speeds and different transmission requirements. This may lead to different optimal solutions. So we suppose we design the optimal solution for the vehicle with the maximum speed (which leads a minimum passing time  $T$ ) and the maximum transmission requirement (which leads a maximum data  $D$ ). If the solution can meet this situation, then vehicles with any speed and transmission requirement can finish their jobs in this road. Table 1 summarizes the key parameter symbols in our article.

*3.1. Network Layer Model and Problem Formulation.* Denote  $s_i$  as one of the CB-nodes; denote  $N$  as the set of all CB-nodes, i.e.,  $s_i \in N$ ; and denote  $n$  as the number of  $N$ . In the scheduling time  $T$ , the whole data  $D$  may be divided into several pieces and be transmitted for several times, and several different CB-nodes may collaborate for each transmission. Denote  $S_j$  as one group of CB-nodes that collaborates for a transmission; denote  $M$  as the set of all CB-node groups, i.e.,  $S_j \in M$ ; and denote  $m$  as the number of  $M$ . Apparently  $m \ll 2^n - 1$ .

Notice that one CB-node  $s_i$  may be in different groups. Denoting  $D(S_j)$  as the data of  $S_j$ 's transmission, we have

$$D = \sum_{j=1}^m D(S_j). \quad (1)$$

One transmission has two stages. First, the vehicle broadcasts data to its nearby CB-nodes. Second, CB-nodes transmit data to the edge server by using a coherent beamforming technique. We use  $x_{s_i}(t)$ ,  $y_{s_i}(t)$ ,  $x_{S_j}(t)$ , and  $y_{S_j}(t)$  to indicate the transmission cases of a CB-node or a set at time  $t$ ; we have

$$\begin{aligned} x_{s_i}(t) &= \begin{cases} 1, & s_i \text{ receives the vehicle's broadcasting data at time } t, \\ 0, & \text{otherwise,} \end{cases} \\ x_{S_j}(t) &= \begin{cases} 1, & S_j \text{ receives the vehicle's broadcasting data at time } t, \\ 0, & \text{otherwise,} \end{cases} \\ y_{s_i}(t) &= \begin{cases} 1, & s_i \text{ transmits data by CB technique at time } t, \\ 0, & \text{otherwise,} \end{cases} \\ y_{S_j}(t) &= \begin{cases} 1, & S_j \text{ transmits data by CB technique at time } t, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (2)$$

For a set  $S_j$ , when it transmits or receives, all CB-nodes in  $S_j$  should be transmitted or received, respectively. We have

$$\begin{aligned} x_{s_i}(t) &\geq x_{S_j}(t) \quad (\forall i, s_i \in S_j, 0 \leq t \leq T), \\ y_{s_i}(t) &\geq y_{S_j}(t) \quad (\forall i, s_i \in S_j, 0 \leq t \leq T). \end{aligned} \quad (3)$$

For a CB-node  $s_i$  or a group  $S_j$ , it can only receive or transmit data at time  $t$ ; we have

$$\begin{aligned} x_{s_i}(t) + y_{s_i}(t) &\leq 1, \\ x_{S_j}(t) + y_{S_j}(t) &\leq 1. \end{aligned} \quad (4)$$

Only one group can receive the broadcasting data at time  $t$ , and only one group can transmit to the edge server at time  $t$ ; we have

$$\sum_{S_j \in M} x_{S_j}(t) \leq 1, \quad (5)$$

$$\sum_{S_j \in M} y_{S_j}(t) \leq 1. \quad (6)$$

Denote  $P_v$  as the transmission power of the vehicle; then, the vehicle's transmission range can be formulated as  $R_v = \sqrt[3]{P_v / \beta N_0}$ , where  $N_0$  is the noise power,  $\lambda$  is the pass loss index, and  $\beta$  is a constant. We can get a similar formulation about a single CB-node's transmission range  $R_s = \sqrt[3]{P_s / \beta N_0}$ ,

where we suppose all CB-nodes have the same transmission power  $P_s$ . Denoting  $\sigma_{v \rightarrow s_i}^{S_j}(t)$  as the signal-to-noise ratio (SINR) from the vehicle to  $s_i$  of the set  $S_j$  at time  $t$  and denoting  $d_{v \rightarrow s_i}(t)$  as the distance between the vehicle and  $s_i$  at time  $t$ , we have

$$\sigma_{v \rightarrow s_i}^{S_j}(t) = \frac{P_v d_{v \rightarrow s_i}^{-\lambda}(t) \cdot x_{S_j}(t)}{N_0} \geq \beta \cdot x_{S_j}(t) \quad (s_i \in S_j, S_j \in M). \quad (7)$$

Denote  $\sigma_{S_j \rightarrow b}(t)$  as the sum of SINR from all CB-nodes in the group  $S_j$  to send data to the edge server collaboratively at time  $t$ , and denote  $d_{s_i \rightarrow b}$  as the distance between  $s_i$  and the edge server. Since we use the CB technique for transmissions, which means several CB-nodes will cooperate for transmitting, we have

$$\sigma_{S_j \rightarrow b}(t) = \frac{P_s \left( \sum_{s_i \in S_j} \sqrt{d_{s_i \rightarrow b}^{-\lambda}} \right)^2 \cdot y_{S_j}(t)}{N_0} \geq \beta \cdot y_{S_j}(t) \quad (S_j \in M). \quad (8)$$

Denote  $r_{v \rightarrow s_i}^{S_j}(t)$  as the vehicle's transmitting data rate to  $s_i$  of the set  $S_j$  at time  $t$ , and denote  $r_{S_j \rightarrow b}(t)$  as the transmitting data rate from the set  $S_j$  to the edge server at time  $t$ . Denote  $W$  as the bandwidth. Since the transmitting data rate should not be larger than the channel capacity, we have

$$\begin{aligned} r_{v \rightarrow s_i}^{S_j}(t) &\leq W \log_2 \left( 1 + \sigma_{v \rightarrow s_i}^{S_j}(t) \right) \quad (s_i \in S_j, S_j \in M, 0 \leq t \leq T), \\ r_{S_j \rightarrow b}(t) &\leq W \log_2 \left( 1 + \sigma_{S_j \rightarrow b}(t) \right) \quad (S_j \in M, 0 \leq t \leq T). \end{aligned} \quad (9)$$

And considering the relationship between the transmitting data rate and the transmitting data of a set  $S_j$ , we have

$$D(S_j) = \int_{t=0}^T r_{v \rightarrow s_i}^{S_j}(t) dt = \int_{t=0}^T r_{S_j \rightarrow b}(t) dt \quad (s_i \in S_j, S_j \in M, 0 \leq t \leq T). \quad (10)$$

And the second step should be started after finishing the first step. We have

$$\int_{t=0}^{\eta} r_{v \rightarrow s_i}^{S_j}(t) dt \geq \int_{t=0}^{\eta} r_{S_j \rightarrow b}(t) dt \quad (s_i \in S_j, S_j \in M, 0 < \eta \leq T). \quad (11)$$

Denote  $n_{\max}$  as the maximum number of roadside CB-nodes arranged on. Apparently, if we deploy enough CB-nodes by the roadside, the vehicle will always complete the communication properly. So suppose we first deploy enough CB-nodes by the roadside, then we try to find as many nodes as possible which are not used in the whole scheduling time,



we will get the optimal solution. That is, we are trying to find CB-nodes  $s_i$  that satisfy  $y_{s_i}(t) = 0$  when each vehicle passes through the entire road. We can set a binary variable  $z_i$  to indicate whether the node on the road is valid or not, i.e.,

$$z_i = \begin{cases} 1, & \exists y_{s_i}(t) = 1, 0 \leq t \leq T, \\ 0, & \forall y_{s_i}(t) = 0, 0 \leq t \leq T. \end{cases} \quad (12)$$

Based on the above discussions, we can get the formula for the final optimization cost, that is,

$$\begin{aligned} \min & \quad \sum_{i=1}^{n_{\max}} z_i, \\ \text{s.t.} & \quad (1), (3) - (12). \end{aligned} \quad (13)$$

However, notice that in Equation (13), we do not know which specific CB-node is in each set  $S_j$ , and the number of all sets  $m$  may be a very large number. We also notice that  $x_{s_i}(t)$ ,  $y_{s_j}(t)$  are continuous variables about time  $t$ , which means we have infinite variables. So Equation (13) cannot be solved directly. We need to find some way to reformulate the problem model so that it can be solved.

**3.2. Problem Refinement.** In Section 3.1, we give the problem formulation. However, this problem cannot be solved directly. We need to reformulate the problem model for solving it. In this subsection, we will give the problem model reformulation.

We notice that for the problem model (13), we divide the whole data  $D$  into many small parts  $D(S_j)$ , and for each part, we use a different CB-node set  $S_j$  for serving it. So the CB-node set  $S_j$  is a very important variable. If we can design an algorithm for establishing suitable  $S_j$ , then we may find an easy way to solve Equation (13). To do that, we need to answer three problems: (i) How many parts we should divided at least for  $D$ ? (ii) For each CB-node set  $S_j$ , how many nodes should be included? (iii) Are there any CB-nodes in different CB-node sets?

We now discuss the first problem. Notice that the vehicle usually travels with a constant speed on roads. Based on this, we can simply assume that the vehicle travels with a speed  $v$  on the whole time  $T$ , then we have  $v \geq v_{\max} = L/T$ . Define the amount of data transmitted by each part as  $D(S_j)$ . Since the transmission radius of the CB-node is  $R_s$ , the time of each transmission of the vehicle is the time when the vehicle passes through the transmission range of the CB-node which is  $2R_s$ . Therefore, it should be ensured that the vehicle completes the transmission within  $2R_s/v_{\max}$  time.

Then, assuming that the transmission range of the CB-nodes in each segment is not intersected with the other segments, we will equally divide the whole road for  $h \geq h_{\min} = L/2R_s$  parts at least, which means we can divide the whole road equally into  $h(=m)$  path parts, and in each path part  $l_j$ , we will arrange a set  $S_j$  for helping transmitting  $D(S_j)$  data to the edge server. We also notice that when the vehicle enter-

ing a path part, the distance between it and the CB-nodes in this part will not change too much. So we can consider  $d_{v \rightarrow s_i}(t)$  as a constant  $d_v$  approximately, and the data transmission rate  $r_{v \rightarrow s_i}^{S_j}$  as a constant  $r_v$  approximately.

For the second problem, since the number of CB-nodes that is needed for each set is decided by the distance between the edge server and the set, and now we have a known  $S_j$  that is arranged for  $l_j$ , we will approximately use the distance between the center point of  $l_j$  and the edge server as the distance between  $S_j$  and the edge server. Denoting the distance as  $d(S_j)$ , and denoting the number of CB-nodes that is needed as  $n(S_j)$ , we have

$$n(S_j) = \left\lceil \frac{d(S_j)}{R_s} \right\rceil. \quad (14)$$

Since we consider that the vehicle has a constant speed  $v$ , and the road is divided equally, we will have that the vehicle passes each road path part with the same length of the time slot. Denote time slot as  $t_i$  ( $i = 1, 2, \dots, h$ ), and we have  $t_1 = t_2 = \dots = t_h$ .

Then, we have

$$D(S_j) = \frac{D}{h} = r_v \cdot t_j \leq W \log_2 \left( 1 + \sigma_{v \rightarrow s_i}^{S_j}(t_j) \right) \cdot t_j. \quad (15)$$

Notice that based on the first discussion and the second discussion, we can get a feasible solution, only if we put enough CB-nodes in each CB-node set.

In Section 3.2, we segmented the time and road and linearized the NP-hard problem in the previous section to facilitate the subsequent solution.

For the third problem, does the same CB-node exist in different sets of CB-nodes? The result of this problem is affected by the length of the segment and the propagation range of the CB-node. We refer to the same CB-node in different sets of CB-nodes as a cross-bit CB-node. In the following, we will set up an algorithm named as the ICBND algorithm. And this algorithm describes the steps of segmentation, and it considers that the same CB-node exists in multiple different sets of nodes. In order to approximate the optimal total number of CB-nodes, according to the number of segments, the minimum number of CB-nodes that is needed for each segment to work separately is obtained. Then, the number of CB-nodes existing in multiple different CB-node sets is obtained. The number of repeated CB-nodes is removed to obtain the optimal total number of CB-nodes. The specific steps are described in Section 3.3.

**3.3. Algorithms.** For the third problem, we notice the whole road has been divided into  $h (\geq h_{\min})$  path parts. We only have a requirement of the minimum number of path parts and not a maximum one. So in the simulation, we will set a suitable maximum number of path parts (i.e.,  $h_{\max}$ ) and try to find the optimal one between  $h_{\min}$  and  $h_{\max}$ . Since the vehicle has a permanent transmission range  $R_v$ , it is easy to find that with the different number of road path parts, several

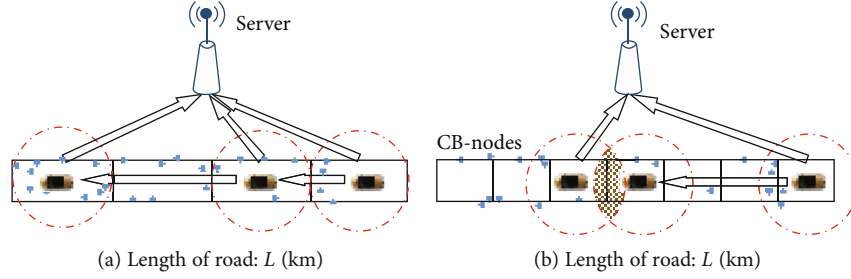


FIGURE 3: CB-node layout scheme.

CB-node sets can be in one vehicle's transmission range (see Figure 3). If the length of each segment is equal to the transmission range of the CB-node, there will be no intersection of the transmission range between the two segments, as shown in Figure 3(a). If the length of each segment is less than the transmission range of the CB-node, there will be an intersection of the transmission range between the two segments, as shown in the shaded section in Figure 3(b). In these situations, in order to describe the problem more conveniently below, we assume that the same CB-node with different sets of CB-nodes is a cross-bit CB-node; if we put the cross-bit CB-nodes in the common parts, we will save more CB-nodes.

Based on these discussions, we will try to propose a heuristic algorithm to solve this problem. The main idea of our algorithm is based on iterative steps. We call our algorithm as the Iterative Coherent Beamforming Node Design (ICBND) algorithm. In the following, we give the main four steps of the algorithm.

First, initialize  $h_{\max}$ ,  $h_{\min}$ . Determine that the road is divided into segment  $h$ , and calculate the distance from the center of each segment to the edge server. Then, the number of CB-nodes required for each segment is calculated according to Equation (6), and the number of CB-nodes required for each segment is temporarily stored in an array.

Second, since the transmission range of each CB-node is fixed; the length of each segment affects the existence of the same CB-node in several different sets of CB-nodes. The longer the length of the segment, the less likely the CB-node is to exist in multiple CB-node sets. We refer to CB-nodes with the same CB-node in different sets of CB-nodes as CB-nodes of cross-position. Input the broadcast range and segment length of the CB-node to get the number of cross-bit CB-nodes. We define the number of CB-nodes at the intersection of segment  $l_j$  and segment  $l_{j+1}$  as  $x_j$ . In the previous step, we figure out the number of CB-nodes needed for each segment, then we place as many CB-nodes at the intersection as possible to improve the utilization rate of CB-nodes and reduce costs. A CB-node is placed outside the crossing position, and the remaining CB-nodes are placed in the crossing position.

Third, after we get the number of cross-bit CB-nodes, each line segment has at least one receiving CB-node, and other CB-nodes can be placed on the intersection of the two line segments.

Fourth, add the minimum number of CB-nodes in each segment and the number of CB-nodes at the intersection to get the minimum value of the total number of CB-nodes.

**3.4. Complexity.** We first show the complexity for the selection step of the number of segments  $h$ . Since we need to go through all possible segment schemes, the number of iterations is  $h_{\max} - h_{\min} + 1$ . Define  $f = h_{\max} - h_{\min} + 1$ , then the complexity of this step is  $O(fn)$ .

Then, we show the minimum number of CB-nodes in each segment scheme. Since this step is a selection step nested within the first step, the complexity of this step is  $O(n^2)$ . And we show the complexity for polynomial in each iteration:

- (i) First, we calculate the distance between the vehicle and the edge server in an array, then calculate the minimum number of CB-nodes required for each segment by the formula in an array. The complexity of this part is  $O(hn)$
- (ii) Second, we need to calculate the number of cross-bit CB-nodes  $x_j$  and the number of CB-nodes except  $x_j$  in each segment. Generally, one CB-node can be placed in each segment except the intersection position, and the remaining number of CB-nodes can be used as the  $x_j$  value. The complexity of this part is  $O(hn)$
- (iii) Third, we need to go through all segments and add the number of CB-nodes of each segment and all the number of cross-bit CB-nodes. The complexity of this part is  $O(hn)$

In summary, the time complexity of our algorithm is  $O(fhn^2)$ .

## 4. Simulation

In this section, we will present the simulation results. We will first give a specific layout scheme for a particular network and then give the comparison results of more network schemes. The parameters involved are set as follows, the straight-line distance between the edge server and the road is  $a = 100$  m. The noise power  $N_0$  is  $10^{-7}$  W, and the road strength loss factor  $\lambda = 3$ . The total bandwidth of road strength  $W = 3.5$  GHz. The speed limit in urban areas is

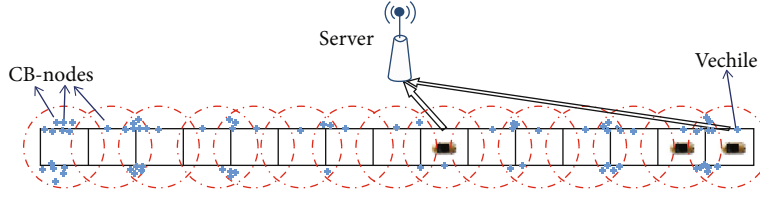


FIGURE 4: The optimal CB-node layout scheme in a specific scenario.

**Require:** initializing  $h_{\min}$ ,  $h_{\max}$ ; initial the CB-nodes sum = 0;

1: Initial  $x = 0$ ;

2: Initial  $j = 1, h = h_{\min}$ ;

3: **Repeat**

4: **Repeat**

5: Calculate the number of cross-bit CB-nodes  $x$ ;

6: Calculate the distance between the vehicle and the edge server at time  $t$ :  $n(S_j)$ ;

7: Calculate the number of CB-nodes in the each region:  $n(S_j) = \lceil d(S_j)/R_s \rceil$ ;

8: Determine the relation between  $R_s$  and the value between  $h_{\min}$  and  $h_{\max}$ , then calculate that the same CB-node exists in several different sets of CB-nodes;

9: Calculate the value of  $x_j$ , subtract  $x_j$  from the number of CB-nodes in this segment:  $n(S_j) = n(S_j) - x_j$ ; sum = sum +  $n(S_j)$ ;

10: sum = sum +  $n(S_j)$ ;

11: **Until** the vehicle ran the last part of the road,  $j = h$ ;

12: sum = sum +  $\sum x_j$ ;

13: Get one kind of task assignment scheme

14: **Until**  $h = h_{\max}$ ;

15: Select the best assignment scheme to be the assignment scheme for this time slice.

ALGORITHM 1: Algorithm for Iterative Coherent Beamforming Node Design.

TABLE 2: The total number of CB-nodes in a scheme with cross-bit CB-node.

Serial number	$h$	Total number of CB-nodes	Serial number	$h$	Total number of CB-nodes	Serial number	$h$	Total number of CB-nodes
1	14	104	2	15	76	3	16	76
4	17	85	5	18	83	6	19	94
7	20	92	8	21	102	9	22	100
10	23	111	11	24	110	12	25	120
13	26	118	14	27	131	15	28	126
16	29	95	17	30	94	18	31	105
19	32	103	20	33	101	21	34	115
22	35	112	23	36	112	24	37	123
25	38	120	26	39	121	27	40	131
28	41	131	29	42	126	30		

40 m/s, and the speed limit on highways is 110 m/s. Therefore, a range of vehicle speed  $v$  is set as [40,110]. Here, we take the speed as 30 m/s. Denote the vehicle's broadcast range as the CB-node's broadcast range.

*4.1. A Special Case with CB-Node Layout.* In this section, under the condition that the road length is  $L = 4000$  meters, the transmission power  $P_s$  of the CB-node is 0.3 W, and the broadcast radius data is  $R_s = 144$  meters, we will calculate the optimal total number of CB-nodes and simulate the optimal CB-node layout scheme (see in Figure 4).

We use Algorithm 1 to compute data for a series of scenarios with a cross-bit CB-node, and we calculate a set of data that is a scheme without a cross-bit CB-node. The data of the two schemes are recorded in Tables 2 and 3.

Table 2 is the scheme with cross-bit CB-nodes, and Table 3 is the scheme without cross-bit CB-nodes.  $h$  is the number of line segments divided by the road, and sum is the total number of CB-nodes arranged by the road. According to the data in Tables 2 and 3, the total number of CB-nodes in Table 2 is always less than the total number of CB-nodes in Table 3, so the scheme with cross-bit CB-

TABLE 3: The total number of CB-nodes in a scheme without cross-bit CB-node.

Serial number	$h$	Total number of CB-nodes	Serial number	$h$	Total number of CB-nodes	Serial number	$h$	Total number of CB-nodes
1	14	104	2	15	113	3	16	122
4	17	127	5	18	136	6	19	143
7	20	150	8	21	157	9	22	168
10	23	173	11	24	182	12	25	189
13	26	196	14	27	209	15	28	210
16	29	217	17	30	224	18	31	231
19	32	240	20	33	247	21	34	258
22	35	263	23	36	270	24	37	281
25	38	284	26	39	295	27	40	300
28	41	309	29	42	314	30		

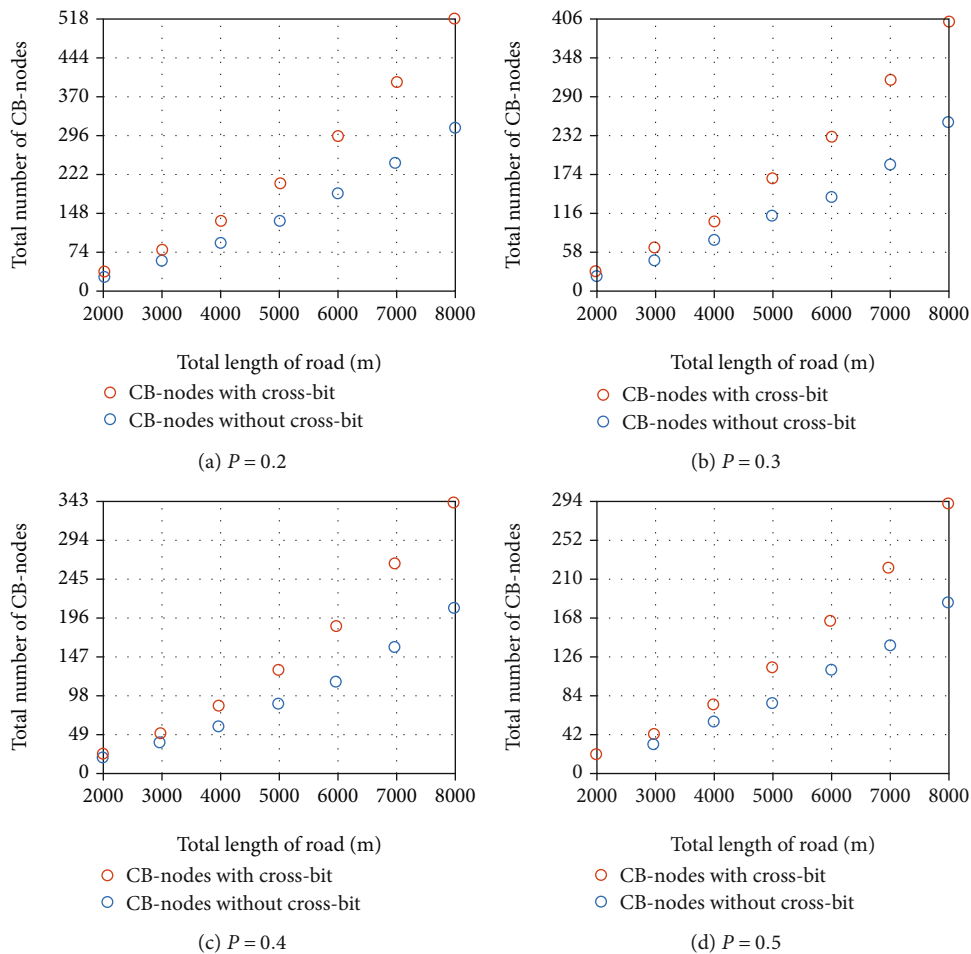


FIGURE 5: Comparisons for the total number of optimal CB-nodes under different levels of power and different path lengths.

nodes has advantages. Table 2 shows that when  $h$  is 15, the total number of CB-nodes is optimal, and the optimal scheme is one server and the total number of CB-nodes is 98. Based on this, we adopt the optimal CB-node layout

scheme to simulate the layout of real road CB-nodes, as shown in Figure 4.

In the above fixed parameter scenario, the optimal CB-node layout plan is shown in Figure 4. The number of CB-

nodes in each segment can meet the requirements of vehicle transmission data. We try to place as many CB-nodes in the intersection as possible, and the CB-nodes outside the intersection should be placed as little as possible. CB-nodes are randomly placed in fixed areas. We can see that for the CB-node layout, the closer to the edge server, the less CB-nodes are arranged, and the farther away from the edge server, the more CB-nodes are arranged.

According to market research, CB-node prices range from about 5.7 dollars to 14.3 dollars. Here, we set the cost of a CB-node at 15 dollars. The cost of a 5G base station ranges from about 28571 dollars to 71429 dollars, so we set the cost of a 5G base station at 28000 dollars. The coverage of 5G base stations is very small. For normal communication, a 5G base station needs to be deployed every 200 meters. According to the data in Table 2, the optimal total number of CB-nodes is 76 and the optimal scheme is 76 CB-nodes and one edge server. According to this arrangement, the optimal cost of the scheme with cross-bit CB-nodes we need is about 29140 dollars. The length of the road is  $L = 4000$  meters, so 20 5G base stations are needed and the cost is about 560000 dollars. As a result, we could save about 530860 dollars based on our plan layout.

**4.2. General Case.** In this section, to make the experimental results more realistic; we change  $P_s$  to get four sets of data. First, we collect a series of  $L$  values, from 2000 meters to 8000 meters, including 2000 meters, 3000 meters, 4000 meters, 5000 meters, 6000 meters, 7000 meters, and 8000 meters. Then, according to the formula  $R_s = \sqrt[3]{P_s/\beta N_0}$ , by changing the power of CB-node  $P_s$ , from 0.2 W to 0.5 W, including 0.2 W, 0.3 W, 0.4 W, and 0.5 W, we make the value of  $R_s$  take 126 meters, 144 meters, 159 meters, and 170 meters. After the experiment, a series of optimal CB-node values are obtained, as shown in Figure 5.

In Figure 5, the blue circle represents the total number of CB-nodes required for a scenario with cross-bit CB-nodes, and the red circle represents the total number of CB-nodes required for a scenario without cross-bit CB-nodes. Whether you change the length of the road or the transport range of CB-node, the blue circle is always less than the red circle. It can be seen that the total number of CB-nodes required by the scheme with cross-bit CB-nodes is the optimal, so the cost is the lowest.

As shown in Figure 5(b), under a series of  $L$  scenarios, the optimal total number of CB-nodes obtained by the scheme with cross-bit CB-nodes is 24, 46, 76, 112, 141, 189, and 250, respectively. The corresponding costs are approximately 28360 dollars, 28690 dollars, 29140 dollars, 29680 dollars, 30115 dollars, 30835 dollars, and 31750 dollars. When  $L$  values are 2000 meters, 3000 meters, 4000 meters, 5000 meters, 6000 meters, 7000 meters, and 8000 meters, the number of 5G base stations required in the layout plan of 5G base stations is 10, 15, 20, 25, 30, 35, and 40, respectively. The corresponding costs are 280000 dollars, 420000 dollars, 560000 dollars, 700000 dollars, 840000 dollars, 980000 dollars, and 1120000 dollars, respectively. Based on this, it can be seen that adopting our method can greatly reduce the cost of infrastructure.

## 5. Conclusion

In this work, we study the use of coherent beamforming technology to reduce communication costs. We build a general model for this problem, with the goal of reducing the communication cost as much as possible while satisfying the transmission conditions. We propose an effective heuristic algorithm ICBND, using the iterative method to calculate the total number of the minimum CB-nodes on the whole road. Specifically, ICBND divides the road into  $h$  segments, calculates the minimum number of CB-nodes for each segment, and sums them up, which is easy to achieve in practice. In the simulation experiment, we compare the scheme with the cross-bit CB-node, the scheme without the cross-bit CB-node, and the scheme of the 5G base station layout. In addition, comparative experiments show that the arrangement of the minimum number of CB-nodes of ICBND has stable and superior performance.

## Data Availability

There is no data set for this article.

## Disclosure

A preliminary version of the material in this paper has been presented at the conference in WASA 2020 (International Conference on Wireless Algorithms Systems and Applications).

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This article was supported by the Key Research and Development Project in Anhui Province (Grant No. 201904a06020024), the National Key Research and Development Plan (Grant No. 2018YFB2000505), and the National Natural Science Foundation of China (Grant No. 61806067).

## 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] Z. Xiong, W. Li, Q. Han, and Z. Cai, "Privacy-preserving auto-driving: a GAN-based approach to protect vehicular camera data," in *2019 IEEE International Conference on Data Mining (ICDM)*, pp. 668–677, Beijing, China, 2019.
- [3] X. Guan, Y. Huang, Z. Cai, and T. Ohtsuki, "Intersection-based forwarding protocol for vehicular ad hoc networks," *Telecommunication Systems*, vol. 62, no. 1, pp. 67–76, 2016.
- [4] L. Claussmann, M. Revilloud, D. Gruyer, and S. Glaser, "A review of motion planning for highway autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 5, pp. 1826–1848, 2020.

- [5] Z. Cai and T. Shi, "Distributed query processing in the edge assisted IoT data monitoring system," *IEEE Internet of Things Journal*, pp. 1–1, 2020.
- [6] X. Hou, Z. Ren, J. Wang et al., "Reliable computation offloading for edge-computing-enabled software-defined IoV," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7097–7111, 2020.
- [7] R. Xie, Q. Tang, Q. Wang, X. Liu, F. R. Yu, and T. Huang, "Collaborative vehicular edge computing networks: architecture design and research challenges," *IEEE Access*, vol. 7, pp. 178942–178952, 2019.
- [8] J. Wang, Z. Cai, and J. Yu, "Achieving personalized k-anonymity-based content privacy for autonomous vehicles in CPS," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4242–4251, 2020.
- [9] H. Peng, Q. Ye, and X. Shen, "Spectrum management for multi-access edge computing in autonomous vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 7, pp. 3001–3012, 2020.
- [10] W. Na, S. Jang, Y. Lee, L. Park, N. Dao, and S. Cho, "Frequency resource allocation and interference management in mobile edge computing for an internet of things system," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4910–4920, 2019.
- [11] Q. Cui, J. Zhang, X. Zhang, K. Chen, X. Tao, and P. Zhang, "Online anticipatory proactive network association in mobile edge computing for IoT," *IEEE Transactions on Wireless Communications*, vol. 19, no. 7, pp. 4519–4534, 2020.
- [12] 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.
- [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] M. N. H. Nguyen, C. W. Zaw, K. Kim, N. H. Tran, and C. S. Hong, "Let's share the resource when we're co-located: collocation edge computing," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5618–5633, 2020.
- [15] S. Huang, B. Lv, R. Wang, and K. Huang, "Scheduling for mobile edge computing with random user arrivals-an approximate MDP and reinforcement learning approach," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 7735–7750, 2020.
- [16] S. A. A. Shah, E. Ahmed, M. Imran, and S. Zeadally, "5G for vehicular communications," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 111–117, 2018.
- [17] A. Gupta and R. K. Jha, "A survey of 5G network: architecture and emerging technologies," *IEEE Access*, vol. 3, pp. 1206–1232, 2015.
- [18] L. Chettri and R. Bera, "A comprehensive survey on internet of things (IoT) toward 5G wireless systems," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 16–32, 2020.
- [19] S. Li, L. Xu, and S. Zhao, "5G internet of things: a survey," *Journal of Industrial Information Integration*, vol. 10, pp. 1–9, 2018.
- [20] X. Cheng, C. Chen, W. Zhang, and Y. Yang, "5G-enabled cooperative intelligent vehicular (5GenCIV) framework: when Benz meets Marconi," *IEEE Intelligent Systems*, vol. 32, no. 3, pp. 53–59, 2017.
- [21] M. Fakh, A. Diallo, P. Le Thuc, R. Staraj, O. Mourad, and E. Rachid, "Characteristic mode theory to enhance the isolation level for full-duplex 5G in mobile handsets," in *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*, pp. 81–85, Oulu, Finland, 2019.
- [22] S. Jayaprakasam, S. K. Abdul Rahim, C. Y. Leow, and M. F. Mohd Yusof, "Beampattern optimization in distributed beamforming using multiobjective and metaheuristic method," in *2014 IEEE Symposium on Wireless Technology and Applications (ISWTA)*, pp. 86–91, Kota Kinabalu, Malaysia, 2014.
- [23] S. Jayaprakasam, S. K. A. Rahim, and C. Y. Leow, "Distributed and collaborative beamforming in wireless sensor networks: classifications, trends, and research directions," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2092–2116, 2017.
- [24] G. Sun, Y. Liu, J. Zhang, A. Wang, and X. Zhou, "Node selection optimization for collaborative beamforming in wireless sensor networks," *Ad Hoc Networks*, vol. 37, pp. 389–403, 2016.
- [25] B. Bjar Haro, S. Zazo, and D. P. Palomar, "Energy efficient collaborative beamforming in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 62, no. 2, pp. 496–510, 2014.
- [26] W. Xiong, Z. Lu, B. Li et al., "A self-adaptive approach to service deployment under mobile edge computing for autonomous driving," *Engineering Applications of Artificial Intelligence*, vol. 81, pp. 397–407, 2019.
- [27] L. Zhang, W. Cao, X. Zhang, and H. Xu, "MAC2: enabling multicasting and congestion control with multichannel transmission for intelligent vehicle terminal in internet of vehicles," *International Journal of Distributed Sensor Networks*, vol. 14, no. 8, Article ID 155014771879358, 2018.
- [28] L. Wang, J. Gui, X. Deng, F. Zeng, and Z. Kuang, "Routing algorithm based on vehicle position analysis for internet of vehicles," *IEEE Internet of Things Journal*, vol. 7, no. 12, pp. 11701–11712, 2020.
- [29] F. Wei, S. Zeadally, P. Vijayakumar, N. Kumar, and D. He, "An intelligent terminal based privacy-preserving multi-modal implicit authentication protocol for internet of connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–13, 2020.
- [30] G. Sun, L. Song, H. Yu, X. Du, and M. Guizani, "A two-tier collection and processing scheme for fog-based mobile crowdsensing in the internet of vehicles," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1971–1984, 2021.
- [31] X. Huang, K. Xu, Q. Chen, and J. Zhang, "Delay-aware caching in internet of vehicles networks," *IEEE Internet of Things Journal*, no. article 1, 2021.
- [32] J. Gao, K. O. Obour Agyekum, E. B. Sifah et al., "A blockchain-SDN-enabled internet of vehicles environment for fog computing and 5G networks," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4278–4291, 2020.
- [33] X. Xu, X. Zhang, X. Liu, J. Jiang, L. Qi, and M. Z. A. Bhuiyan, "Adaptive computation offloading with edge for 5G-envisioned internet of connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2020.
- [34] M. LiWang, S. Dai, Z. Gao, X. Du, M. Guizani, and H. Dai, "A computation offloading incentive mechanism with delay and cost constraints under 5G satellite-ground IoV architecture," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 124–132, 2019.
- [35] S. Wan, R. Gu, T. Umer, K. Salah, and X. Xu, "Toward offloading internet of vehicles applications in 5G networks," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–9, 2020.

- [36] Y. Shi and Y. E. Sagduyu, "Coherent communications in self-organizing networks with distributed beamforming," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 760–770, 2020.
- [37] J. A. Nanzer, R. L. Schmid, T. M. Comberiate, and J. E. Hodkin, "Open-loop coherent distributed arrays," *IEEE Transactions on Microwave Theory and Techniques*, vol. 65, no. 5, pp. 1662–1672, 2017.
- [38] C. Bai, X. Zhang, X. Qiao, Y. Sang, H. Zhong, and M. Wan, "Ultrasound transcranial imaging based on fast coherent-time-delay and correlative pixel-based beamforming," in *2018 IEEE International Ultrasonics Symposium (IUS)*, pp. 1–4, Kobe, Japan, 2018.
- [39] D. Scherber, P. Bidigare, R. O'Donnell et al., "Coherent distributed techniques for tactical radio networks: enabling long range communications with reduced size, weight, power and cost," in *MILCOM 2013-2013 IEEE military communications conference*, pp. 655–660, San Diego, CA, USA, 2013.
- [40] H. Deng, Z. Geng, and B. Himed, "Mimo radar waveform design for transmit beamforming and orthogonality," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 3, pp. 1421–1433, 2016.
- [41] T. P. Bidigare, U. Madhow, D. R. Brown et al., "Wideband distributed transmit beamforming using channel reciprocity and relative calibration," in *2015 49th Asilomar conference on signals, systems and computers*, pp. 271–275, Pacific Grove, CA, USA, 2015.
- [42] A. G. Marques, Xin Wang, and G. B. Giannakis, "Minimizing transmit power for coherent communications in wireless sensor networks with finite-rate feedback," *IEEE Transactions on Signal Processing*, vol. 56, no. 9, pp. 4446–4457, 2008.

## Research Article

# Dynamically Subarray-Connected Hybrid Precoding Scheme for Multiuser Millimeter-Wave Massive MIMO Systems

Guangyan Liao<sup>1</sup> and Feng Zhao <sup>2</sup>

<sup>1</sup>Key Laboratory of Cognitive Radio and Information Processing, Guilin University of Electronic Technology, Guilin 541004, China

<sup>2</sup>Key Laboratory of Complex System Optimization and Big Data Processing, Guangxi Colleges and Universities, Yulin Normal University, Yulin 537000, China

Correspondence should be addressed to Feng Zhao; zhaofeng@guet.edu.cn

Received 17 February 2021; Revised 7 April 2021; Accepted 19 April 2021; Published 28 April 2021

Academic Editor: Yan Huang

Copyright © 2021 Guangyan Liao and Feng Zhao. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Hybrid precoding is widely used in millimeter wave (mmWave) massive multiple-input multiple-output (MIMO) systems. However, most prior work on hybrid precoding focused on the fully connected hybrid architectures and the subconnected but fixed architectures in which each radio frequency (RF) chain is connected to a specific subset of the antennas. The limited work shows that dynamic subarray architectures address the tradeoff between achievable spectral efficiency and energy efficiency of mmWave massive MIMO systems. Nevertheless, in the multiuser hybrid precoding systems, the existing dynamic subarray schemes ignore the fairness of users and the problem of user selection. In this paper, we propose a novel multiuser hybrid precoding scheme for dynamic subarray architectures. Firstly, we select a multiuser set among all users according to the analog effective channel information of the base station (BS) and then design the subset of the antennas to each RF by the fairness antenna-partitioning algorithm. Finally, the optimal analog precoding vector is designed according to each subarray, and the digital precoding is designed by the minimum mean-squared error (MMSE) criterion. The simulation results show that the performance advantages of the proposed multiuser hybrid precoding scheme for dynamic subarray architectures.

## 1. Introduction

Millimeter-wave massive MIMO, as a key technology in the next-generation wireless systems, can achieve multigigabit data rates benefiting from its abundant frequency resource [1] and provide highly directional beamforming gains to compensate for the path loss of mmWave signals [2]. For mmWave massive MIMO systems, hybrid precoding not only solves the problem that the fully digital precoding is unsuitable for these systems, but also can achieve performance close to a full digital precoding [3, 4].

The studies of hybrid precoding focused on the fully connected architecture [5–8] and partially connected architecture [9–11], and there are two main ways to solve the hybrid precoding problem. One is the joint analog domain precoding and digital domain precoding simultaneously [5–7, 9]. And another is the analog domain precoding designed by the channel gain at first and then the digital

domain precoding designed by the analog equivalent channel [12–15], and this way is usually applied to multiuser mmWave systems. However, most previous works did not explore the problem of the tradeoff between spectral efficiency and energy efficiency.

Limited works have been carried out for dynamic subarray-connected structures were proposed in [16–20]. This new structure is different from the previous structure, which uses a dynamic subarray connection structure to connect the RF chain and antenna according to the channel state information of the users. In [16], adaptive hybrid precoding (AHP) was developed to achieve the highest system spectral efficiency, but it is only applicable for the number of employed RF chains which is equal to the number of users. In [17], based on knowing the long-term channel characteristics, a technique of dynamically hybrid subarrays was developed, but the user's fairness was ignored. In [18], consider the number of RF chains which is more than or equal to that of



scheduled users and utilized AHP based on the singular value decomposition (SVD). However, that scheme does not study the problem of user selection when the number of users increases. Moreover, in [19], utilize the subchannel difference of the users, and a dynamically connected hybrid precoding was developed. This scheme allocates a different number of antennas for each RF chain, but sometimes some RF chains cannot allocate an antenna, which was unfair to some users. User selection was considered in [20], and a multiuser hybrid precoding framework based on codebooks was proposed, which adopted the antenna-partitioning algorithm based on the maximum increment of the signal-to-interference-noise ratio (SINR), but it has the same problem as [19].

In this paper, we proposed a new antenna-partitioning algorithm and hybrid analog/digital precoding schemes for the dynamic subarray characteristic. The main contributions of this paper can be summarized as follows.

- (1) Proposing a new antenna-partitioning algorithm for multiuser mmWave system with a dynamic subarray characteristic. The algorithm guaranteed fairness because each user can obtain approximately equal channel gain. The new antenna-partitioning algorithm proposed in this paper avoids the problem that the number of antennas in some subarrays which is zero may occur in [20], and it can be used to the number of transmitting antennas and the number of subarrays which is not multiple. However, the AHP scheme in [16] is only applicable when the number of transmitting antennas is a multiple of the number of subarrays
- (2) The dynamic hybrid structure model is obtained by the new antenna partition algorithm, which is based on the maximum channel gain of each user rather than SINR. Therefore, the spectrum efficiency of the system can be improved by eliminating the interuser interference through digital baseband precoding. Firstly, we design the analog precoding based on phase-only precoding. Then, in order to eliminate the interuser interference and ensure the high performance of the system under low SNR, we design the digital baseband precoding based on the minimum mean square error criterion. The simulation results show the advancement of our scheme

The remainder of the paper is as follows: Section 2 describes the multiuser massive MIMO system model and millimeter-wave channel model. The problem description is proposed in Section 3. Section 4 provides a detailed description of the proposed solution. Simulation results are presented in Section 5. Finally, conclusions are drawn in Section 6.

*Notation.* In this paper, we use the notation as the following:  $\mathbf{X}$  is a matrix,  $\mathbf{x}$  is a vector, and  $x$  is a scalar.  $(\mathbf{X})^T$ ,  $(\mathbf{X})^H$ , and  $(\mathbf{X})^\dagger$  denote the transpose, transpose-conjugate operation, and pseudo-inverse of  $\mathbf{X}$ , respectively.  $\|\mathbf{X}\|_F$  and  $|\mathbf{X}|$  denote the Frobenius norm and determinant of  $\mathbf{X}$ .  $\text{Tr}\{\mathbf{X}\}$  and  $\mathbb{E}\{\mathbf{X}\}$  denote the trace and expectation of  $\mathbf{X}$ .  $\mathcal{CN}(\mathbf{m}, \mathbf{R})$

is a complex Gaussian random vector with mean  $\mathbf{m}$  and covariance  $\mathbf{R}$ .

## 2. System Model and Channel Model

*2.1. System Model.* In this study, we describe three different multiuser hybrid precoding system models, the fully connected structures, the subconnected structures, and the dynamic subarray structures. Firstly, a fully connected multiuser downlink massive MIMO hybrid precoding system models as shown in Figure 1, where the BS is equipped with  $N_{\text{RF}}$  RF chains and  $N_T$  antennas. We assume that the BS simultaneously communicates with  $K$  mobile users. Each mobile user is equipped with one RF chain and  $N_R$  antennas, where  $K \leq N_{\text{RF}} \leq N_T$ . The BS antennas are tagged as  $1, 2, \dots, N_T$ , and let  $\mathcal{E} = \{1, 2, \dots, N_T\}$ . The baseband precoding matrix  $\mathbf{F}_{\text{BB}} = [\mathbf{F}_{\text{BB}}^1, \mathbf{F}_{\text{BB}}^2, \dots, \mathbf{F}_{\text{BB}}^K]$ , where  $\mathbf{F}_{\text{BB}}^k \in \mathbb{C}^{N_{\text{RF}} \times N_R}$  is the baseband precoding matrix for  $k$  th mobile user. The analog precoding weight is a  $N_T \times N_{\text{RF}}$  matrix  $\mathbf{F}_{\text{RF}} = [\mathbf{f}_{\text{RF}}^1, \mathbf{f}_{\text{RF}}^2, \dots, \mathbf{f}_{\text{RF}}^{N_{\text{RF}}}]$ , where  $\mathbf{f}_{\text{RF}}^k \in \mathbb{C}^{N_T \times 1}$  is the analog precoding vector for the  $k$  th RF chain.  $\mathbf{F}_{\text{RF}}$  and  $\mathbf{F}_{\text{BB}}$  are linked by the total power constrains, i.e.,  $\|\mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}}\|_F^2 = K$ . The sample transmitted signal can be represented as [21].

$$\mathbf{x} = \mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}} \mathbf{s}, \quad (1)$$

where  $\mathbf{s} = [s_1, s_2, \dots, s_K]^T$  is the  $K \times 1$  vector of transmitted symbols, which satisfy  $\mathbb{E}\{\mathbf{s}\mathbf{s}^H\} = (P/K)\mathbf{I}_K$ , and  $P$  is the average total transmitted power.

Generally, the received downlink signal of the mobile user  $k$  can be represented as [21].

$$\mathbf{y}_k = \mathbf{w}_k \mathbf{H}_k \mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}}^k s_k + \sum_{i \neq k} \mathbf{w}_k \mathbf{H}_k \mathbf{F}_{\text{RF}} \mathbf{f}_{\text{BB}}^i s_i + \mathbf{w}_k \mathbf{n}_k, \quad (2)$$

where  $\mathbf{H}_k \in \mathbb{C}^{N_R \times N_T}$  is the channel matrix from the BS to the  $k$  th mobile user. Let  $\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_N]^T$ .  $\mathbf{n}_k$  is an additive Gaussian white noise vector such that  $\mathbf{n}_k \sim \mathcal{CN}(\mathbf{0}, \delta^2 \mathbf{I})$ .  $\mathbf{w}_k \in \mathbb{C}^{1 \times N_R}$  denotes the digital vector at the  $k$  th mobile user.

Then, we describe the two structures shown in Figure 2 and define the form of their analog precoding matrices. The conventional subarray architecture in which each RF chain is connected to a specific subset of the antennas as shown in Figure 2(a), and each RF chain connected to the antenna is fixed in the order. Therefore, we can define the analog precoding matrix  $\mathbf{F}_{\text{RF}}$  as [9].

$$\mathbf{F}_{\text{RF}} = \begin{bmatrix} \mathbf{f}_{\text{RF}}^1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{f}_{\text{RF}}^2 & & \mathbf{0} \\ \vdots & & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{f}_{\text{RF}}^{N_{\text{RF}}} \end{bmatrix}_{N_{\text{RF}} M \times N_{\text{RF}}}, \quad (3)$$

where  $\mathbf{f}_{\text{RF}}^j = [f_j^{(j-1)M+1}, f_j^{(j-1)M+2}, \dots, f_j^{jM}]^T$  denotes the analog weighting vector for the  $j$  th RF chain,  $M = N_T/N_{\text{RF}}$ , and  $j = 1, 2, \dots, N_{\text{RF}}$ .

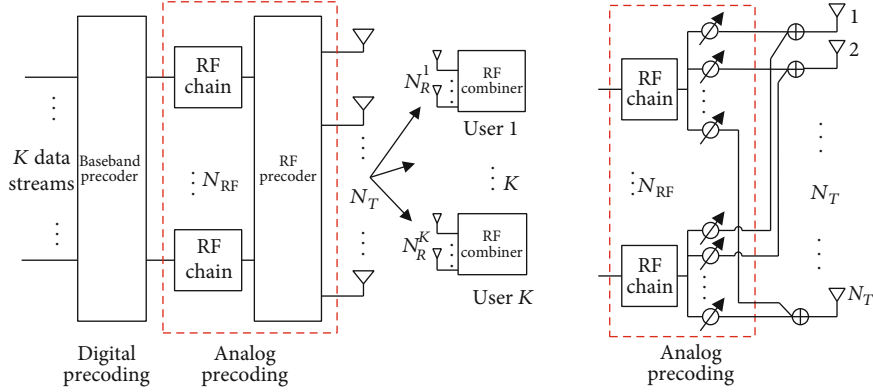


FIGURE 1: Fully connected architecture of the multiuser hybrid precoding in mmWave massive MIMO system.

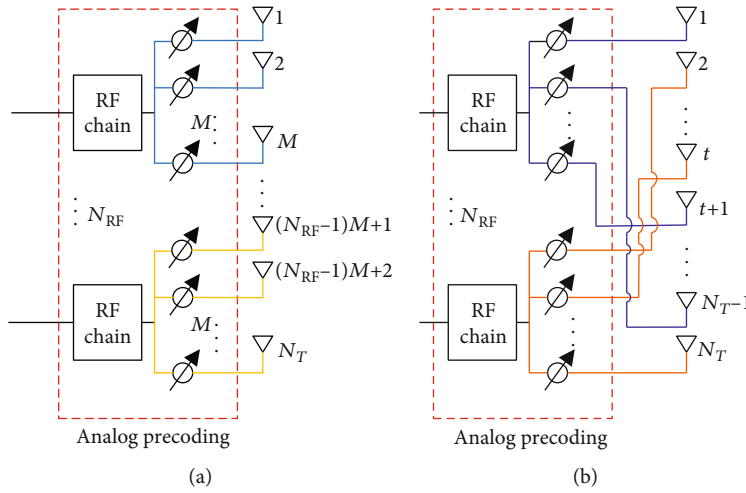


FIGURE 2: Two subconnected architectures of the multiuser hybrid precoding in mmWave massive MIMO system: (a) conventional subarray architecture, where each RF chain is connected to a fixed subset of the antennas; (b) dynamic subarray architecture, where each RF chain is connected to interlaced antennas.

The dynamic subarray architectures are shown in Figure 2(b), which each RF chain is connected to a dynamic subset of the antennas. The analog precoding matrix  $\mathbf{F}_{\text{RF}} = [\mathbf{f}_{\text{RF}}^{S_1}, \mathbf{f}_{\text{RF}}^{S_2}, \dots, \mathbf{f}_{\text{RF}}^{S_{N_{\text{RF}}}}]$  is derived with

$$\mathbf{f}_{\text{RF}}^{S_j}(t, j) = \begin{cases} f_j^t, & t \in S_j, \\ 0, & t \notin S_j, \end{cases} \quad (4)$$

where  $\mathbf{f}_{\text{RF}}^{S_j} \in \mathbb{C}^{N_T \times 1}$  is the analog precoding vector for the  $j$  th RF chain, and  $f_j^t$  denotes the element of  $t$  th antenna connected to the  $j$  th RF chain. Assume that  $f_j^t$  is a value of  $\mathbf{f}_{\text{RF}}^{S_j}(t, j)$ . The set  $S_j$  contains label of antenna connected to the  $j$  th RF chain, and let set  $S = \{S_1, S_2, \dots, S_{N_{\text{RF}}}\}$ ,  $\sum_{j=1}^{N_{\text{RF}}} |S_j| = N_T$ .

**2.2. mmWave Channel Model.** Millimeter wave channels with high free-space path loss and large dense antenna arrays are

unlikely to follow the rich scattering model at low frequencies. According to the characteristics of millimeter wave propagation, the dense array makes the antenna highly correlated, and the high path loss in high free-space leads to the limited space selectivity. Thus, the millimeter-wave channel is considered to have finite scattering [1]. Traditional MIMO channel modeling methods are not accurate in millimeter wave channel. The geometric channel model can embody the low rank and spatial correlation characteristics of mmWave communications [9]. Thus, we consider a geometric channel model with  $L_k$  scatterers for the  $k$  th user and assume that each scatterer contributes a single propagation path between the BS and the user [18, 21]. Under this model, the  $k$  th user's channel can be represented as

$$\mathbf{H}_k = \sqrt{\frac{N_T N_R}{L_k}} \sum_{l=1}^{L_k} \beta_{l,k} \mathbf{a}_{\text{R}}(\theta_{l,k}^{\text{R}}) \mathbf{a}_{\text{T}}^H(\theta_{l,k}^{\text{T}}), \quad (5)$$

where  $L_k$  is the number of scatterers and  $\beta_{l,k}$  is the complex

gain of the  $l$  th path between the BS and the  $k$  th user.  $\theta_{l,k}^R$  and  $\theta_{l,k}^T$ , respectively, represent the azimuth of arrival (AOA) and departure (AOD) of the  $l$  th path between the BS and the  $k$  th user, and the range of values for both  $\theta_{l,k}^R$  and  $\theta_{l,k}^T$  is  $[0, 2\pi]$ .  $\mathbf{a}_T(\theta_{l,k}^T)$  and  $\mathbf{a}_R(\theta_{l,k}^R)$  are the antenna array response vectors of the base station and  $k$  th user, separately. For a uniform linear array (ULA), the array response vector can be written as

$$\mathbf{a}_T(\theta_{l,k}^T) = \frac{1}{\sqrt{N_T}} \left[ 1, e^{j(2\pi d/\lambda) \sin(\theta_{l,k}^T)}, \dots, e^{j(N_T-1)(2\pi d/\lambda) \sin(\theta_{l,k}^T)} \right]^T, \quad (6)$$

where  $\lambda$  is the signal wavelength and  $d = \lambda/2$  denotes the spacing distance between two adjacent antennas. The array response vector  $\mathbf{a}_R(\theta_{l,k}^R)$  takes a similar fashion.

### 3. Problem Formulation

In this section, we propose the problem of multiuser hybrid precoding scheme based on dynamic subarray architectures. As shown in Figure 2(b), the antennas of each subarray are not adjacent to each other, and the optimal set  $S^*$  needs to get at first. Then, design the hybrid precoding matrix  $\mathbf{F}_{\text{RF}}$  and  $\mathbf{F}_{\text{BB}}$  for the dynamic subarray architectures at the BS. In this paper, we assume that the channel state information (CSI) is perfect. The SINR of the  $k$  th user is written as

$$\text{SINR}_k = \frac{P \left\| \tilde{\mathbf{H}}_k \mathbf{F}_{\text{RF}} \mathbf{f}_{\text{BB}}^k \right\|_F^2}{K\sigma_k^2 + P \sum_{i \neq k} \left\| \tilde{\mathbf{H}}_k \mathbf{F}_{\text{RF}} \mathbf{f}_{\text{BB}}^i \right\|_F^2}, \quad (7)$$

where  $\tilde{\mathbf{H}}_k = \mathbf{w}_k \mathbf{H}_k$ , and  $\mathbf{w}_k = \mathbf{U}_k^H$  is obtained by SVD of  $\mathbf{H}_k$ , which is  $\mathbf{H}_k = \mathbf{U}_k \mathbf{\Lambda}_k \mathbf{V}_k^H$ .

The achievable rate of the  $k$  th user can be written as

$$R_k = \log_2(1 + \text{SINR}_k). \quad (8)$$

We aim to jointly design the  $S^*$ ,  $\mathbf{F}_{\text{RF}}^*$ , and  $\mathbf{F}_{\text{BB}}^*$  to maximize the achievable sum-rate of dynamic subarray architectures, which can be formulated as

$$\{S^*, \mathbf{F}_{\text{RF}}^*, \mathbf{F}_{\text{BB}}^*\} = \arg \max_{\mathbf{F}_{\text{RF}}, \mathbf{F}_{\text{BB}}, S} \sum_{k=1}^K \log_2(1 + \text{SINR}_k), \quad (9)$$

$$s.t. \|\mathbf{f}_{\text{RF}}(t, j)\|_F^2 = 1, t = 1, 2, \dots, N_T,$$

$$\|\mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}}\|_F^2 = K.$$

The optimization problem of (9) is a nonconvex problem, and it is almost impossible to solve the global optimal solution directly. We decompose problem (9) into three subproblems. Firstly, we design the initial precoding matrix for each user to get the optimal set  $S^*$ . Secondly, we design the optimal analog precoding matrix  $\mathbf{F}_{\text{RF}}^*$  according to  $S^*$ , and finally, we design the optimal digital precoding matrix  $\mathbf{F}_{\text{BB}}^*$  through the equivalent channel matrix  $\tilde{\mathbf{H}}_k = \mathbf{H}_k \mathbf{F}_{\text{RF}}^*$ . The detail of the scheme will be explained in the next section.

## 4. Hybrid Precoding Strategy

In this section, we introduce a dynamically subarray-connected hybrid precoding scheme for multiuser mmWave massive MIMO systems, including analog precoding initialization and user selection, dynamic antenna partitioning, and the hybrid precoding optimize for the dynamic subarray structures.

**4.1. Initialization Analog Precoding and User Selection.** In the multiuser mmWave massive MIMO system, the mmWave channel has a finite number of strong beam directions; thus, the best beams of the users lie on their own scattering paths. When the base station serves all users in the same time slot, the performance of the system will be degraded with the increase of interuser interference. Increasing the number of served users at the same time slot will result in performance degradation [22, 23]. In this paper, we select a group of users with minimum interuser interference and maximum objective channel gains. Generally, maximizing the SINR is the criterion for user selection [15].

Firstly, we define a set of all candidate users  $\mathcal{N} = \{1, 2, \dots, N\}$  and an empty set  $\mathcal{U}$ . The initial analog precoding vector  $\mathbf{f}_n^0$  of  $n$  th user ( $n = 1, 2, \dots, N$ ) can be obtained by solving

$$\begin{aligned} \mathbf{f}_n^0 &= \arg \max \|\mathbf{H}_n \mathbf{f}_n\|_F^2, \\ s.t. |\mathbf{f}_n(t, n)| &= 1, \forall n, t. \end{aligned} \quad (10)$$

The SVD of channel  $\mathbf{H}_n$  can be expressed as  $\mathbf{H}_n = \mathbf{U}_n \mathbf{\Lambda}_n \mathbf{V}_n^H$ . The unconstrained analog precoding vector for user  $n$  can be written as  $\mathbf{f}_n^{\text{opt}} = \mathbf{v}_n^1$ , where  $\mathbf{v}_n^1$  is the first columns of  $\mathbf{V}_n$ .

The optimization problem (10) can be rewritten as

$$\begin{aligned} \mathbf{f}_n^0 &= \arg \min \|\mathbf{f}_n^{\text{opt}} - \mathbf{f}_n\|_F^2, \\ s.t. |\mathbf{f}_{\text{RF}}(t, n)| &= 1, \forall n, t. \end{aligned} \quad (11)$$

The solutions of similar problems with (11) are given in [9]. Therefore, the solution of (11) can be obtained directly.

$$\mathbf{f}_n^0 = e^{j\angle \mathbf{f}_n^{\text{opt}}} = e^{j\angle \mathbf{v}_n^1}. \quad (12)$$

Then, the process of multiuser selection is described as follows. Using the initial analog precoding vectors, the SINR of the user  $n$  written as

$$\text{SINR}_n = \frac{P \|\mathbf{w}_n \mathbf{H}_n \mathbf{f}_n^0\|_F^2}{K\sigma_n^2 + P \sum_{i \neq n} \|\mathbf{w}_n \mathbf{H}_n \mathbf{f}_i^0\|_F^2}. \quad (13)$$

The first user  $\mathcal{U}(1)$  selected by maximum channel gain  $\|\mathbf{w}_n \mathbf{H}_n\|_F^2$  from set  $\mathcal{N}$ , and then, we update both set  $\mathcal{N}$  and  $\mathcal{U}$ . The remaining users are selected from all candidate users set  $\mathcal{N}$  according to the maximum SINR criterion, which can be described as follows:

$$\mathcal{U}(k) = \arg \max_{\mathcal{N}, \mathcal{U}} \frac{P \left\| \mathbf{w}_{\mathcal{N}(n)} \mathbf{H}_{\mathcal{N}(n)} \mathbf{f}_{\mathcal{N}(n)}^0 \right\|_F^2}{K \sigma_{\mathcal{N}(n)}^2 + P \sum_{i=1}^{k-1} \left\| \mathbf{w}_{\mathcal{N}(n)} \mathbf{H}_{\mathcal{N}(n)} \mathbf{f}_{\mathcal{U}(i)}^0 \right\|_F^2},$$

$$k \leq K \leq N_{\text{RF}}, i = 1, 2, \dots, k-1, \quad (14)$$

where  $\mathbf{H}_{\mathcal{N}(n)}$  is the mmWave channel of the  $n$  th user in set  $\mathcal{N}$ ,  $\mathbf{f}_{\mathcal{N}(n)}^0$  and  $\mathbf{f}_{\mathcal{U}(i)}^0$  denote the initial analog precoding vector of the  $n$  th user in set  $\mathcal{N}$  and the initial analog precoding vector of the  $i$  th user in set  $\mathcal{U}$ , respectively. The user  $\mathcal{U}(k)$  with maximum SINR value is removed from  $\mathcal{N}$  and added to  $\mathcal{U}$ . The update strategy is as follows:

$$\mathcal{U} \leftarrow \mathcal{U} \cup \mathcal{U}(k), \mathcal{N} \leftarrow \mathcal{N} \setminus \mathcal{U}(k). \quad (15)$$

The process of multiuser selection is summarized in Algorithm 1.

**4.2. Dynamic Subarray for Multiuser mmWave System.** The dynamic subarray hybrid architecture is a more efficient structure, in which the antenna elements are dynamically partitioned to each RF chain based on the long-term channel information [16, 18], and it addresses the tradeoff between the achievable spectral efficiency and energy efficiency. The existing antenna partition algorithm for dynamic subarray hybrid architecture is proposed in [20], but it has poor user fairness because of insufficient conditions. It uses the maximum increment of the SINR for each selected user, the conditions to ensure that each RF chain can be allocated one antenna at least have not been set, resulting in some RF chains not being partition of any antenna under special circumstances, such as some users can obtain the maximum SINR increment in all loop. In this subsection, based on the dynamic subarray hybrid structure proposed in Section 2, we design the new antenna partition algorithm to obtain higher spectral efficiency and guarantee user fairness.

The objective of this subsection is to address the optimal  $S_k^*$  of the user  $k$ . We assume that the number of chain  $N_{\text{RF}}$  is equal to the number of user  $K$ , and the BS transmits data to every user via only one stream. As shown in Figure 2(b), there are two constraints as

$$\sum_{j=1}^{N_{\text{RF}}} |\mathbf{f}_{\text{RF}}(t, j)| = 1, \quad (16)$$

$$\sum_{t=1}^{N_T} |\mathbf{f}_{\text{RF}}(t, j)| \geq N, \quad (17)$$

where  $N = (N_T - N_T \bmod N_{\text{RF}}) / N_{\text{RF}}$ , and the constrain of (16) ensures that each antenna is only connected to one RF chain, and the constrain of (17) ensures that each RF chain is connected to  $N$  antennas at least.

Then, in order to ensure that each user gets maximize channel gain and ensure fairness among users, we neglect the resulting interference among users in this stage, and the user interference elimination problem is solved in the next

subsection. Directly solving three unknown matrix variables  $S^*$ ,  $\mathbf{F}_{\text{RF}}^*$ , and  $\mathbf{F}_{\text{BB}}^*$  will lead to high computational complexity. Therefore, we use the initial analog precoding vectors as significant information for partitioning antenna elements and let the dynamic subarray of each user start out as an empty set. The antenna-partitioning problem is described as

$$t^* = \arg \max_{t \in \mathcal{E}} \left\| \mathbf{w}_k \mathbf{H}_k \mathbf{f}_{S_k \cup t}^0 \right\|_F^2, \quad (18)$$

where  $\mathbf{f}_{S_k}^0$  is a  $N_T \times 1$  vector, which is obtained by  $\mathbf{f}_k^0$  according to (4), and  $S_k \cup t$  represents the antenna  $t$  being added to the subarray  $S_k$  of  $k$  th user. Then, set  $\mathcal{E}$  and subarray  $S_k$  are updated as follows:

$$S_k \leftarrow S_k \cup t^*, \mathcal{E} \leftarrow \mathcal{E} \setminus t^*, \quad (19)$$

Within each inner loop of Algorithm 2, due to the constrain in (17), (18), and (19), will be carried out for all users once, and this inner loop will be carried out for  $N$  times. After the  $N$  loop, if  $\mathcal{E} \neq \emptyset$ , it means that there are still remaining antennas unallocated, and we allocate the remaining antennas to the user with the smallest channel gain. The details of the process are described in Algorithm 2.

For the simplicity of the process analysis, we assume that the number of chain  $N_{\text{RF}}$  is equal to the number of user  $K$ , and the BS transmits data to every user via only one stream. In fact, by changing the constraints (16) and (17), this new antenna partition algorithm can easily be extended to user multistream transmission.

**4.3. Hybrid Precoding Design.** In this subsection, we will design the hybrid precoding for dynamic subarray hybrid architecture that obtained in the previous subsection. For the dynamic subarray hybrid architecture, the beam shape and width are changed after the antenna partitioning of this architecture, and the initial analog precoding vector is invalid. Therefore, the analog precoding needs to be redesigned.

Utilize the subarray  $S_1, S_2, \dots, S_K$ , the channel matrix for each user can be updated as

$$\mathbf{H}_k^{S_k}(:, t) = \begin{cases} \mathbf{H}_k(:, t), & t \in S_k, \\ 0, & t \notin S_k, \end{cases} \quad (20)$$

$$\bar{\mathbf{H}}_k = \mathbf{H}_k^{S_k},$$

where  $\bar{\mathbf{H}}_k$  is a  $N_R \times N_T$  channel matrix for  $k$  th user, and define the effective channel  $\mathbf{e}_k = \mathbf{w}_k \bar{\mathbf{H}}_k$  for  $k$  th user, where  $\mathbf{w}_k$  can obtain by SVD of  $\bar{\mathbf{H}}_k$ .

In order to obtain the large array gain, we consider the phase-only control at the RF domain [24] and perform the analog precoding for each user according to

$$\mathbf{f}_k^{S_k} = e^{j(-\varphi_k)}, \quad (21)$$

where  $\varphi_k$  is the phase of the  $k$  th element in the vector  $\mathbf{e}_k$  and  $-\varphi_k$  is the conjugate transpose of  $\varphi_k$ .

```

1:Input:  $\mathbf{H}, \mathcal{N}, \mathcal{U}, K$ 
2:Initialize:  $\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_N]^T$ ,  $\mathcal{N} = \{1, 2, \dots, N\}$ ,  $\mathcal{U} = \emptyset$ 
3:for  $\mathbf{n} = 1:N$  do
4: Calculate the SVD of  $\mathbf{H}_n$  to obtain  $\mathbf{W}_n$  and  $\mathbf{V}_n$ ,  $\mathbf{w}_n = \mathbf{W}_n^H(1, :)$ ,  $\mathbf{f}_n^0 = e^{j\angle \mathbf{v}_n^1}$ , where  $\mathbf{v}_n^1$  is the first columns of  $\mathbf{V}_n$ .
5:  $\nabla_n = \|\mathbf{w}_n \mathbf{H}_n\|_F^2$ 
6:end for
7:  $\mathcal{U}(1) = \arg \max \nabla_n$ 
8:  $\mathcal{U} \leftarrow \mathcal{U} \cup \mathcal{U}(1)$ ,  $\mathcal{N} \leftarrow \mathcal{N} \setminus \mathcal{U}(1)$ 
9:for  $\mathbf{k} = 2:K$  do
10:  $\mathcal{U}(k) = \arg \max_{\mathcal{N}, \mathcal{U}} (P \|\mathbf{w}_{\mathcal{N}(n)} \mathbf{H}_{\mathcal{N}(n)} \mathbf{f}_{\mathcal{N}(n)}^0\|_F^2 / K \sigma_{\mathcal{N}(n)}^2 + P \sum_{i=1}^{k-1} \|\mathbf{w}_{\mathcal{N}(n)} \mathbf{H}_{\mathcal{N}(n)} \mathbf{f}_{\mathcal{U}(i)}^0\|_F^2)$ 
11:  $\mathcal{U} \leftarrow \mathcal{U} \cup \mathcal{U}(k)$ ,  $\mathcal{N} \leftarrow \mathcal{N} \setminus \mathcal{U}(k)$ 
12:end for
13:output:  $\mathcal{U}$ 

```

ALGORITHM 1: Multiuser selection based on SINR.

```

1:Input:  $\mathbf{H}, \mathcal{U}, K, S = \{S_1, S_2, \dots, S_k, \dots, S_K\}$ ,  $\mathbf{f}_n^0$ 
2:Initialize:  $\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_N]^T$ ,  $\mathcal{U}$  and  $\mathbf{f}_n^0$  are obtain by Algorithm 1,  $K = |\mathcal{U}|$ ,
 $S_1 = S_2 = \dots = S_K = \emptyset$ ,  $N = (N_T - N_T \bmod N_{\text{RF}}) / N_{\text{RF}}$ 
3:for loop = 1:N do
4:   for  $\mathbf{k} = 1:K$  do
5:      $t^* = \arg \max_{t \in \mathcal{E}} \|\mathbf{w}_k \mathbf{H}_k \mathbf{f}_{S_k \cup t}^0\|_F^2$ 
6:      $S_k \leftarrow S_k \cup t^*$ ,  $\mathcal{E} \leftarrow \mathcal{E} \setminus t^*$ 
7:      $\Delta_k = \|\mathbf{w}_k \mathbf{H}_k \mathbf{f}_{S_k}^0\|_F^2$ 
8:   end for
9:end for
10:while  $\mathcal{E} \neq \emptyset$  do
11:    $k^* = \arg \min_{k=1, \dots, K} \Delta_k$ 
12:    $t^* = \arg \max_{t \in \mathcal{E}} \|\mathbf{w}_{k^*} \mathbf{H}_{k^*} \mathbf{f}_{S_{k^*} \cup t}^0\|_F^2$ 
13:    $S_{k^*} \leftarrow S_{k^*} \cup t^*$ ,  $\mathcal{E} \leftarrow \mathcal{E} \setminus t^*$ 
14:    $\Delta_{k^*} = \|\mathbf{w}_{k^*} \mathbf{H}_{k^*} \mathbf{f}_{S_{k^*}}^0\|_F^2$ 
15:end while
16:output:  $S_1, S_2, \dots, S_K$ 

```

ALGORITHM 2: Multiuser dynamic subarray partitioning.

In the stage of digital baseband precoding, we focused on the elimination of interuser interference. Some classical digital precoding algorithms are used to eliminate the interuser interference, such as ZF and BD. However, in multiuser systems with a low signal-to-noise ratio (SNR), ZF or BD precoding performs poorly [12]. Thus, we use the minimum mean-squared error criteria to design digital baseband precoding. According to the received signal model in (2), the estimated symbol of  $k$  th user is given by

$$\hat{\mathbf{s}}_k = \mathbf{w}_k \bar{\mathbf{H}}_k \mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}}^k \mathbf{s}_k + \mathbf{w}_k \bar{\mathbf{H}}_k \sum_{i \neq k} \mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}}^i \mathbf{s}_i + \mathbf{w}_k \mathbf{n}_k. \quad (22)$$

The estimated symbols for all users given (22) can be stacked into one vector.

$$\hat{\mathbf{s}} = \mathbf{W} \bar{\mathbf{H}} \mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}} \mathbf{s} + \mathbf{W} \mathbf{n}, \quad (23)$$

where  $\bar{\mathbf{H}} = [\bar{\mathbf{H}}_1^T, \bar{\mathbf{H}}_2^T, \dots, \bar{\mathbf{H}}_K^T]^T$ ,  $\mathbf{W} = \text{blkdiag}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_K)$ ,  $\mathbf{F}_{\text{RF}} = [\mathbf{f}_1^S, \mathbf{f}_2^S, \dots, \mathbf{f}_K^S]$ ,  $\mathbf{F}_{\text{BB}} = \text{blkdiag}(\mathbf{F}_{\text{BB}}^1, \mathbf{F}_{\text{BB}}^2, \dots, \mathbf{F}_{\text{BB}}^K)$ , and  $\mathbf{n} = [\mathbf{n}_1^T, \mathbf{n}_2^T, \dots, \mathbf{n}_K^T]^T$ . The aim is to get estimated symbols  $\hat{\mathbf{s}}$  close to  $\mathbf{s}$ . The mean-squared error cost function can be expanded as

$$\begin{aligned} \mathbb{E} \left[ \|\mathbf{s} - \sqrt{\gamma} \mathbf{s} \wedge\|^2 \right] &= \mathbb{E} \left[ \|\mathbf{s} - \sqrt{\gamma} \mathbf{W} \bar{\mathbf{H}} \mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}} \mathbf{s} - \sqrt{\gamma} \mathbf{W} \mathbf{n}\|^2 \right] \\ &= \|\mathbf{I}_K - \mathbf{W} \bar{\mathbf{H}} \mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}}\|_F^2 + K \gamma \sigma^2, \end{aligned} \quad (24)$$

where  $\gamma$  is the scaling factor.

Then, the problem of minimum mean-squared error can be expressed as

$$\underset{\mathbf{F}_{\text{RF}}, \mathbf{F}_{\text{BB}}, \gamma}{\text{minimize}} \quad \|\mathbf{I}_K - \mathbf{W} \bar{\mathbf{H}} \mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}}\|_F^2 + K \gamma \sigma^2 \text{ s.t. } \quad \text{Tr} \{ \mathbf{F}_{\text{RF}}^H \mathbf{F}_{\text{RF}} \mathbf{F}_{\text{BB}} \mathbf{F}_{\text{BB}}^H \} \leq \gamma P. \quad (25)$$

The analog precoding matrix  $\mathbf{F}_{\text{RF}}$  can be obtained by (21); thus, the problem (25) will be transformed into a convex optimization problem. The closed-form optimal solution of the similar problem is given in [25]. Its optimal solution is given by

$$\mathbf{F}_{\text{BB}} = \left( \mathbf{T}^H \mathbf{T} + \frac{K\delta^2}{P} \mathbf{I}_K \right)^\dagger \mathbf{T}^H, \quad (26)$$

where  $\mathbf{T} = \mathbf{W}\bar{\mathbf{H}}\mathbf{F}_{\text{RF}}$ , the optimal scaling factor  $\gamma = \|\mathbf{F}_{\text{RF}}\mathbf{F}_{\text{BB}}\|_F^2/P$ , normalizes  $\mathbf{F}_{\text{BB}} = \sqrt{1/\gamma}\mathbf{F}_{\text{BB}}$ .

In the process of MMSE-based digital precoding computation, the main computational complexity comes from formula (26). In (26), the complexity of matrix multiplication is  $\mathcal{O}(K^2N_RN_TN_{\text{RF}})$ , and the complexity of pseudo-inverse is  $\mathcal{O}(K^3)$ . Therefore, the total complexity is  $\mathcal{O}(K^2N_RN_TN_{\text{RF}} + K^3)$ . Since the baseband digital precoding design in hybrid precoding is carried out in low dimensions, the increase in the number of base station transmitting antennas will not bring huge computational complexity. In practical application, this method may bring some challenges when the number of users increases rapidly. In general, the MMSE-based baseband digital precoding is an effective method.

## 5. Simulation Results

In this section, simulation results are given to illustrate the performance advantages of the proposed new antenna-partitioning algorithm and hybrid analog/digital precoding scheme in spectrum efficiency, energy efficiency, and user fairness. We compare the performance of three hybrid precoding schemes with different structures, namely, conventional subconnected structure, existing dynamic subarray-connected structure [20], and fully connected structure. Firstly, we consider a mmWave channel and dynamic subarray architecture described in Section 2. The carrier frequency is set as 28 GHz [1]. The number of scatterers for each user is  $L_k = 3$ . The transmit antenna arrays are ULAs with antenna spacing  $d = \lambda/2$ . We assume that both the AoAs and AoDs are following the uniform distribution within  $[-\pi, \pi]$ . Finally, SNR is defined  $P/\sigma^2$ .

### 5.1. Performance Comparisons of Spectrum Efficiency.

Figure 3 shows the spectral efficiency comparison in different architectures, where the BS is equipped with 64 antennas and 2 RF chains serving two users, and each user is equipped with two antennas. We observe from Figure 3 that the performance of the proposed new antenna-partitioning-based hybrid precoding scheme outperforms the existing dynamic subarray precoding scheme [20] in the whole simulated SNR range. For instance, when SNR = 0 dB, our proposed scheme can achieve 86.1% of the spectral efficiency obtained by fully connected architecture, and compared with the conventional subarray architecture, the spectral efficiency of the proposed scheme is improved by 11.1%. Moreover, compared with the existing dynamic subarray architecture scheme, the spectral efficiency of the proposed scheme is also

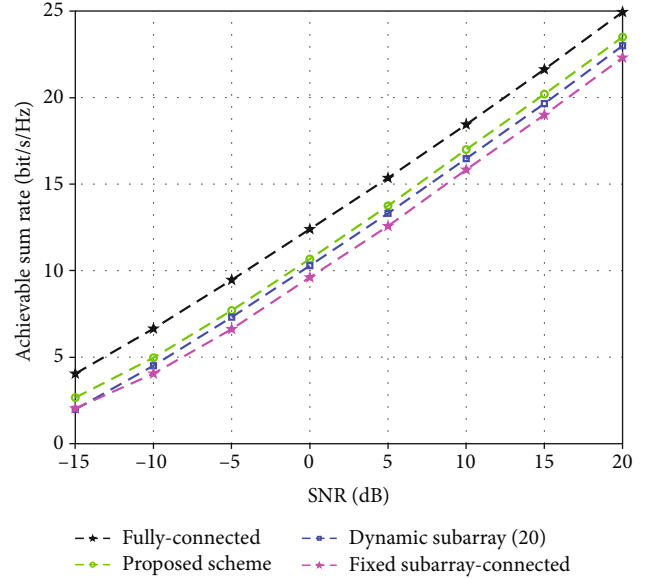


FIGURE 3: Spectral efficiency comparison for the mmWave MIMO system with  $N_T = 64$ ,  $N_R = 2$ ,  $K = 2$ , and  $N_{\text{RF}} = 2$  RF chains.

improved; especially at low SNR, the spectral efficiency is improved by 4.7% when SNR = 0 dB.

In Figure 4, the BS is equipped with 128 antennas and 2 RF chains serving two users. Similar to the trend in Figure 3, the spectrum efficiency of all architectures is improved, and our proposed scheme is still superior to the existing dynamic subarray architecture scheme. As can be seen from Figures 3 and 4, our proposed scheme is closer to the performance achieved by fully connected architecture, and the advantages are obvious at low SNR.

Figure 5 provides a spectrum efficiency comparison against the number of BS transmitting antennas. The BS is equipped with different antennas and two RF chains serving two users. As observed from Figure 5, the spectral efficiency increases with the increase of the number of antennas, but the rate of increase slows down when the SNR = -10 dB and the SNR = 0 dB. More importantly, the existing dynamic subarray schemes [20] do not show significant advantages when the BS equipped small transmitting antennas, such as 16 and 32. Moreover, our proposed scheme in this paper has obvious advantages in the whole simulation range.

### 5.2. Performance Comparisons of Energy Efficiency.

The energy consumption model was proposed in [26], and based on this model, the energy efficiency  $\eta$  can be defined as

$$\eta = \frac{R_{\text{sum}}}{P_{\text{total}}} = \frac{\sum_{k=1}^K \log_2(1 + \text{SINR}_k)}{P_t + N_{\text{RF}}P_{\text{RF}} + N_{\text{PS}}P_{\text{PS}}} \quad (\text{bps/Hz/W}), \quad (27)$$

where  $P_{\text{total}}$  is the total energy consumption,  $P_t$  is the transmission power,  $P_{\text{RF}}$  is the energy consumed by RF chain, and  $P_{\text{PS}}$  is the energy consumed by phase shifter.  $N_{\text{RF}}$  and  $N_{\text{PS}}$  are the number of required RF chains and phase shifters, respectively.

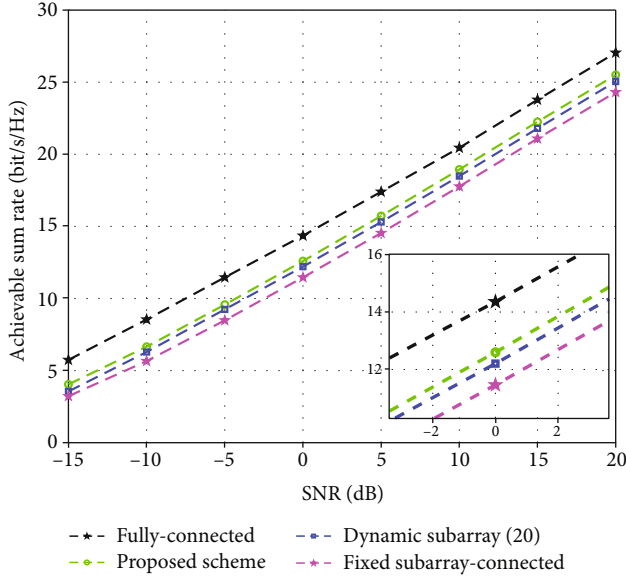


FIGURE 4: Spectral efficiency comparison for the mmWave MIMO system with  $N_T = 128$ ,  $N_R = 2$ ,  $K = 2$ , and  $N_{RF} = 2$  RF chains.

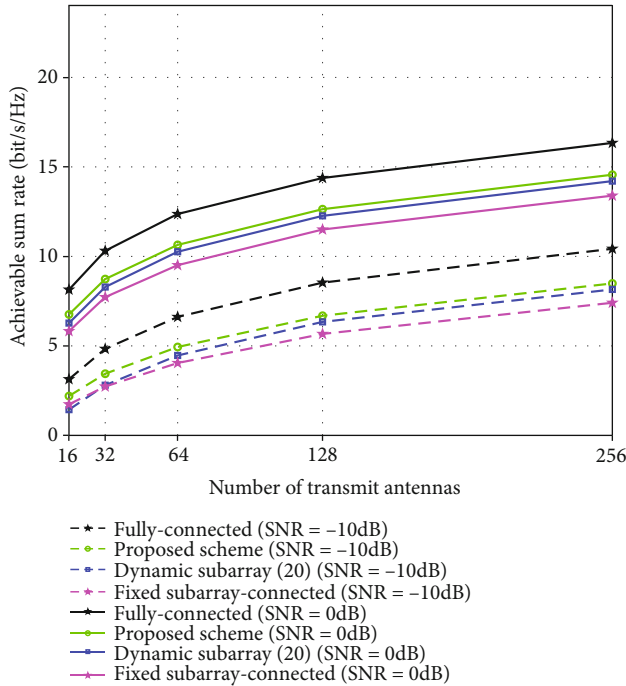


FIGURE 5: Spectral efficiency comparison for the mmWave MIMO system with different numbers of transmitting antennas.

In this paper, we use  $P_{RF} = 250\text{mW}$  [26] and  $P_{PS} = 1\text{mW}$  [27] in simulation,  $P_t$  is constrained to  $\|\mathbf{F}_{RF}\mathbf{F}_{BB}\|_F^2 = K$ . The energy efficiency comparison against the number of RF chains  $N_{RF}$  is shown in Figure 6, where  $N_T = 128$ ,  $N_R = 2$ , and  $\text{SNR} = 0\text{ dB}$ . We can see that with

the increase of the number of RF chains, the energy efficiency gradually decreases, which is consistent with the theoretical analysis. Moreover, it is clear that the dynamic subarray-connected architectures have higher energy effi-

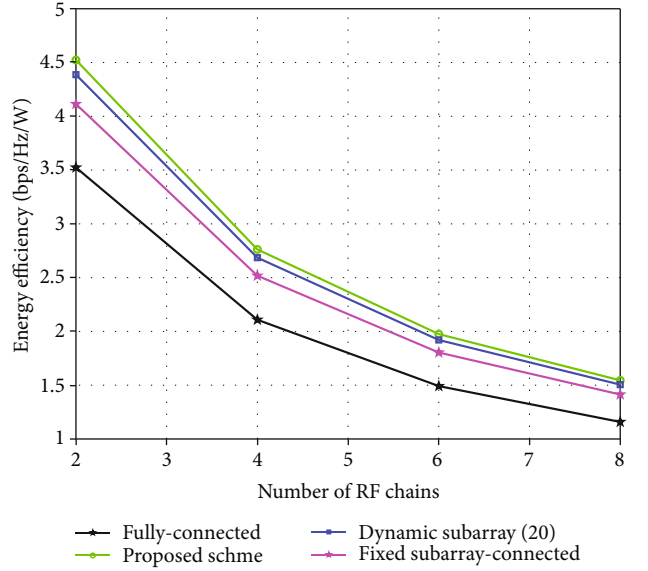


FIGURE 6: Energy efficiency comparison against the numbers of RF chains, where  $N_T = 128$ ,  $\text{SNR} = 0\text{ dB}$ ,  $N_R = 2$ , and  $K = 2$ .

ciency than the fully connected architectures. In addition, compared with the existing dynamic subarray schemes, the energy efficiency of the proposed scheme is improved by 3.5% on average.

Figure 7 shows the energy efficiency comparison against the number of transmitting antennas of BS, where  $N_T = 16, 32, 64, 128$ , and  $256$ ,  $\text{SNR} = 0\text{ dB}$ ,  $N_{RF} = 2$ , and  $K = 2$ . As the number of base station antennas increases, the energy efficiency first increases and then decreases, especially for fully connected architectures, which are faced with rapid performance degradation. In addition, the proposed schemes achieve better energy efficiency than existing dynamic subarray schemes and can maintain a high and stable energy efficiency between 64 and 256 antennas.

**5.3. User Fairness Analysis.** As an indicator of the allocation of shared resources, fairness is widely used in various fields, for example, data collection in cognitive radio networks [28, 29] and data transfer in wireless communication systems. In a multiuser communication system, it is important to ensure that each user has access to roughly the same data transfer rate. In Section 4, we introduce user selection to provide better service for a small number of users, where it is important to ensure fairness among users. In order to investigate the fairness of each user, we introduce the Jain fairness index [30]. The Jain fairness index is a very effective measure of distributive fairness. Jain fairness index defines fairness among multiple users as

$$J(\mathbf{R}) = \frac{\left(\sum_{k=1}^K R_k\right)^2}{K \sum_{k=1}^K R_k^2}, \quad (28)$$

where  $J(\mathbf{R})$  is the coefficient of fairness between  $K$  users,  $J(\mathbf{R}) \in [1/K, 1]$ . According to the definition of the Jain

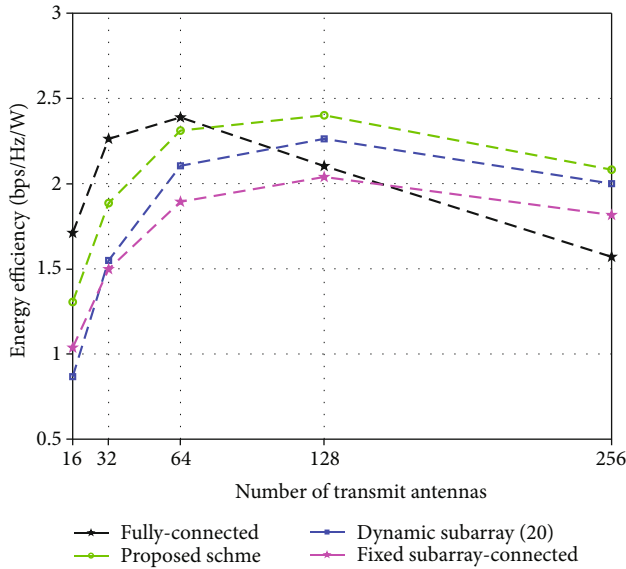


FIGURE 7: Energy efficiency comparison against the numbers of transmit antennas, where SNR = 0 dB,  $N_R = 2$ , and  $K = 2$ .

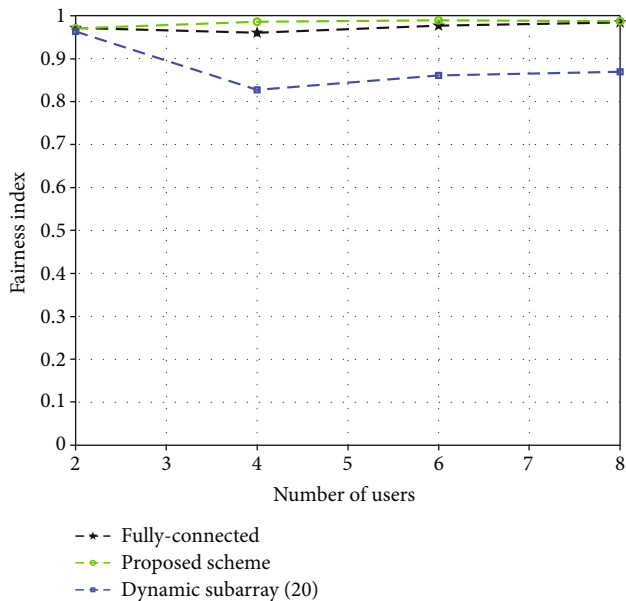


FIGURE 8: Fairness index comparison against the numbers of users  $K$ , where  $N_T = 128$ ,  $N_R = 2$ , and SNR = -10 dB.

fairness index, if the value of  $J(\mathbf{R})$  tends to  $1/K$ , the fairness is worse; on the other hand, if the value of  $J(\mathbf{R})$  tends to 1, the fairness is better. In Figure 8, we compared the fairness of different schemes with different numbers of users  $K$ , where  $N_T = 128$ , SNR = -10 dB, and  $N_R = 2$ . We observe that our proposed scheme can make the fairness index close to 1 under the different number of users. However, the fairness index of the existing dynamic subarray schemes [20] decreases with the increase of the number of users.

## 6. Conclusion

In this paper, we propose a new antenna-partitioning algorithm and a multiuser hybrid precoding scheme for the dynamic subarray of multiuser mmWave massive MIMO systems. The idea is to ensure that each user has the same opportunity to obtain the optimal antenna according to the maximum channel gain of the selected user. Then, we design the hybrid precoding with high performance for dynamic subarray architecture. Simulation results show that the proposed scheme in this paper can make the spectrum efficiency closer to the fully connected architectures and the energy efficiency closer to the traditional subconnected architectures. Moreover, the scheme proposed in this paper, especially at low SNR, is more advanced than the existing dynamic subarray schemes. In terms of user fairness, our scheme can almost make every user reach the same achievable rate. For future work, we are interested to evaluate performance in the dynamic subarray architecture with uniform planar arrays (UPAs) and under imperfect channel state information.

## 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 no conflicts of interest regarding the publication of 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] T. S. Rappaport, S. Sun, R. Mayzus et al., "Millimeter wave mobile communications for 5G cellular: it will work!," *IEEE Access*, vol. 1, pp. 335–349, 2013.
- [2] S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter-wave cellular wireless networks: potentials and challenges," *Proceedings of the IEEE*, vol. 102, no. 3, pp. 366–385, 2014.
- [3] O. E. Ayach, S. Rajagopal, S. Abu-Surra, Z. Pi, and R. W. Heath, "Spatially sparse precoding in millimeter wave MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 13, no. 3, pp. 1499–1513, 2014.
- [4] J. Zhao, F. Gao, W. Jia, S. Zhang, S. Jin, and H. Lin, "Angle domain hybrid precoding and channel tracking for millimeter wave massive MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 16, no. 10, pp. 6868–6880, 2017.
- [5] R. Méndez-Rial, C. Rusu, N. González-Prelcic, and R. W. Heath, "Dictionary-free hybrid precoders and combiners for mmWave MIMO systems," in *2015 IEEE 16th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 151–155, Stockholm, Sweden, 2015.
- [6] X. Yu, J. Shen, J. Zhang, and K. B. Letaief, "Alternating minimization algorithms for hybrid precoding in millimeter wave



- MIMO systems,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 485–500, 2016.
- [7] W. Ni, X. Dong, and W. Lu, “Near-optimal hybrid processing for massive MIMO systems via matrix decomposition,” *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 3922–3933, 2017.
- [8] Y. Wang and W. Zou, “Low complexity hybrid precoder design for millimeter wave MIMO systems,” *IEEE Communications Letters*, vol. 23, no. 7, pp. 1259–1262, 2019.
- [9] X. Gao, L. Dai, S. Han, I. Chih-Lin, and R. W. Heath, “Energy-efficient hybrid analog and digital precoding for mmWave MIMO systems with large antenna arrays,” *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 4, pp. 998–1009, 2016.
- [10] Y. Liu, Q. Feng, Q. Wu, Y. Zhang, M. Jin, and T. Qiu, “Energy-efficient hybrid precoding with low complexity for mmWave massive MIMO systems,” *IEEE Access*, vol. 7, pp. 95021–95032, 2019.
- [11] Y. Chen, D. Chen, T. Jiang, and L. Hanzo, “Millimeter-wave massive MIMO systems relying on generalized sub-array-connected hybrid precoding,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 8940–8950, 2019.
- [12] D. H. N. Nguyen, L. B. Le, T. Le-Ngoc, and R. W. Heath, “Hybrid MMSE precoding and combining designs for mmWave multiuser systems,” *IEEE Access*, vol. 5, pp. 19167–19181, 2017.
- [13] X. Gao, L. Dai, Y. Sun, S. Han, and I. Chih-Lin, “Machine learning inspired energy-efficient hybrid precoding for mmWave massive MIMO systems,” in *2017 IEEE International Conference on Communications (ICC)*, pp. 1–6, Paris, 2017.
- [14] J. Chen, “Energy-efficient hybrid precoding design for millimeter-wave massive MIMO systems via coordinate update algorithms,” *IEEE Access*, vol. 6, pp. 17361–17367, 2018.
- [15] D. Zhang, Y. Wang, X. Li, and W. Xiang, “Hybrid beamforming for downlink multiuser millimetre wave MIMO-OFDM systems,” *IET Communications*, vol. 13, no. 11, pp. 1557–1564, 2019.
- [16] X. Zhu, Z. Wang, L. Dai, and Q. Wang, “Adaptive hybrid precoding for multiuser massive MIMO,” *IEEE Communications Letters*, vol. 20, no. 4, pp. 776–779, 2016.
- [17] S. Park, A. Alkhateeb, and R. W. Heath, “Dynamic subarrays for hybrid precoding in wideband mmWave MIMO systems,” *IEEE Transactions on Wireless Communications*, vol. 16, no. 5, pp. 2907–2920, 2017.
- [18] L. Zhang, L. Gui, Q. Qin, and Y. Tu, “Adaptively-connected structure for hybrid precoding in multi-user massive MIMO systems,” in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–7, Bologna, Italy, 2018.
- [19] X. Jing, L. Li, H. Liu, and S. Li, “Dynamically connected hybrid precoding scheme for millimeter-wave massive MIMO systems,” *IEEE Communications Letters*, vol. 22, no. 12, pp. 2583–2586, 2018.
- [20] J. Jiang, Y. Yuan, and L. Zhen, “Multi-user hybrid precoding for dynamic subarrays in mmWave massive MIMO systems,” *IEEE Access*, vol. 7, pp. 101718–101728, 2019.
- [21] A. Alkhateeb, G. Leus, and R. W. Heath, “Limited feedback hybrid precoding for multi-user millimeter wave systems,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 6481–6494, 2015.
- [22] Z. Cai, Y. Duan, and A. G. Bourgeois, “Delay efficient opportunistic routing in asynchronous multi-channel cognitive radio networks,” *Journal of Combinatorial Optimization*, vol. 29, no. 4, pp. 815–835, 2015.
- [23] Q. Chen, Z. Cai, L. Cheng, and H. Gao, “Low-latency data aggregation scheduling for cognitive radio networks with non-predetermined structure,” *IEEE Transactions on Mobile Computing*, p. 1, 2020.
- [24] L. Liang, W. Xu, and X. Dong, “Low-complexity hybrid precoding in massive multiuser MIMO systems,” *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 653–656, 2014.
- [25] D. H. Nguyen and T. Le-Ngoc, “MMSE precoding for multi-user MISO downlink transmission with non-homogeneous user SNR conditions,” *EURASIP Journal on Advances in Signal Processing*, vol. 2014, Article ID 85, 2014.
- [26] P. V. Amadori and C. Masouros, “Low RF-complexity millimeter-wave beamspace-MIMO systems by beam selection,” *IEEE Transactions on Communications*, vol. 63, no. 6, pp. 2212–2223, 2015.
- [27] W. Croswell, “Antenna theory, analysis, and design,” *IEEE Antennas and Propagation Society Newsletter*, vol. 24, no. 6, pp. 28–29, 1982.
- [28] Z. Cai, S. Ji, J. He, and A. G. Bourgeois, “Optimal distributed data collection for asynchronous cognitive radio networks,” in *2012 IEEE 32nd International Conference on Distributed Computing Systems*, pp. 245–254, Macau, China, 2012.
- [29] Z. Cai, S. Ji, J. He, L. Wei, and A. G. Bourgeois, “Distributed and asynchronous data collection in cognitive radio networks with fairness consideration,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2020–2029, 2014.
- [30] A. B. Sediq, R. H. Gohary, R. Schoenen, and H. Yanikomeroglu, “Optimal tradeoff between sum-rate efficiency and Jain’s fairness index in resource allocation,” *IEEE Transactions on Wireless Communications*, vol. 12, no. 7, pp. 3496–3509, 2013.

## Research Article

# Multijob Associated Task Scheduling for Cloud Computing Based on Task Duplication and Insertion

Lei Shi , Jing Xu, Lunfei Wang, Jie Chen, Zhifeng Jin, Tao Ouyang, Juan Xu ,  
and Yuqi Fan 

*School of Computer Science and Information Engineering, Intelligent Interconnected Systems Laboratory of Anhui Province, Hefei University of Technology, Hefei, Anhui 230601, China*

Correspondence should be addressed to Yuqi Fan; [yuqi.fan@hfut.edu.cn](mailto:yuqi.fan@hfut.edu.cn)

Received 27 November 2020; Revised 8 March 2021; Accepted 10 April 2021; Published 28 April 2021

Academic Editor: Yan Huang

Copyright © 2021 Lei Shi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the emergence and development of various computer technologies, many jobs processed in cloud computing systems consist of multiple associated tasks which follow the constraint of execution order. The task of each job can be assigned to different nodes for execution, and the relevant data are transmitted between nodes to complete the job processing. The computing or communication capabilities of each node may be different due to processor heterogeneity, and hence, a task scheduling algorithm is of great significance for job processing performance. An efficient task scheduling algorithm can make full use of resources and improve the performance of job processing. The performance of existing research on associated task scheduling for multiple jobs needs to be improved. Therefore, this paper studies the problem of multijob associated task scheduling with the goal of minimizing the jobs' makespan. This paper proposes a task Duplication and Insertion algorithm based on List Scheduling (DILS) which incorporates dynamic finish time prediction, task replication, and task insertion. The algorithm dynamically schedules tasks by predicting the completion time of tasks according to the scheduling of previously scheduled tasks, replicates tasks on different nodes, reduces transmission time, and inserts tasks into idle time slots to speed up task execution. Experimental results demonstrate that our algorithm can effectively reduce the jobs' makespan.

## 1. Introduction

Due to the rapid development of cloud computing and cloud infrastructure, an increasing number of applications are migrating to the cloud. Because of the strong extensibility and need for cloud computing, many existing tasks need powerful computing power and computing resources. The core of cloud computing is to coordinate many computer resources together, such that users can obtain unlimited resources and process users' jobs. A cloud platform is built on a variety of computing and network components to achieve the supply and distribution of various resources, and different components have great heterogeneity. Usually, users submit jobs to the cloud platform. The cloud platform divides jobs into multiple associated tasks and assigns them to different components for execution. Components need to communicate to exchange data and get the final result.

The associated tasks of a job can usually be represented by a directed acyclic graph (DAG). Each node in the DAG represents a task, and the directed arc represents the sequence constraints between tasks [1, 2]. Only when the current node has completed can the tasks of the subsequent nodes be executed. The execution of the subsequent node may need the result of the predecessor node as the input, so there is data transmission between the nodes. Based on the need for job processing performance, there has been extensive research on associated task scheduling. The DAG-based task scheduling is widely used in DNA detection, image recognition, geological survey, climate prediction, and other practical applications [3]. A heuristic algorithm based on genetic algorithms and task duplication was proposed in [4]. A task duplication-based scheduling algorithm was introduced in [5]. The joint problem of task assignment and scheduling considering multidimensional task diversity

was modeled as a reverse auction with task owners being auctioneers; four auction schemes were designed to satisfy different application requirements [6, 7]. The preference of participants for different perceptual tasks is also a key factor to be considered in the auction mechanism, because assigning the least favorite task will hinder participants from participating in future perceptual tasks. A concept of “mutual preference degree” was proposed to capture a participant’s preference, and a preference-based auction mechanism (PreAm) was designed to simultaneously guarantee individual rationality, budget feasibility, preference truthfulness, and price truthfulness [8]. A solution was proposed to detect and remove both short-term and long-term traffic redundancy through a two-layer redundancy elimination design [9]. A list-based scheduling algorithm called Predict Earliest Finish Time (PEFT) was proposed to introduce an optimistic cost table (OCT), which was used for task ranking and processor selection [10].

The processor may have idle time slots during processing multiple jobs. If the idle time slots can be fully utilized, the processing time of jobs can be reduced and the performance of job processing can be enhanced. A simple task scheduling algorithm may not be able to make full use of idle resources, so a more refined task scheduling algorithm is needed. In this paper, we study the associated task scheduling algorithm to minimize the jobs’ makespan when the components of the cloud computing platform are heterogeneous.

The main contributions of this paper are summarized as follows.

- (1) Aiming at minimizing the jobs’ makespan, we model the associated task scheduling problem
- (2) We propose a list scheduling algorithm based on task replication and task insertion (DILS), which combines dynamic completion time prediction, task replication, and task insertion. According to the previous task scheduling situation, the algorithm dynamically predicts the remaining completion time of tasks, replicates tasks on different nodes, reduces transmission time, and inserts tasks into idle time slots to accelerate task execution
- (3) Experimental results demonstrate that our algorithm can effectively improve the performance of job processing

The remainder of this paper is organized as follows. Section 2 summarizes the related work. Section 3 proposes the problem model of associated task scheduling. Section 4 describes the multijob task scheduling algorithm. Section 5 compares our algorithm with other algorithms. Finally, Section 6 summarizes the work of this paper.

## 2. Related Work

In cloud computing platforms, there are many cases of processing multijob tasks, and the scheduling problem of multijob tasks is important to improve the performance of job processing. Previous studies usually took the makespan and

energy consumption as the optimization objectives. Related works can be divided into the following categories.

The first category of the research investigates the scheduling strategy based on task priority. For example, a task scheduling algorithm based on the multipriority queue genetic algorithm (MPQGA) for heterogeneous computing systems was proposed [11]. In this paper, the authors used a genetic algorithm to assign priority to each task and used the heuristic method of the earliest completion time to complete the processor assignment; the authors also designed crossover, mutation, and fitness functions suitable for DAG scheduling. An algorithm was proposed to calculate the priority of each task to schedule tasks. The algorithm first processes the tasks with higher priority to satisfy the deadline [12]. A performance-effective task scheduling (PETS) algorithm was proposed to calculate the priority of each task based on task communication cost and average computation cost and then select the processor with the minimum earliest finishing time for each task [13].

The second category of the research studies the scheduling strategy based on a genetic algorithm. A learner genetic algorithm (denoted by LAGA) was presented to address static scheduling for processors in homogeneous computing systems [14]; the authors proposed two learning criteria named the steepest ascent learning criterion and next ascent learning criterion where they use the concepts of penalty and reward for learning; the algorithm exploits an efficient search method for solving a scheduling problem, such that the speed of finding a schedule could be accelerated and the trapping in local optimal could be alleviated; the algorithm also takes into consideration the idle time reuse criterion during the scheduling process to reduce the makespan. A decentralized scheduling algorithm based on genetic algorithms for the problem of DAG scheduling was proposed [15]; the genetic algorithm presents an effective method for optimization and could consider multiple criteria in the optimization process.

The third category of the research studies the scheduling strategy based on load balancing. A task scheduling mechanism was proposed based on two levels of load balance, which satisfies the dynamic task requirements of users and improves the utilization of resources [16]. A stochastic load balancing scheme was designed to provide a probabilistic guarantee against the resource overloading with virtual machine migration, while minimizing the total migration overhead [17].

The fourth category of the research focuses on QoS or heterogeneity during scheduling. A complex scene was considered where multiple moving MDs share multiple heterogeneous MEC servers, and a problem named the minimum energy consumption problem in a deadline-aware MEC system was formulated [18]. A network model aimed at improving user experience by pushing the scheduling problem to the task layer was proposed [19]. A QoE requirement was designed to generalize the QoS requirements of a task, which is the ratio requirement. Following this design, a corresponding scheduling policy was proposed to capture QoE for each task and then reached an application-aware transmission allocation. The  $q$ -coverage MLAS problem was investigated, which can guarantee that the aggregated nodes are

distributed evenly [20]; two algorithms were proposed by scheduling the communication tasks in the bottom-up and top-down manner, respectively; three algorithms were then proposed to ensure that the aggregation nodes are evenly distributed in the network with low delay; additionally, the method to extend the proposed algorithms for the BF-WSNs with multiple channels was also studied. Three allocation-aware task scheduling algorithms were proposed for a multicloud environment [21]; the algorithms are based on the traditional Min–Min and Max–Min algorithms and extended for multicloud environment; all the algorithms undergo three common phases, namely, matching, allocating, and scheduling, to fit them in the multicloud environment. A heuristic algorithm was presented for constructing streaming DAG which converts a large graph together to a DAG with the graph traversal algorithm, and then, a three-stage heterogeneous-aware cluster-scheduling algorithm was proposed to schedule the DAG to a heterogeneous cloud for parallel processing [22]; in the first stage, a parallel linear clustering algorithm was designed to cluster the DAG into a series of linear clusters with different granularities; in the second stage, a heterogeneous-aware load balancing algorithm was designed to map these clusters to different computing nodes in the cloud; in the last stage, a task sorting algorithm was designed to allocate the start time of these clusters as early as possible.

The fifth category of the research investigates the scheduling strategy based on task insertion. A Heterogeneous Earliest Finish Time (HEFT) algorithm was proposed to select the task with the highest upward rank value each time, then assign the task to the processor, and make use of task insertion to speed up the completion of the task and minimize the earliest completion time of the task [23]. A heterogeneous scheduling algorithm was proposed [24]; the algorithm takes the improved quantity as the calculation weight and communication weight and adopts an input task repeated selection strategy and an optimization strategy of idle time slot task insertion to improve the efficiency of task scheduling.

However, the existing works do not consider the scheduling problem in the case of multijob associated tasks. In this paper, we use DAGs to represent multijob associated tasks and then propose a task Duplication and Insertion algorithm based on List Scheduling (DILS), which can accelerate the execution of associated tasks to minimize the jobs' makespan.

### 3. Associated Task Scheduling Model

*Example 1.* An example of an associated task scheduling problem is shown in Figure 1. The job consists of four tasks, task set  $T_k = \{t_1^k, t_2^k, t_3^k, t_4^k\}$  and directed arc set  $\{e_{i,j}^k \mid t_i^k, t_j^k \in T_k\}$ . The weight  $c(t_i^k, t_j^k)$  of directed arc  $e_{i,j}^k$  is the time required for data transmission from the server where task  $t_i^k$  is running to the server where task  $t_j^k$  will run. The weight of the link from task to server is the execution time required for the task to run on the server.

Assume that the job set with different priorities to be processed is  $J = \{J_1, \dots, J_k, \dots\}$ , where  $J_k$  is the  $k$ th DAG-based job with multiple associated tasks. The tuple  $J_k = \langle T_k, E_k \rangle$

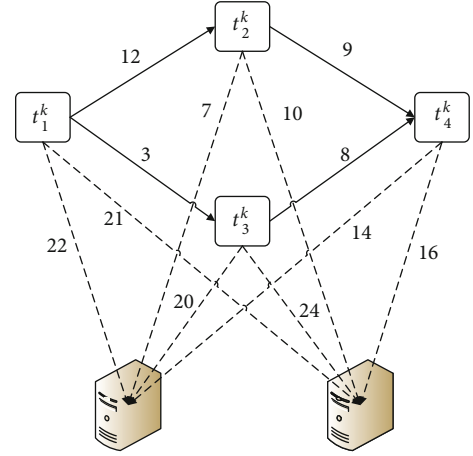


FIGURE 1: Architecture of the associated tasks and processor.

represents the job  $J_k$ , where  $T_k$  represents the task set of  $J_k$  and  $E_k$  is the directed arc set  $\{e_{i,j}^k \mid t_i^k, t_j^k \in T_k\}$  between tasks. Each directed arc represents the execution order constraint of two tasks. For example,  $e_{i,j}^k$  means that the task  $t_j^k$  cannot be executed until the task  $t_i^k$  is executed; that is,  $t_i^k$  is the preceding task of  $t_j^k$ ,  $t_j^k$  is the successive task of  $t_i^k$ , and  $t_i^k$  and  $t_j^k$  are the  $i$ th and  $j$ th tasks in job  $J_k$ , respectively. The directed arc also indicates that the execution of the subsequent task needs the result of the antecedent task execution. The weight  $c(t_i^k, t_j^k)$  of directed arc  $e_{i,j}^k$  is the time required for data to be transferred from the server where task  $t_i^k$  is running to the server running  $t_j^k$ . When two tasks connected by a directed arc run on the same server, weight  $c(t_i^k, t_j^k) = 0$ ; that is, the communication within the same server is negligible.

A DAG-based job is shown in Figure 2. In the DAG, task set  $T_k = \{t_1^k, \dots, t_i^k, \dots, t_{10}^k\}$  contains 10 tasks and directed arc set  $\{e_{i,j}^k \mid t_i^k, t_j^k \in T_k\}$  includes 15 directed arcs. For example, the directed arc from node  $t_2^k$  to node  $t_8^k$  indicates that  $t_2^k$  is the preceding task of task  $t_8^k$ , and the weight of this directed arc represents the communication time between the server running task  $t_2^k$  and the server running task  $t_8^k$  when two tasks are running on different servers.

Due to the heterogeneity of servers, the execution time of the same task may be different on different servers.  $P = \{p_1, \dots, p_s, \dots\}$  represents the set of servers that will run the tasks. We use an execution time matrix  $W_k = T_k \times P$  to list all possible mapping between the task  $t_i^k$  and the server  $p_s$ . Each element  $w(t_i^k, p_s)$  in the matrix  $W_k$  represents the time required for the task  $t_i^k$  to run on the server  $p_s$ . The execution time matrix  $W_k$  of DAG is shown in Table 1. The server set  $P = \{p_1, p_2, p_3\}$  contains 3 servers, and the element of mapping task  $t_2^k$  and server  $p_2$  indicates that the execution time of task  $t_2^k$  on server  $p_2$  is 18.

*Definition 2.* In DAG-based job  $J_k$ , ingress task  $t_{in}^k$  is the task without any preceding tasks, and egress task  $t_{out}^k$  is the task without any successive tasks.

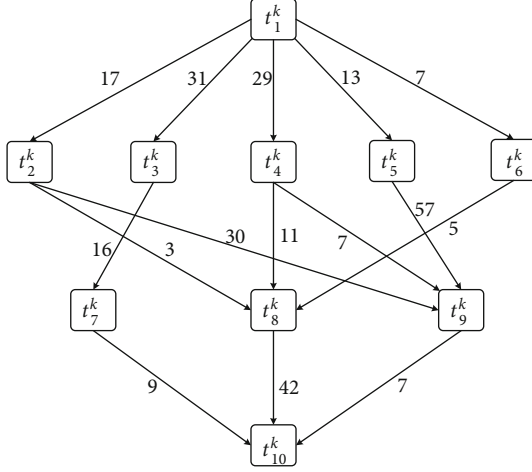


FIGURE 2: DAG-based job example.

TABLE 1: Execution time matrix example.

	$p_1$	$p_2$	$p_3$
$t_1^k$	22	21	36
$t_2^k$	22	18	18
$t_3^k$	32	27	43
$t_4^k$	7	10	4
$t_5^k$	29	27	35
$t_6^k$	26	17	24
$t_7^k$	14	25	30
$t_8^k$	29	23	36
$t_9^k$	15	21	8
$t_{10}^k$	13	16	33

For the DAG-based job shown in Figure 2, the ingress task and egress task are tasks  $t_1^k$  and  $t_{10}^k$ , respectively. If a job includes more than one ingress task, we add a virtual task node with zero computation cost as the virtual ingress task and add a directed arc from the virtual ingress task to each of the original ingress task nodes with zero weight.

**Definition 3.**  $EST(t_i^k, p_s)$ , the earliest start time (EST) that task  $t_i^k$  can be executed on server  $p_s$ , is defined via Equation (1), where  $EAT(p_s)$  is the earliest time that server  $p_s$  is available,  $pre(t_i^k)$  is the set of the preceding tasks of task  $t_i^k$ , and  $AFT(t_j^k)$  is the actual finish time of task  $t_j^k$ :

$$EST(t_i^k, p_s) = \max \left\{ EAT(p_s), \max_{t_j^k \in pre(t_i^k)} \left\{ AFT(t_j^k) + c(t_j^k, t_i^k) \right\} \right\}, \quad \forall t_i^k \in T, \forall p_s \in P. \quad (1)$$

**Definition 4.**  $EFT(t_i^k, p_s)$ , the earliest finish time (EFT) that server  $p_s$  can complete the execution of task  $t_i^k$ , is calculated via

$$EFT(t_i^k, p_s) = EST(t_i^k, p_s) + w(t_i^k, p_s), \quad \forall t_i^k \in T, \forall p_s \in P. \quad (2)$$

**Definition 5.** For job set  $J$  in which the jobs have different priorities, the makespan of job set  $J$ ,  $\Gamma(J)$ , is the completion time of all the jobs and can be calculated via

$$\Gamma(J) = \max_{k \in J} AFT(t_{out}^k). \quad (3)$$

Given job set  $J$ , the DAG of each job, server set  $P$ , and the execution time of each job on each server, we need to schedule the jobs in  $J$  on the servers in  $P$ , so as to minimize  $\Gamma(J)$ , i.e., the makespan of job set  $J$ . The detailed notations are listed in Table 2.

#### 4. Multijob Task Scheduling Algorithm

The single-job associated task scheduling problem with the aim of minimizing the makespan is an NP-hard problem [25]. Therefore, the multijob associated task scheduling problem is also an NP-hard problem [26].

In this section, a task Duplication and Insertion based on List Scheduling (DILS) algorithm is proposed, which incorporates dynamic finish time prediction, task replication, and task insertion. The algorithm dynamically predicts the remaining execution time of each task according to the scheduling of scheduled tasks. Then, the algorithm schedules the tasks according to the latest time remaining on the server, which can minimize the remaining time. After each task is scheduled, the algorithm can advance the start time of the task by adopting task replication and task insertion. The remaining execution time of each reserved task is updated when the task is scheduled. The algorithm DILS shown in Algorithm 1 consists of five parts: calculation of task remaining time, selection of tasks to be scheduled, server allocation of tasks to be scheduled, task replication, and task insertion.

**4.1. Task Remaining Time Calculation.** There is a weight for each task which is the total communication and computation time of all the subsequent tasks. A predicted remaining time (PRT) table maintains the weights of all the tasks, and each element  $PRT(t_i^k, p_s)$  in the table PRT represents the predicted remaining time required for executing all the successive tasks of task  $t_i^k$  ( $1 \leq i \leq N$ ), when it is allocated to server  $p_s \in P$ . The weight of task  $t_i^k$  is closely related to its successive tasks and the number of servers, and we let

$$A_{t,s}^k = \max_{t_j^k \in suc(t_i^k)} \left\{ \min_{p_t \in P} \left\{ PRT(t_j^k, p_t) + w(t_j^k, p_t) + c(t_i^k, t_j^k) \right\} \right\}, \quad (4)$$

TABLE 2: Table of notations.

Notation	Description
$J_k$	The $k$ th DAG-based job with multiple associated tasks.
$T_k$	The task set of $J_k$ .
$E_k$	The directed arc set $\{e_{ij}^k \mid t_i^k, t_j^k \in T_k\}$ between tasks.
$P$	The set of servers.
$t_i^k$	The $i$ th task of $J_k$ .
$e_{ij}^k$	The directed arc of $t_i^k$ and $t_j^k$ .
$c(t_i^k, t_j^k)$	The transmission time required for data.
$p_s$	The server that will run the tasks.
$w(t_i^k, p_s)$	The time required for the task $t_i^k$ to run on the server $p_s$ .
$t_{in}^k$	The task without any preceding tasks.
$t_{out}^k$	The task without any successive tasks.
$\text{pre}(t_i^k)$	The set of the preceding tasks of task $t_i^k$ .
$\text{suc}(t_i^k)$	The set of the successive tasks of task $t_i^k$ .
$M$	The number of servers.

**Input:** Server set  $P$ , job set  $J$  with each job's DAG, and execution time matrices

**Output:** Makespan of the jobs

```

1: for each job  $J_K \in J$  do
2:   Calculate the predicted remaining time of each task via Eq. (6) and create the Prediction of Remaining Time (PRT) table;
3:   Create an empty ready task list and add the ingress task to the list;
4:   while ready task list is not empty do
5:     for each task  $t_i^k$  in ready task list do
6:       Compute the average path length of task  $t_i^k$  via Eq. (8);
7:     end for
8:     Select the task with the maximum average path length with Eq. (9);
9:     Assign the server leading to the minimum estimated path length via Eq.(10) to the task;
10:    Task duplication and task insertion;
11:    Update the ready task list;
12:  end while
13: end for
14: return the completion time of the last scheduled task.

```

ALGORITHM 1: Algorithm DILS.

where  $\text{suc}(t_i^k)$  is the set of successive tasks of task  $t_i^k$ , and let

$$B_i^k = \frac{\sum_{t_j^k \in \text{suc}(t_i^k)} \sum_{p_t \in P} \text{PRT}(t_j^k, p_t) / M}{M}, \quad (5)$$

where  $M$  is the number of servers. Note that no matter which server runs the egress task, the weight of the exit task of each job is 0. That is, for any  $p_t \in P$ ,  $\text{PRT}(t_{out}^k, p_t) = 0$ :

$$\text{PRT}(t_i^k, p_s) = \max \{A_{i,s}^k, B_i^k\}, \quad \forall t_i^k \in T, \forall p_s \in P. \quad (6)$$

The weight of task  $t_i^k$  is calculated via Equation (6). In each DAG, starting from the egress task to the ingress task, algorithm DILS recursively calculates backwards the weight of each task on each server and obtains the PRT table.

*4.2. Selection of the Task to Be Scheduled.* When all the preceding tasks of task  $t_i^k$  are scheduled, we call the task  $t_i^k$  *ready*. We create a ready task list (RTL) to maintain all ready tasks. Initially, the ready task list contains only one node, ingress task  $t_{in}^k$ , because only the ingress task of each DAG-based job is ready. We calculate the earliest start

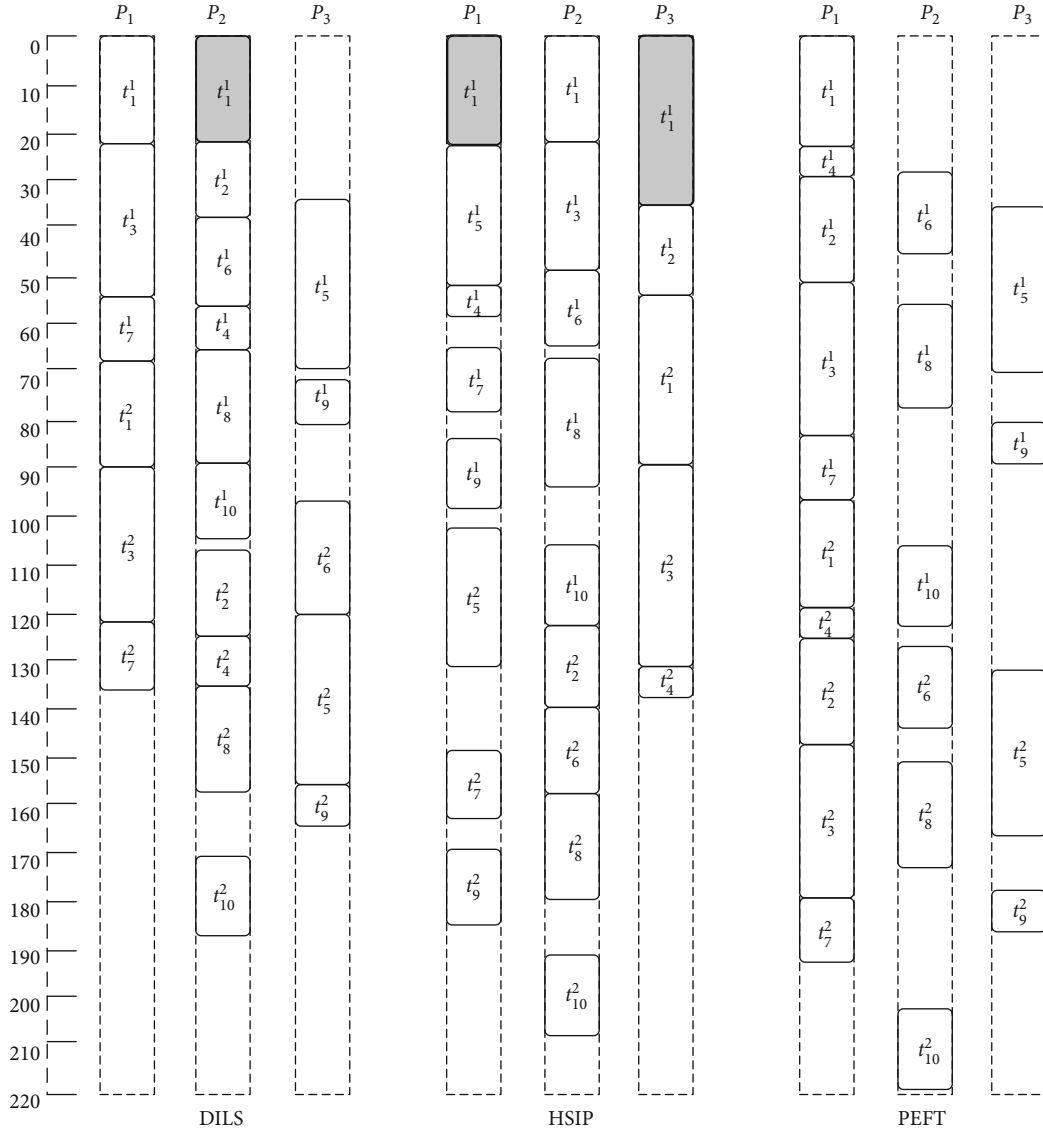


FIGURE 3: The example of scheduling result.

time (EST) of each task in the ready task list. The EST of the task  $t_i^k$  on the server  $p_s$ ,  $t_i^k$  on server  $p_s$ ,  $\text{EST}(t_i^k, p_s)$  is calculated via Equation (1). The estimated path length (EPL) of the task  $t_i^k$  assigned to the server  $p_s$  is calculated via

$$\text{EPL}(t_i^k, p_s) = \text{EST}(t_i^k, p_s) + w(t_i^k, p_s) + \text{PRT}(t_i^k, p_s), \quad \forall t_i^k \in T_k, \forall p_s \in P. \quad (7)$$

Since each task may be assigned to any server  $p_s \in P$ , the average path length (APL)  $\text{APL}(t_i^k)$  of task  $t_i^k$  which is ready can be calculated with

$$\text{APL}(t_i^k) = \frac{\sum_{p_s \in P} \text{EPL}(t_i^k, p_s)}{M}, \quad \forall t_i^k \in T_k. \quad (8)$$

Considering the difference in execution time of tasks on different paths, the task  $t_i^k$  which is ready and has the maximum average path length is selected as the task to be scheduled; that is,

$$t_i^k = \text{argmax}_{t_j^k \in \text{RTL}} \left\{ \text{APL}(t_j^k) \right\}. \quad (9)$$

The selection of tasks is related to the EST of the task, and EST changes dynamically with the scheduling results of the previous tasks. Therefore, in the process of scheduling associated tasks, the selection of tasks to be scheduled will be changed dynamically.

**4.3. Allocation of the Server for the Task to Be Scheduled.** When the task  $t_i^k$  can get the minimum estimated path

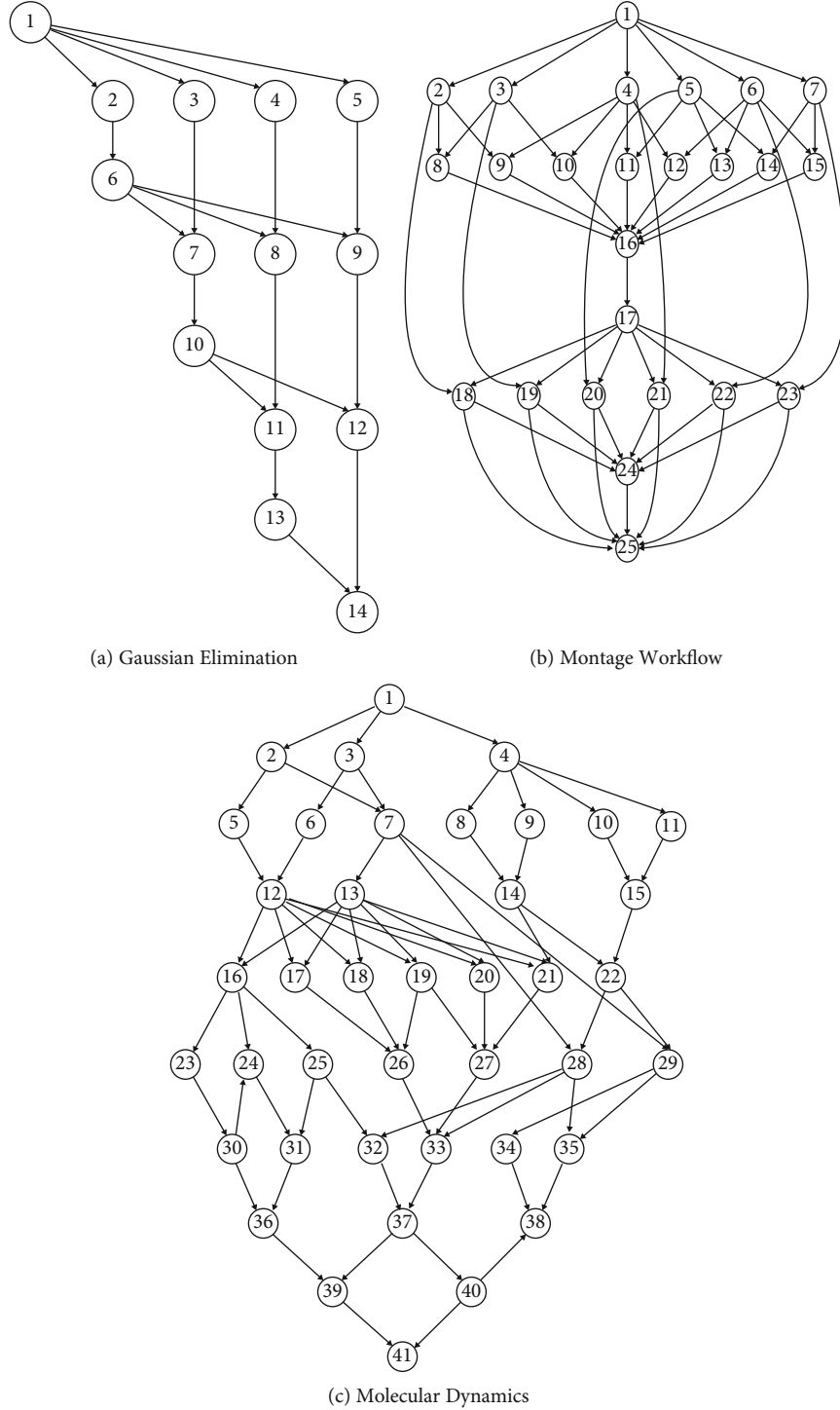


FIGURE 4: Real-world DAGs.

length on the server  $p_s$ , we assign the to-be-scheduled task  $t_i^k$  to server  $p_s$  to reduce the makespan; that is,

$$p_s = \operatorname{argmin}_{p_t \in P} \left\{ \text{EPL} \left( t_i^k, p_t \right) \right\}. \quad (10)$$

For task  $t_i^k$ , if multiple servers can get the same minimum estimated path length, we randomly assign task  $t_i^k$  to one of the servers. The actual start time of the task

$t_i^k$  can be calculated by Equation (2). When the server of the task to be scheduled is assigned, some other tasks may be ready to be scheduled, so the ready task list needs to be updated.

**4.4. Task Duplication.** Task duplication obtains multiple copies of the task  $t_i^k$  and then assigns the task copies to different servers for execution, such that the direct successive tasks of task  $t_i^k$  can



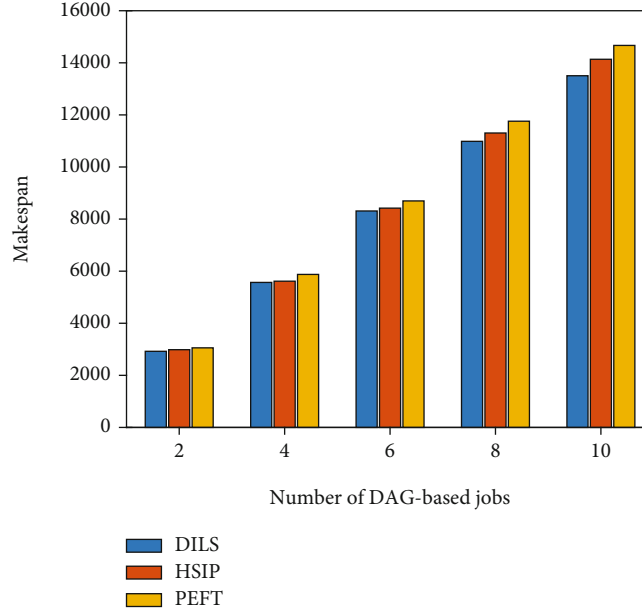


FIGURE 5: The makespan with the different numbers of DAG-based jobs.

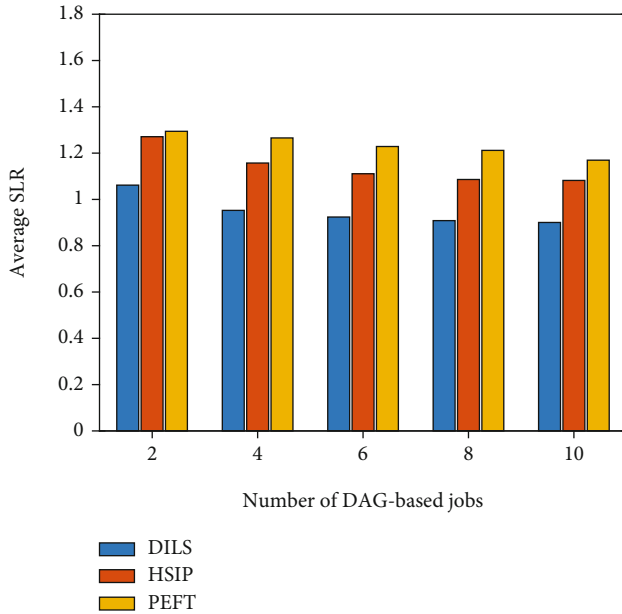


FIGURE 6: The average SLR with the different numbers of DAG-based jobs.

be executed immediately after the execution of  $t_i^k$ , instead of waiting for the data generated by task  $t_i^k$  to be transmitted through the network. Task duplication works as follows.

- (1) Each task  $t_j^k$  in  $\text{pre}(t_i^k)$  is assigned a time weight  $\tau_j^k$ , that is, the time required for the data generated by the preceding task of task  $t_i^k$  to be transmitted to the processor  $p_s$  where task  $t_j^k$  is to be executed
- (2) All the preceding tasks in  $\text{pre}(t_i^k)$  are sorted in nonascending order according to the time weights

- (3) Each of the preceding tasks in  $\text{pre}(t_i^k)$  is iterative processed. Schedule a copy of task  $t_j^k \in \text{pre}(t_i^k)$  to the earliest idle time slot (ITS) of server  $p_s$ , if the following conditions are met: (I)  $\tau_j^k > \text{EAT}(p_s)$ ; that is,  $\tau_j^k$  is greater than the earliest available time of processor  $p_s$ ; (II) there is an idle time slot for the copy of  $t_j^k$ ; and (III) task  $t_i^k$  can start earlier by duplicating  $t_j^k$

**4.5. Task Insertion.** Task insertion inserts a task into an idle time slot on a processor which is occupied by some tasks after the time slot. Task insertion works as follows.

- (1) When task  $t_i^k$  is to be scheduled on server  $p_s$ , we search all the idle time slots on server  $p_s$ . When the idle time slot meets the following conditions: (I)  $\text{EST}(t_i^k, p_s)$ , the earliest start time of task  $t_i^k$ , is no earlier than the start time of the idle time slot; (II)  $\text{EFT}(t_i^k, p_s)$ , the earliest finish time of task  $t_i^k$  on processor  $p_s$ , is no later than the end time of the idle time slot; we select this idle time slot to insert task  $t_i^k$
- (2) If more than one idle time slot satisfies the above conditions, we choose the idle time slot with the smallest difference between the length of the idle time slot and the execution time of task  $t_i^k$

**4.6. Time Complexity.** Suppose the number of DAGs is  $K$ , the number of processors is  $M$ , and the number of task nodes in each DAG is  $N_1, N_2, \dots, N_K$ , respectively. For the  $i$ th ( $1 \leq i \leq k$ ) DAG, DILS requires the computation of a PRT table, which consumes  $O(M \times N_i^2)$  time. The time complexity of calculating the earliest start time and estimated path length of the task on the processor is  $O(M^2 \times N_i^2)$ . By executing

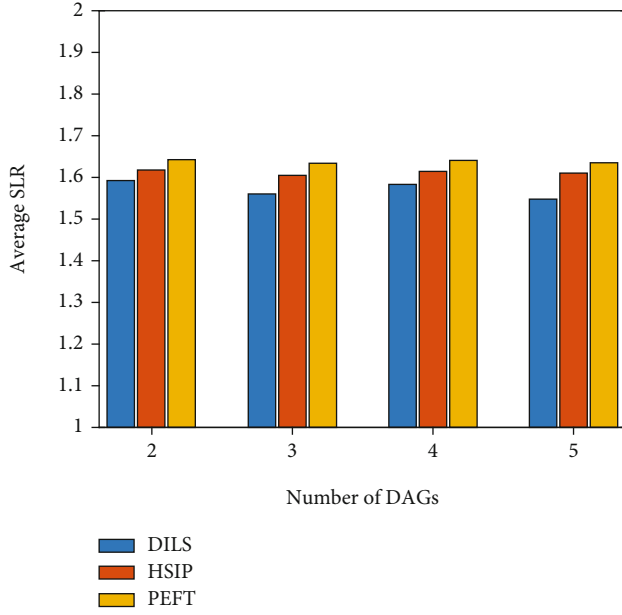


FIGURE 7: Average SLR with different numbers of DAG-based jobs for Gaussian Elimination.

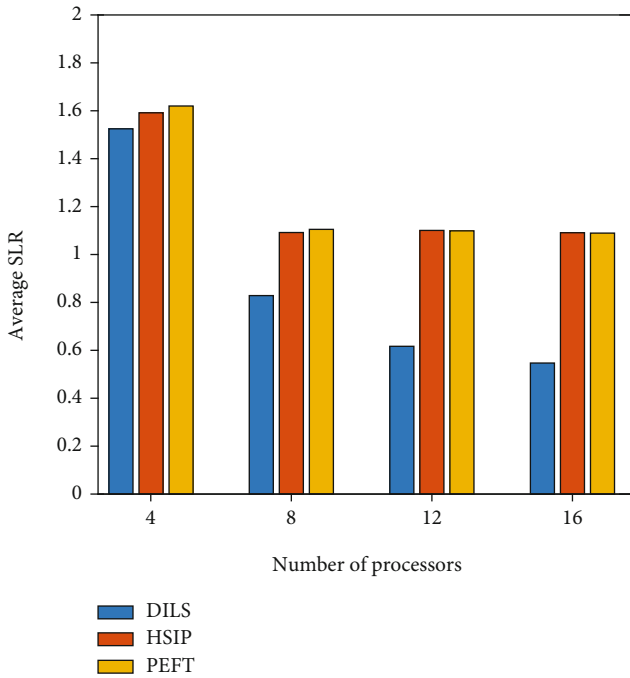


FIGURE 8: Average SLR with different numbers of processors for Gaussian Elimination.

the insertion mechanism in  $O(M \times N_i^2)$  and executing task replication in  $O(M \times N_i^2)$ , the time complexity of updating the task ready list is  $O(N_i^2)$ . Therefore, the time complexity of Algorithm 1 is  $O(M^2 \times N_1^2) + O(M^2 \times N_2^2) + \dots + O(M^2 \times N_k^2) = O(M^2 \times \sum_i^k N_i^2)$ . Algorithms HSIP [24] and PEFT [10] are also scheduling strategies based on task insertion, which

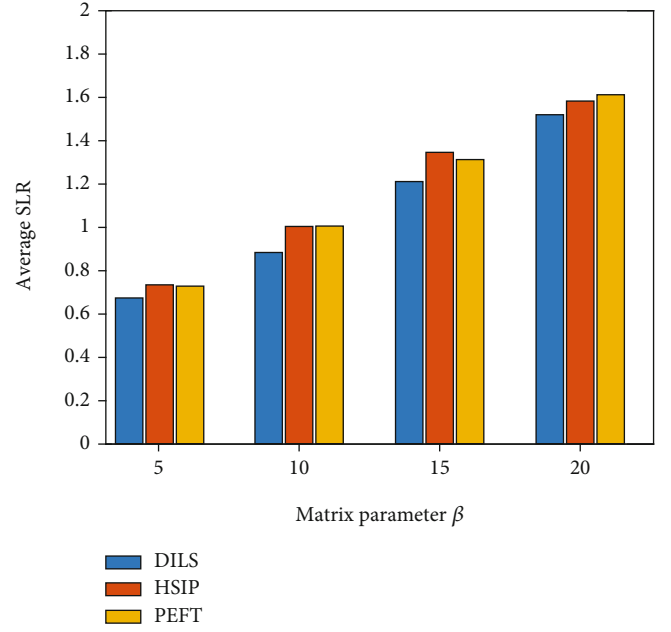


FIGURE 9: The average SLR with different matrix parameter  $\beta$ .

deal with the case of a single DAG with the time complexity of  $O(N^2 \times M)$ .

## 5. Simulation

In this section, we compare the performance of the proposed algorithm DILS with two state-of-the-art algorithms PEFT [10] and HSIP [24]. For this purpose, we consider the effect of the number of DAG-based jobs and processors on the performance of the algorithms. We also investigate the influence of some important parameters on the performance of the algorithms.

*Example 7.* The scheduling results of the three algorithms for job set  $J_s = \langle J_1, J_2 \rangle$  are illustrated in Figure 3, where DAGs of jobs  $J_1$  and  $J_2$  are the same as that described in Figure 2 and Table 1. The gray shadowed block represents the task to be copied by algorithms DILS and HSIP. In this example, we can see that algorithm DILS has better results than HSIP and PEFT.

*5.1. Simulation Setup.* In this paper, we will evaluate the performance of the three algorithms in terms of the scheduling length ratio (SLR). SLR is the ratio of the scheduling length to the minimum scheduling length by ignoring the communication time, which can be calculated by

$$\text{SLR} = \frac{\sum_{J_k \in J} \text{makespan}}{\sum_{J_k \in J} \sum_{t_i^k \in \text{CP}_{\text{MIN}}} \min_{p_s \in P} \{w(t_i^k, p_s)\}}, \quad (11)$$

where  $\text{CP}_{\text{MIN}}$  is the minimum length of the critical path in the DAG-based job after ignoring the communication time between tasks, while the critical path of a DAG-based job is the longest path from the ingress task to the egress task.

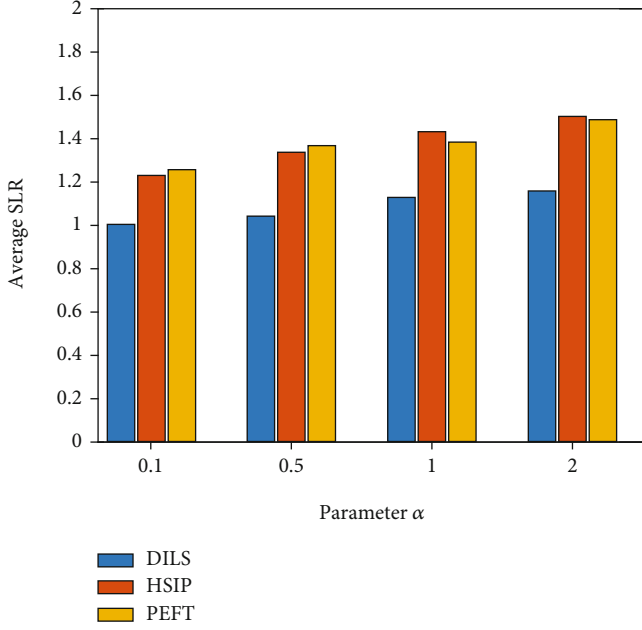


FIGURE 10: Average SLR with different  $\alpha$  for Montage Workflow.

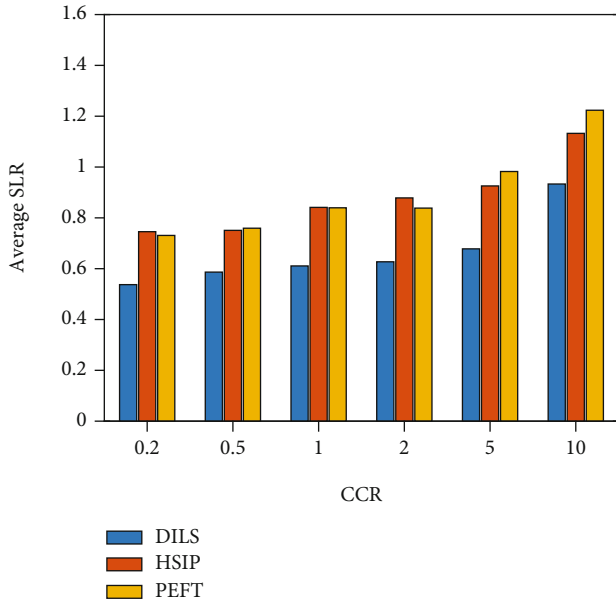


FIGURE 11: Average SLR with different CCR for Montage Workflow.

In the simulations, we use DAG-based jobs randomly generated by a DAG generator and real-world DAG-based jobs. The DAG generator uses the same parameters as in [25, 27]. The DAG topology parameters are as follows:

- (1) DAG average calculation time  $c_{\text{DAG}}^-$ : indicating the average execution time of the tasks in the DAG, which is randomly set during the simulation
- (2) Communication calculation ratio CCR: the ratio of the average communication time and the average

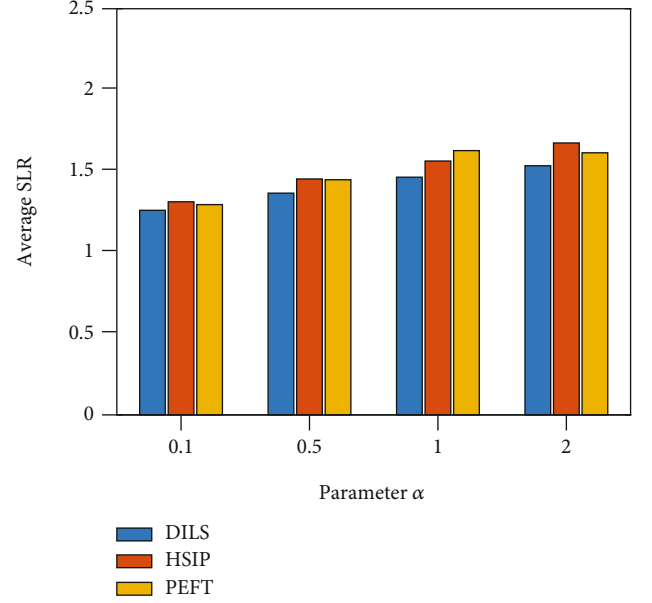


FIGURE 12: Average SLR with different  $\alpha$  for Molecular Dynamics Code.

execution time; the larger the value, the more communication-intensive the DAG-based job; the smaller the value, the more computation-intensive the DAG-based job

- (3) Heterogeneous parameter  $\alpha$ : representing the task execution time range on different processors; the larger the value, the more heterogeneous the processors

We select three classic DAG-based jobs from the real-world applications in the simulations.

- (1) Gaussian Elimination: it is used in linear algebraic programming for solving linear equations as shown in Figure 4(a). The number of nodes is  $N = (\beta^2 + \beta - 2)/2$  according to matrix parameter  $\beta$
- (2) Montage Workflow: it is applied to construct astronomical image mosaic. An example of Montage Workflow is shown in Figure 4(b)
- (3) Molecular Dynamics Code: it is an algorithm to implement the atomic and the molecular physical motion. An example of the Molecular Dynamics Code is depicted in Figure 4(c)

5.2. Performance of Algorithm DILS. Figure 5 shows the effect of changing the number of DAG-based jobs on the performance of the makespan of the three algorithms. The number of servers is set to 4, DAG-based jobs are randomly generated by the DAG generator, and the parameters CCR and  $\alpha$  are both set to 1. It can be seen from Figure 5 that the makespan increases with the increase of DAG-based jobs. In short, among the three algorithms, DILS achieves the best performance, followed by algorithm HSIP and algorithm PEFT.

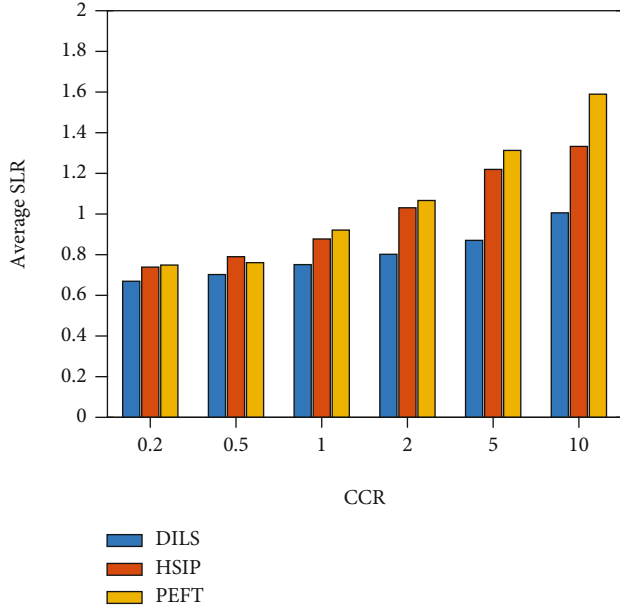


FIGURE 13: Average SLR with different CCR for Molecular Dynamics Code.

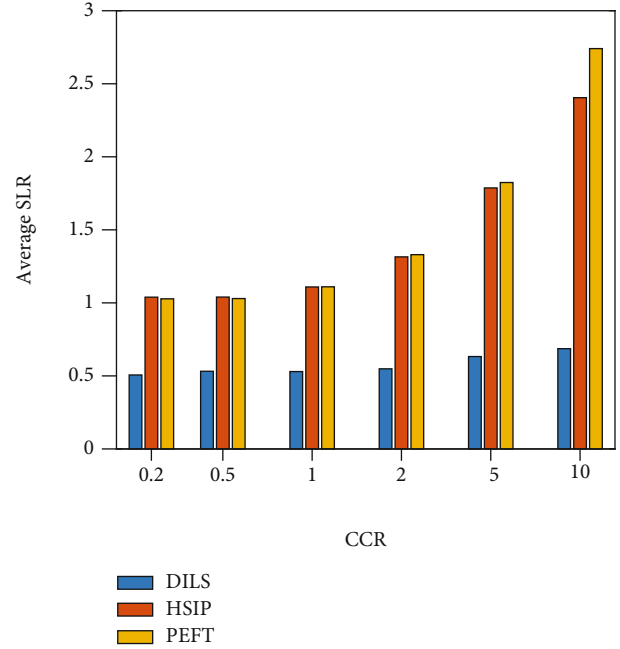


FIGURE 15: Average SLR with different CCR for Gaussian Elimination.

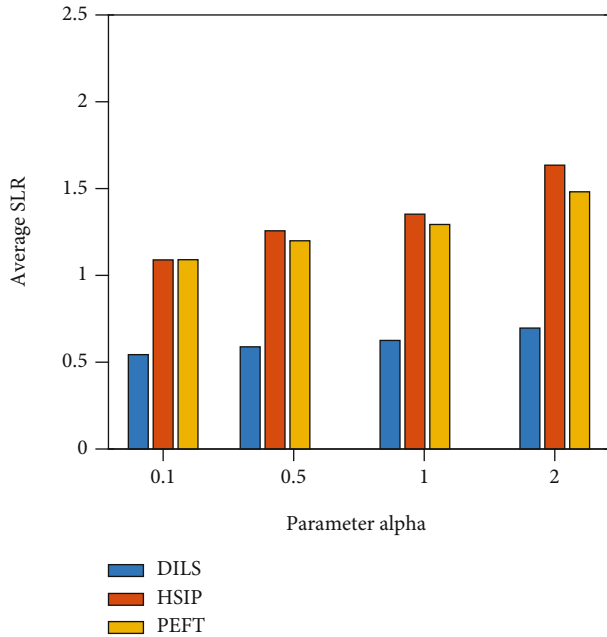


FIGURE 14: Average SLR with different  $\alpha$  for Gaussian Elimination.

When there are 10 DAG-based jobs, the algorithm DILS improves the performance of algorithm HSIP and PEFT by 4.5% and 7.9%, respectively.

Figure 6 shows the effect of the number of DAG-based jobs on the three algorithms on the average SLR. The number of servers is set to 8, DAG-based jobs are randomly generated by the generator, and the parameters CCR and  $\alpha$  are both set to 1. The average SLR generated by the three algorithms decreases with the increase of the number of DAG-based

jobs, because more idle time slots are utilized. Among the three algorithms, DILS can get the minimum average SLR, because DILS makes full use of the idle time slots. When the number of DAG-based jobs increases from 2 to 10, the performance of algorithm DILS compared with algorithm HSIP and PEFT is improved from 16.3% to 17.6% and from 17.9% to 25.0%, respectively.

In Figure 7, we show the impact of the number of DAG-based jobs on the average SLR for Gaussian Elimination jobs. The number of processors is 4, the parameter CCR is set to 1, and  $\alpha$  is set as 0.1. As the number of DAG-based jobs increases, the increase of average SLR is small. The difference between the three algorithms is obvious. However, our algorithm outperforms the other two algorithms.

Figure 8 illustrates the average SLR of the three algorithm changes with the increasing number of processors for Gaussian Elimination jobs. With the increase in the number of processors, the average SLR of our algorithm decreases significantly. Obviously, compared with other algorithms, our algorithm achieves better performance.

**5.3. Impact of Parameters.** Figure 9 illustrates the average SLR performance versus different matrix size parameter  $\beta$  for the Gaussian Elimination jobs when the number of servers is 2. It can be observed that the average SLR of the three algorithms increases with the increase of  $\beta$ . The jobs with a larger  $\beta$  consist of more tasks and hence require more computation and transfer time for all the associated tasks. The performance improvement of algorithm DILS on algorithms HSIP and PEFT is up to 11.9% and 12.1%, respectively.

Figure 10 describes the average SLR for Montage Workflow. Parameter  $\alpha$  varies in  $\{0.1, 0.5, 1, 2\}$ , when the parameter CCR = 1 and the numbers of DAG-based jobs and

servers are set as 5 and 4, respectively. The performance of algorithm DILS compared with algorithms HSIP and PEFT is improved from 18.4% to 22.9% and from 20.1% to 22.4%, respectively. For Figure 11, the parameter CCR increases from 0.2 to 10, when  $\alpha = 1$  and the numbers of DAG-based jobs and servers are set to 5 and 8, respectively. The average SLR increases with the increase of the parameter CCR, since the communication between tasks consumes more time as parameter CCR increases. It can be seen that the algorithm DILS always achieves the best performance among the three algorithms.

Figure 12 depicts the impact of parameter  $\alpha$  on the average SLR for Molecular Dynamics Code. The parameter CCR is equal to 1, and the numbers of DAG-based jobs and servers are set to 5 and 4, respectively. It can be seen that the average SLR of the three algorithms increases with the increase of heterogeneous parameter  $\alpha$ . Algorithm DILS always obtains the smallest results.

Figure 13 shows the impact of CCR on average SLR for Molecular Dynamics Code, when  $\alpha = 1$  and the numbers of DAG-based jobs and servers are set to 5 and 8, respectively. The performance of algorithm DILS compared with HSIP and PEFT is improved from 9.4% to 24.5% and from 10.6% to 36.7%, respectively.

Figure 14 describes the average SLR for Gaussian Elimination with different parameter  $\alpha$  varying in  $\{0.1, 0.5, 1, 2\}$ , when the parameter CCR = 1 and the numbers of DAG-based jobs and servers are set as 5 and 16, respectively. In general, algorithm DILS always achieves the best performance among the three algorithms, and algorithm HSIP performs better than algorithm PEFT. With the heterogeneous parameter  $\alpha$  increasing from 0.1 to 2, the average SLR of algorithm DILS is improved from 50% to 57.3% and from 50.2% to 53% compared with HSIP and PEFT, respectively.

Figure 15 illustrates the average SLR for Gaussian Elimination with different parameter CCR increasing from 0.2 to 10, when  $\alpha = 1$  and the numbers of DAG-based jobs and servers are set to 5 and 16, respectively. Compared with the other two algorithms, the average SLR performance of DILS increases from 51.3% to 71.5% and from 50.7% to 75%, respectively.

## 6. Conclusion

Careful multijob task scheduling is the key to achieve efficient job processing. This paper studied the problem of associated task scheduling of multiple jobs with the aim of minimizing jobs' makespan. We propose a task Duplication and Insertion algorithm based on List Scheduling (DILS). The algorithm combines dynamic prediction of task completion time, task replication, and task insertion. For multiple jobs, the expected completion time of their associated tasks is calculated to determine which task is scheduled and which processor to run the task. Some tasks are copied to different processors to reduce the transmission delay. There are many jobs to be processed, and hence, there may be idle time slots on the processor. Therefore, the tasks that meet the insertion conditions can be inserted into the idle time slots to speed up

the execution of the jobs. Simulation results demonstrated that algorithm DILS achieved good performance.

## Data Availability

There is no dataset for this article.

## Disclosure

An earlier version of the paper has been presented as a conference paper in WASA2020: International Conference on Wireless Algorithms Systems and Applications.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This document is the result of the Key Research and Development Project in Anhui Province (Grant No. 201904a06020024), National Key Research and Development Plan (Grant No. 2018YFB2000505), and National Natural Science Foundation of China (Grant No. 61806067).

## References

- [1] W. Chen, G. Xie, R. Li, Y. Bai, C. Fan, and K. Li, "Efficient task scheduling for budget constrained parallel applications on heterogeneous cloud computing systems," *Future Generation Computer Systems*, vol. 74, no. C, pp. 1–11, 2017.
- [2] H. Arabnejad and J. Barbosa, "Fairness resource sharing for dynamic workflow scheduling on heterogeneous systems," in *2012 IEEE 10th International Symposium on Parallel and Distributed Processing with Applications*, pp. 633–639, Leganes, Spain, 2012.
- [3] S. K. Panda and P. K. Jana, "Efficient task scheduling algorithms for heterogeneous multi-cloud environment," *Journal of Supercomputing*, vol. 71, no. 4, pp. 1505–1533, 2015.
- [4] T. Tsuchiya, T. Osada, and T. Kikuno, "A new heuristic algorithm based on GAs for multiprocessor scheduling with task duplication," in *Proceedings of 3rd International Conference on Algorithms and Architectures for Parallel Processing*, pp. 295–308, Melbourne, VIC, Australia, 1997.
- [5] R. Bajaj and D. P. Agrawal, "Improving scheduling of tasks in a heterogeneous environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 15, no. 2, pp. 107–118, 2004.
- [6] Z. Duan, W. Li, and Z. Cai, "Distributed auctions for task assignment and scheduling in mobile crowdsensing systems," in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pp. 635–644, Atlanta, GA, USA, 2017.
- [7] Z. Cai, Z. Duan, and W. Li, "Exploiting multi-dimensional task diversity in distributed auctions for mobile crowdsensing," *IEEE Transactions on Mobile Computing*, no. 99, p. 1, 2020.
- [8] Z. Duan, W. Li, X. Zheng, and Z. Cai, "Mutual-preference driven truthful auction mechanism in mobile crowdsensing," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 1233–1242, Dallas, TX, USA, 2019.

- [9] L. Yu, H. Shen, K. Sapra, L. Ye, and Z. Cai, "CoRE: cooperative end-to-end traffic redundancy elimination for reducing cloud bandwidth cost," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 2, pp. 446–461, 2017.
- [10] H. Arabnejad and J. G. Barbosa, "List scheduling algorithm for heterogeneous systems by an optimistic cost table," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 3, pp. 682–694, 2014.
- [11] Y. Xu, K. Li, J. Hu, and K. Li, "A genetic algorithm for task scheduling on heterogeneous computing systems using multiple priority queues," *Information Sciences*, vol. 270, pp. 255–287, 2014.
- [12] T. Choudhari, M. Moh, and T. Moh, "Prioritized task scheduling in fog computing," in *Proceedings of the ACMSE 2018 Conference*, pp. 1–8, Richmond, Kentucky, USA, 2018.
- [13] E. Ilavarasan, P. Thambidurai, and R. Mahilmanan, "Performance effective task scheduling algorithm for heterogeneous computing system," in *4th International Symposium on Parallel and Distributed Computing (ISPDC'05)*, pp. 28–38, Lillie, France, 2005.
- [14] H. Izadkhah, "Learning based genetic algorithm for task graph scheduling," *Applied Computational Intelligence and Soft Computing*, vol. 2019, 15 pages, 2019.
- [15] F. Pop, C. Dobre, and V. Cristea, "Genetic algorithm for dag scheduling in grid environments," in *IEEE International Conference on Intelligent Computer Communication & Processing*, Cluj-Napoca, Romania, 2009.
- [16] Y. Fang, F. Wang, and J. Ge, "A task scheduling algorithm based on load balancing in cloud computing," in *International conference on web information systems and mining*, pp. 271–277, Berlin, Heidelberg, 2010.
- [17] L. Yu, L. Chen, Z. Cai, H. Shen, Y. Liang, and Y. Pan, "Stochastic load balancing for virtual resource management in datacenters," *IEEE Transactions on Cloud Computing*, vol. 8, no. 2, pp. 459–472, 2020.
- [18] T. Zhu, T. Shi, J. Li, Z. Cai, and X. Zhou, "Task scheduling in deadline-aware mobile edge computing systems," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4854–4866, 2019.
- [19] X. Zheng, Z. Cai, J. Li, and H. Gao, "A study on application-aware scheduling in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 1787–1801, 2017.
- [20] Z. Cai and Q. Chen, "Latency-and-coverage aware data aggregation scheduling for multihop battery-free wireless networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 1770–1784, 2021.
- [21] S. K. Panda, I. Gupta, and P. K. Jana, "Task scheduling algorithms for multi-cloud systems: allocation-aware approach," *Information Systems Frontiers*, vol. 21, no. 2, pp. 241–259, 2019.
- [22] K. Hu, G. Zeng, S. Ding, and H. Jiang, "Cluster-scheduling big graph traversal task for parallel processing in heterogeneous cloud based on dag transformation," *IEEE Access*, vol. 7, pp. 77070–77082, 2019.
- [23] H. Topcuoglu, S. Hariri, and M.-Y. Wu, "Performance-effective and low-complexity task scheduling for heterogeneous computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 3, pp. 260–274, 2002.
- [24] G. Wang, Y. Wang, H. Liu, and H. Guo, "HSIP: a novel task scheduling algorithm for heterogeneous computing," *Scientific Programming*, vol. 2016, 11 pages, 2016.
- [25] Y. Fan, L. Tao, and J. Chen, "Associated task scheduling based on dynamic finish time prediction for cloud computing," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, Dallas, Texas, USA, 2019.
- [26] J. Ullman, "Np-complete scheduling problems," *Journal of Computer and System Sciences*, vol. 10, no. 3, pp. 384–393, 1975.
- [27] D. Cordeiro, G. Mounie, P. Swann, D. Trystram, J.-M. Vincent, and F. Wagner, "Random graph generation for scheduling simulations," in *Proceedings of the 3rd International Icst Conference on Simulation Tools and Techniques (SIMUTools'10)*, Torremolinos, Malaga, Spain, March 15–19 2010.

## Research Article

# Min- $k$ -Cut Coalition Structure Generation on Trust-Utility Relationship Graph

XiangLong Kong <sup>1</sup>, XiangRong Tong <sup>1,2</sup> and YingJie Wang <sup>1,2</sup>

<sup>1</sup>School of Computer and Control Engineering, Yantai University, Yantai 264005, China

<sup>2</sup>Yantai Key Laboratory of High-End Ocean Engineering Equipment and Intelligent Technology, Yantai University, Yantai 264005, China

Correspondence should be addressed to XiangRong Tong; txr@ytu.edu.cn

Received 28 August 2020; Revised 8 October 2020; Accepted 13 March 2021; Published 14 April 2021

Academic Editor: Maode Ma

Copyright © 2021 XiangLong Kong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Trust relationships have an important effect on coalition formation. In many real scenarios, agents usually cooperate with others in their trusted social networks to form coalitions. Therefore, the trust value between agents should constrain the utility of forming coalitions when cooperating. At the same time, most studies ignore the impact of the number of coalitions in coalition structure. In this paper, the coalition formation of trust-utility relationship in social networks is researched. Each node represents an agent, and the trust-utility networks that connect the agents constrain coalition formation. To solve the task assignment problem, this paper proposes a greedy algorithm which is based on the edge contraction. Under the premise of ensuring the agent's individually rationality, this algorithm simulates the formation process of coalitions between agents through continuous edge contraction and constrains the number of forming coalitions to  $k$  to solve the problem of coalition structure. Finally, the simulation results show that our algorithm has great scalability because of the ability of solving the coalition structure on a large-scale agent set. It can meet the growing demand for data intensive applications in the Internet of things and artificial intelligence era. The quality of the solution is much higher than other algorithms, and the running time is negligible.

## 1. Introduction

Coalition formation (CF) is an important research area in Multiagent Systems (MAS) [1]. It is known that combining several agents together can achieve goals that every individual cannot achieve alone, or get better utility [2]. Most of CF researches are based on models in the field of cooperative game theory, e.g., characteristic function game [3, 4].

Although these models can be used in the study of coalition formation, they are usually too abstract to be used in the actual cooperation scenarios. Most of them use graphs to represent the constraints of the relationship between agents [5–7], and each node in the graph represents an agent, while the edge represents the relationship between agents. Agents can form coalitions only when they are the vertices of the connected subgraphs in a graph, which is called the problem of *Graph Coalition Structure Generation* (GCSG); that is, in the process of Coalition Structure Generation (CSG), the

connectivity between agents that make up the coalition is required. Such constraints exist in most real scenarios, such as social trust constraints, physical constraints [8], and wireless communication constraints [9]. The proposed algorithm in this paper uses trust constraint; agents usually look for others with high trust and utility to cooperate, which is suitable for most scenarios. For example, in the field of privacy protection [10–14], if the trust value or utility between agents is higher, the more privacy data the agents know about each other. Then, the agents in the coalition will not disclose the information between each other and play the role of privacy protection.

In addition, most previous studies about the GCSG problem assumed that agents can form any number of coalitions and failed to consider to bind the number of coalitions in a coalition structure. However, in many real scenarios, the number of coalitions should be constrained. For example, in the field of task assignment [15, 16], agents in MAS may

need to cooperate to complete all tasks. Since the resources or capabilities of a single agent to perform tasks are limited, it is necessary to assign tasks to the coalition. This requires to constrain the number of coalitions consistent with the number of tasks, so as not to waste resources or fail to complete tasks. In addition, when performing tasks, agents usually do not cooperate with unfamiliar agents, but they can establish contact with other team members through the introduction of trusted agents.

To solve the above problems, we develop a solution for the GCSG problem in the context of task assignment and provide a model which can be used to solve the problem of a large number of agents in the real world. We use trust and utility to discuss CF, where the weight of the edge represents the strength of the cooperative relationship between agents, and its weight is represented by two tuples, which are the trust value and utility value, respectively. A connection between disconnected agents can be established through trust transfer. Based on edge contraction, this paper proposes an *Edge Contraction Coalition Formation* (ECCF) algorithm to solve the GCSG problem. This algorithm refers to the min- $k$ -cut algorithm, constrains the number of formed coalitions to  $k$ , and generates the coalition structure under the set conditions. The process of coalition formation between agents is simulated by continuous edge contraction. Each edge contraction process includes the following: (i) delete an edge from the graph, and (ii) merge two nodes previously connected by this edge. Merging two nodes is corresponded to merging two coalitions represented by these nodes, from two coalitions to one coalition, to ensure the generation of a coalition structure with exact  $k$  coalitions. The ECCF algorithm is an approximate algorithm with  $O(n^2e)$  time complexity. In addition, the ECCF algorithm is a greedy algorithm essentially, which can quickly find the feasible solution of the set coalition structure in polynomial time. Finally, ECCF can be applied in large-scale systems. It can well meet the demand of the growing data intensive applications in the Internet of things and artificial intelligence era. The simulation results show that the ECCF algorithm can always find a feasible coalition structure in a short time.

The main contributions of this paper are summarized as follows:

- (1) Trust is transitive for agents, and each agent is independent. The *six degrees of separation* and *trust transfer* are used to establish the trust relationship, so as to obtain a trust-utility relationship graph between agents, which can be applied in large-scale agent sets in most real scenarios
- (2) In order to solve the problem of task assignment, we propose a new polynomial time complexity algorithm ECCF, which is based on edge contraction and uses the min- $k$ -cut method to segment the trust-utility network graph and generate the coalition structure. Under the condition of ensuring the agent's individual rationality and generating accurate  $k$  coalitions, the coalition structure with  $k$  coalitions in the given situation is quickly generated
- (3) The simulation results demonstrate the efficiency and runtime of ECCF algorithm. The ECCF algorithm can be applied in large-scale MAS, and its solution quality is much higher than other algorithms; furthermore, its runtime is negligible

The rest of the paper is organized as follows. Section 2 discusses the relationship between our work and existing literature, and Section 3 introduces some Preliminary Definitions. Section 4 formally defines the ECCF, and Section 5 discusses our empirical evaluation. Finally, Section 6 concludes the paper.

## 2. Related Work

In this section, we will discuss the CF, graph theory techniques, and trust transitivity, respectively.

**2.1. Coalition Formation.** Many algorithms have been developed to solve traditional CSG problems. Yeh [17] first proposed the method based on Dynamic Programming (DP) to solve the CSG problem. The exact coalition structure can be obtained through DP. Rahwan et al. [4] developed an anytime algorithm based on the integer partition (IP). Michalak et al. [18] combined the IP algorithm and the DP method to improve the classical CSG algorithm ODP-IP. However, the traditional solutions can only process dozens of agents. To improve the scalability of the algorithm, Ueda et al. [19] have examined alternative function representations, which can reduce the computational complexity of the associated CF problem. Wu et al. [20, 21] sampled the coalition structure graph based on the Monte-Carlo tree search method and iteratively expanded the search tree and proposed a scalable anytime method called CSG-UCT. The CSG-UCT algorithm needs to be performed once for each iteration. The Monte-Carlo tree search process requires a large number of iterations in the process of finding the optimal solution. When the number of agents increases, the time required is greater. However, these models may not be applicable to the real world, such as the scenario in this paper that agents connect through the trust network; that is, they cannot be applied to the GCSG problem.

In this context, graph-based cooperative game has attracted the attention of researchers in various fields [22–25]. Voice et al. [26] proposed an algorithm for GCSG. However, the assumption made by voice can only be applied to the characteristic function of the independent disconnected members (IDM) property. It was hard to be applied in most real scenarios. Filippo et al. [27] proposed a branch and bound algorithm CFSS (Coalition Formation for Sparse Synergies), which can solve the problem of GCSG and provide anytime approximate solution of quality guarantees. However, all of these works assumed that any number of coalitions can be formed, and there is no constraint on the number of coalitions in the coalition structure.

**2.2. Graph Theory Techniques and Trust Transitivity.** The ECCF algorithm forms the coalition structure of agent set through edge contraction and constrains the number of coalitions. This graph theory technology and is famous for its



application in the algorithm of solving minimum cut problem [28]. Filippo et al. [27] for the first time applied edge contraction to enumerate feasible coalition structure and proposed a CFSS algorithm. At the same time, the search tree is used to represent the search space of the coalition structure. The tree generated by the search space contains all coalition structures on the graph; a CFSS algorithm was proposed, which can prune the search space and solve the coalition structure with quality guarantees at any time when it is used to solve the  $m + a$  function coalition structure. However, this algorithm is only suitable for solving the  $m + a$  function coalition structure, and it can not be used in the context of task assignment.

In the CF game, under the background of task assignment, a social network is mostly used as the constraint of cooperation between agents, which requires connectivity between agents that make up the coalition. Sless et al. [29] avoided this constraint by setting the organizer. Organizers are required to have the ability to promote the connection between unrelated agents. Organizers can add edge sets to social networks to connect unrelated agents, which depends on the social status of organizers and their own experience.

In MAS, the research on trust attracted increasing attention of researchers [30–32]. When two parties cooperate, trust is a subjective evaluation of one party to the other. Trust is not only subjective but also asymmetric and transitive. Trust is an important factor in decision-making when an agent chooses who to cooperate with to complete the task. Generally, the higher the trust level, the easier it is to generate a long-term stable cooperative relationship. Trust relationship depends on the network to spread. In the real world, the transitivity of trust can make the unfamiliar agents communicate and promote the cooperation between agents. In this way, in the process of CSG, there is a relationship between agents. At the same time, any coalition generated also has connectivity. Thus, our approach establishes a connection between disconnected agents through trust transfer and then solves the set coalition structure through edge contraction.

### 3. Preliminary Definitions

Let  $G = (A, E, \rho, \omega)$  represent a directed weighted graph, where  $A = \{a_1, a_2, \dots, a_n\}$  is a finite, non-empty set of agents. Each edge  $(a_i, a_j) \in E$  has two weights  $\rho$  and  $\omega$ , which represent trust and utility relationship between agents, respectively.  $\rho_{i,j}$  represents the trust value of agent  $a_i$  from  $a_j$ , and  $\omega_{i,j}$  represents the utility value of agent  $a_i$  from  $a_j$ . Agents without edge connection can get the trust value and utility value through transfer calculation.

Let  $f_{\rho,\omega}(a_i, a_j)$  represent the trust-utility relationship function obtained by considering the trust relationship and utility relationship between agent  $a_i$  and agent  $a_j$ , which represents the ability of smooth cooperation between agents.  $f_{\rho,\omega}(a_i, a_j)$  could be expressed by

$$f_{\rho,\omega}(a_i, a_j) = \frac{P_{i,j}\Omega_{i,j} + P_{j,i}\Omega_{j,i}}{2}. \quad (1)$$

where  $P$  and  $\Omega$  are the trust relationship and utility relationship between agents after transmission. The trust-utility relationship can be used to get the trust-utility relationship graph between agents  $G_f = (A, E, f)$ . Only the trust relationship between agents is considered when  $\omega$  is zero or the same between agents in cooperation. Only the utility relationship is considered in turn when  $\rho$  is zero or the same between agents in cooperation.

Let  $\mu(a_i, C)$  denote the utility  $a_i \in C$ ; its value is the sum of the utility among  $a_i$  and other members in  $C$ . It is shown by

$$\mu(a_i, C) = \sum_{a_j \in C} \omega(a_i, a_j). \quad (2)$$

It is denoted as  $\mu_i$ . The utility of an agent is given by other agents who have trust-utility relationship with him in the coalition. The utility of a coalition is composed of the utility of all agents in the coalition.  $V(C)$  is used to represent the value of  $C \in A$  in the coalition, and the value is defined as the values of all agents in  $C$ .  $V(C)$  is obtained by

$$V(C) = \sum_{a_i \in C} \mu(a_i, C). \quad (3)$$

Assume that the required number of tasks to be completed by agents is  $k$ ,  $0 < k < n$ .  $S_k(A)$  indicates that the agent set  $A$  is divided into a set of partitions containing  $k$  nonempty subsets. Each case in  $S_k(A)$  is called the coalition structure, which is denoted as CS.

$\gamma_{CS} : A \rightarrow CS$  represents the mapping function of the agent set to its coalition, i.e.,  $CS \in S_k(A)$ ,  $C_i \in CS$ , and  $a_i \in C_i$ , then  $\gamma_{CS}(a_i) = C_i$ . The utility of agents in different coalition structures is shown by Figure 1. The social utility of coalition structure CS is defined as  $SW(CS)$ , which is shown by

$$SW(CS) = \sum_{C \in CS} V(C). \quad (4)$$

*Definition 1 (block).* Given a coalition structure  $CS = \{C_1, C_2, \dots, C_k\}$ ,  $\exists B \in A$ , if  $\forall a_i, a_j \in B$ ,  $\forall a_i' \in \gamma_{CS}(a_i)$ ,  $\forall a_j' \in \gamma_{CS}(a_j)$  satisfied  $f_{\rho,\omega}(a_i, a_j) > f_{\rho,\omega}(a_i, a_i')$  and  $f_{\rho,\omega}(a_i, a_j) > f_{\rho,\omega}(a_j, a_j')$ ; then, a coalition  $B$  is *block*.

According to Definition 1, for any two agents in a coalition, if there is another coalition  $B$  containing the two agents and the rewards of agent in  $B$  are greater than the rewards of the current coalition, that is, these two agents tend to leave the current coalition, and  $B$  is called *block*. Clearly, *block B* is an unstable factor for CS.

From Figure 2, we can see that if there is a coalition structure  $CS = \{\{a_1, a_2, a_3\}, \{a_4\}, \{a_5, a_6\}\}$ , a coalition  $B = \{a_4, a_5\}$ . According to Definition 1, we can see that coalition  $B$  is a *block*.

Note that even if the optimal CS is obtained, there may be *blocks* in the solution. The movement of agents in the original coalition may form any number of coalitions. Clearly, we must exclude this situation in the model. Therefore, a  $k$ -*block*

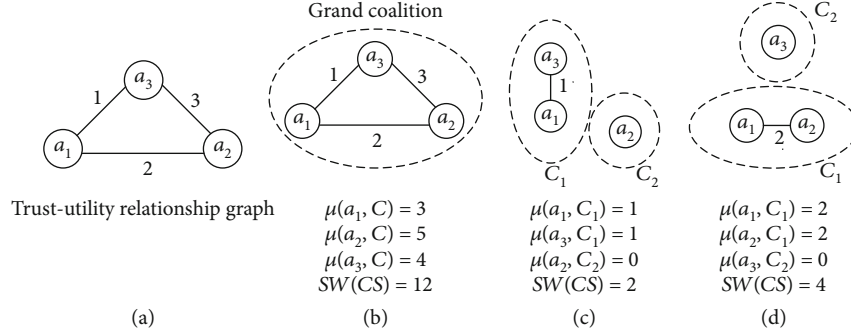


FIGURE 1: The utility of agents in different coalition structures.

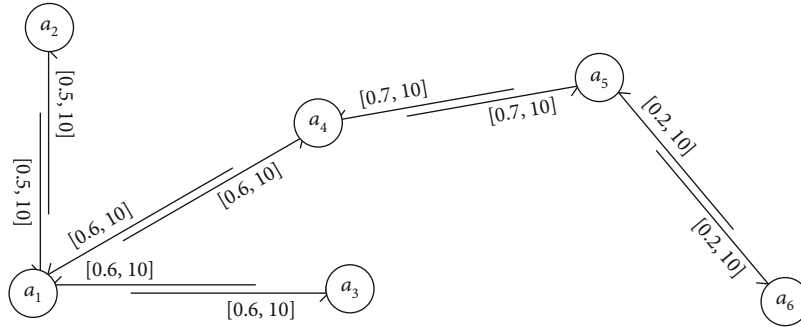


FIGURE 2: Trust network.

*coalition* must be defined so that the final coalition structure contains exactly  $k$  coalitions.

**Definition 2 ( $k$ -block coalition).** Given a coalition structure  $CS = \{C_1, C_2, \dots, C_k\}$ ,  $\exists B \in A$ , if  $\forall a_i, a_j \in B$ ,  $\forall a_i' \in \gamma_{CS}(a_i)$ ,  $\forall a_j' \in \gamma_{CS}(a_j)$  can satisfy  $f_{\rho, \omega}(a_i, a_j) > f_{\rho, \omega}(a_i, a_i')$  and  $f_{\rho, \omega}(a_i, a_j) > f_{\rho, \omega}(a_j, a_j')$ . At the same time,  $\exists C_m \in CS$ ,  $C_m \in B$ ; then a coalition  $B$  is  $k$ -block.

In Figure 2, it is known that there are coalition structure  $CS = \{\{a_1, a_2, a_3\}, \{a_4\}, \{a_5, a_6\}\}$  and *block*  $B = \{a_4, a_5\}$ ,  $\exists C_m = \{a_4\}$ , making  $C_m \in CS$  and  $C_m \in B$ . According to Definition 2,  $B$  is a 3-block coalition.

Definition 2 is an extension of Definition 1. Coalition structure after any agent movement must contain exact  $k$  coalitions. Therefore, it is necessary to ensure that agents tend to join the existing coalition; that is, agents with movement tendency in the coalition structure will only join the current coalition to another formed coalition and will not form a new coalition. In addition, it is necessary to ensure that there is only one coalition. If there are multiple such coalitions, the number of coalitions in the generated coalition structure will be less than  $k$ . If there is less than one such coalition, the final coalition structure will contain  $k + 1$  coalitions. Only if there is one coalition, the movement of agents will guarantee the coalition structure to contain accurate  $k$  coalitions.

**Definition 3 (individual rationality).** Given a coalition structure  $CS \in S_k(A)$ , if each agent gets at least the same utility

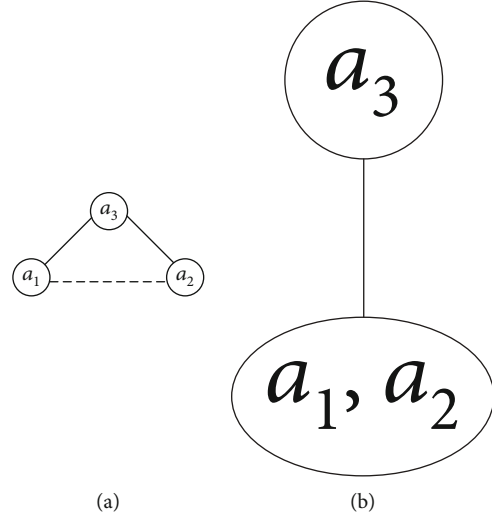


FIGURE 3: Example of an edge contraction.

from  $CS$  to its own, then  $CS$  satisfies individual rationality, i.e.,  $\forall a \in C_i$ ,  $i \in (1, \dots, k)$ ,  $\mu(a_i, C_i) \geq \mu(a, \{a\})$ .

**Definition 4 (trust transfer).** Given the trust network, trust can be transferred. As shown in Figure 3, assuming that agent  $a_1$  trusts agent  $a_2$  and agent  $a_2$  trusts agent  $a_4$ , it can be determined that agent  $a_1$  may trust agent  $a_4$ . There are two ways for *trust transfer*: (i) take the minimum value, and (ii) take the weighted average value. In a large network system, the

**Input:** Trust network  $G = (A, E, \rho, \omega)$ , number of coalitions parameter  $k$  from CS  
**Output:** Coalition structure CS

- 1: Calculate the trust and utility after transfer
- 2: For all agent  $a_i, a_j \in A$
- 3:  $P_{i,j} = \min \{\rho_{ij}^{(k)}, k = 2, 3, \dots, 5\}$ .
- 4:  $\Omega_{i,j} = \min \{\omega_{ij}^{(k)}, k = 2, 3, \dots, 5\}$ .
- 5: End for
- 6: Calculate the trust and utility relationship and get the trust-utility relationship graph
- 7: For each edge  $(a_i, a_j) \in E$
- 8:  $f_{\rho,\omega}(a_i, a_j) = (P_{i,j}\Omega_{i,j} + P_{j,i}\Omega_{j,i})/2$ .
- 9: End for
- 10: Perform the edge contraction operation and generate the image segmentation coalition
- 11: For all Agent  $a_i, a_j \in A$
- 12: For  $i = 1 \dots n-k$
- 13:  $(a_i, a_j) \leftarrow \max \{f_{\rho,\omega}(a_i, a_j)\}$
- 14:  $Contract(a_i, a_j)$
- 15: For all  $a \in A$
- 16:  $f_{\rho,\omega}(a, m) = f_{\rho,\omega}(a_i, a) + f_{\rho,\omega}(a_j, a)$
- 17: End for
- 18: End for
- 19: For each edge  $(a_i, a_j) \in E$
- 20: Remove  $(a_i, a_j)$
- 21: End for
- 22: End for
- 23: **return**CS

ALGORITHM 1: ECCF

weighted average will make the trust value tend to zero, so we take the minimum value in the transfer process, that is,

$$P_{i,j} = \min \left\{ \rho_{ij}^{(k)}, k = 2, 3, \dots, 5 \right\}. \quad (5)$$

In the same way, the utility between agents is expressed as follows:

$$\Omega_{i,j} = \min \left\{ \omega_{ij}^{(k)}, k = 2, 3, \dots, 5 \right\}. \quad (6)$$

According to the trust and utility function, trust and utility relationship between the agents can be obtained after coherently considering the trust relationship and utility relationship, so as to obtain the trust-utility relationship graph  $G_{f=(A,E,f)}$  between the agents, where  $A$  represents the agent set, the members are generally  $a_i$ ,  $E$  is the edge set, and each edge  $e$  has weight  $f$  that represents the trust-utility relationship value between the agents obtained through the trust-utility function.

The trust network in Figure 2 can provide an example of trust transfer. According to the equations, we can get  $P_{15} = 0.6$ ,  $\Omega_{15} = 10$ , and  $f_{15} = 6$ .

*Definition 5 (edge contraction).* Given the trust-utility graph  $G_f = (A, E, f)$ ,  $(a_i, a_j) \in E$ , the result of contraction for  $e$  is the graph  $G'_f$  obtained by deleting  $e$  and corresponding vertices  $a_i$  and  $a_j$  and, at the same time, adding new vertices  $m$

$= a_i \cup a_j$ . In addition, each edge connecting with  $a_i$  or  $a_j$  in  $G_f$  will be connected to  $m$  in  $G'_f$ , then merging the parallel edges (i.e., the edges connected to the combined two vertices) and the corresponding weights of parallel edges (i.e., the trust-utility relationship).

Intuitively, each edge contraction merges two vertices, corresponding to merging two coalitions represented by these nodes. Note that the order of the contraction of edges can be in any given order and get the same result, i.e., contracting  $e$  first, then contracting  $e'$  which produces the same graph to contracting  $e'$  then contracting  $e$ . The resulting graph after contracting represents a feasible coalition structure, where the coalition corresponds to the vertices of the graph.

When performing edge contraction, the number of subsets of edges in the graph is greater than the number of feasible coalition structures. This redundancy is due to the contraction of any two or three edges resulting in the same coalition structure. In this paper, the edge with the largest weight is firstly selected for contraction operation. When the weight of the edge is different, there will be no redundant operation. When there are multiple edges with the largest weight, any edge with the largest weight will be contracted randomly, and the edge with the largest weight will be also selected for operation in the next contraction. So there will be no redundant coalition structure.

Figure 3 shows the contraction of the edge  $(a_1, a_2)$  in the trust-utility graph  $G_f = (A, E, f)$ , which results in a new

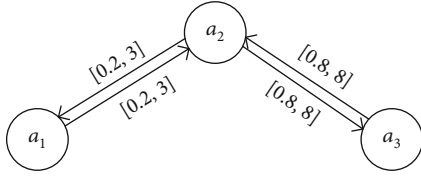


FIGURE 4: Simple trust network.

TABLE 1: Initialization trust-utility.

	$a_1$	$a_2$	$a_3$
$a_1$		[0.2, 3]	
$a_2$	[0.2, 3]		[0.8, 8]
$a_3$		[0.8, 8]	

TABLE 2: Postprocessing trust-utility.

	$a_1$	$a_2$	$a_3$
$a_1$		[0.2, 3]	[0.2, 3]
$a_2$	[0.2, 3]		[0.8, 8]
$a_3$	[0.2, 3]	[0.8, 8]	

vertex  $m = a_i \cup a_j$  connected to the vertex  $a_3$  and merges the parallel edges  $(a_1, a_3)$  and  $(a_2, a_3)$ .

#### 4. Maximize the Utility of Coalition Structure

This section mainly discusses the problem of the coalition structure that tries to maximize the utility given a trust-utility relationship graph. Agents in the graph are reasonably assigned to coalitions to maximize the utility of coalition structure.

*Definition 6 ( $k$ -cut).*  $\text{Cut}_k(C_1, C_2, \dots, C_k)$  means that agents in the trust-utility relationship graph are divided into  $k$  disjoint parts according to the trust-utility relationship, where each part forms a coalition, and  $k$  coalitions are formed.  $\text{Cut}_k(C_1, C_2, \dots, C_k)$  is called  $k$ -cut.

Brânzei and Larson [33] first noted the relationship between the social welfare maximization coalition structure and the min- $k$ -cut of the graph; that is, the coalition structure with the number  $k$  of maximizing utility is the partition of the min- $k$ -cut graph. This is reasonable because the sum of the weights of all edges is constant. By minimizing the utility of edges outside the coalition, the utility within the coalition can be maximized. Therefore, the min- $k$ -cut algorithm can be used to find the maximum utility coalition structure. Based on the above researches, Sless et al. [29] proved that min- $k$ -cut is in  $P$  for nearly positive graphs and a fixed  $k$ . However, the algorithm has high complexity and can only be applied in nearly positive graphs, while our algorithm could solve approximate solutions with low complexity.

Based on the concept of edge contraction in graph theory, an approximate algorithm is proposed. Given  $k$ , the edge contraction method is used to perform  $n - k$  number of edge

TABLE 3: Trust-utility function postrelationship.

	$a_1$	$a_2$	$a_3$
$a_1$		0.6	0.6
$a_2$	0.6		6.4
$a_3$	0.6	6.4	

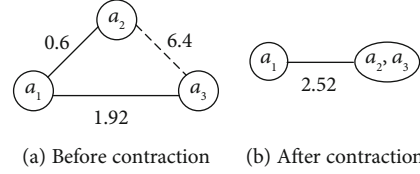


FIGURE 5: Trust-utility relationship graph.

contraction. Therefore, it is ended with  $k$  nodes. Intuitively, each node after the contraction is a coalition. Therefore, though deleting the remaining edge, it can form  $k$ -connected subgraphs, i.e.,  $k$  coalitions.

In each iterative contraction, the edge with the largest weight is selected in the trust-utility graph, and define operation  $\text{Contract}(a_i, a_j)$ .  $\text{Contract}(a_i, a_j)$  means to delete points  $a_i$  and  $a_j$  and edge  $(a_i, a_j)$ , and add a new nodes  $m = a_i \cup a_j$ , for any  $a \in A$ ; there is  $f_{\rho, \omega}(a, m) = f_{\rho, \omega}(m, a) = f_{\rho, \omega}(a_i, a) + f_{\rho, \omega}(a_j, a)$ .

*4.1. ECCF.* In this section, we give a comprehensive consideration for the trust relationship and utility relationship between the agents in the trust network  $G$ ; then  $P_{i,j}$  and  $\omega_{i,j}$  are calculated according to the trust network  $G$ . Then, the trust-utility relationship between the agents is calculated according to the values of  $P_{i,j}$  and  $\omega_{i,j}$ . Moreover, the contract of edge is performed according to  $f$  to complete the image cutting. Finally, coalition structure is obtained. The specific steps are as shown by Algorithm 1.

Given the trust network  $G = (A, E, \rho, \omega)$ , the number of vertices is  $n$ , the  $P_{i,j}$ ,  $\Omega_{i,j}$ , and  $f_{i,j}(a_i, a_j)$  are the trust relationship, the utility relationship, and the trust-utility relationship after comprehensive consideration, respectively. Firstly, input the trust network and calculate the trust value and utility value between agents according to the trust transfer formula. Secondly, according to the results of the first step, the trust and utility are considered synthetically, and the trust-utility relationship between agents is obtained by using the trust-utility relationship function. Thirdly, according to the trust-utility relationship size, the proposed algorithm is performed, and according to the operation of edge contraction,  $n - k$  times edge contraction is performed, then the remaining weight is deleted, and the image is cut. The agent set is divided into  $k$  disjoint sets, i.e.,  $k$  coalitions, and the final CS is obtained.

In each iteration of ECCF algorithm, the edge with the largest trust-utility relationship between agents is selected to contract. The CS obtained by the ECCF algorithm is an approximate solution, which is essentially a greedy algorithm. The ECCF algorithm does not pursue the optimal

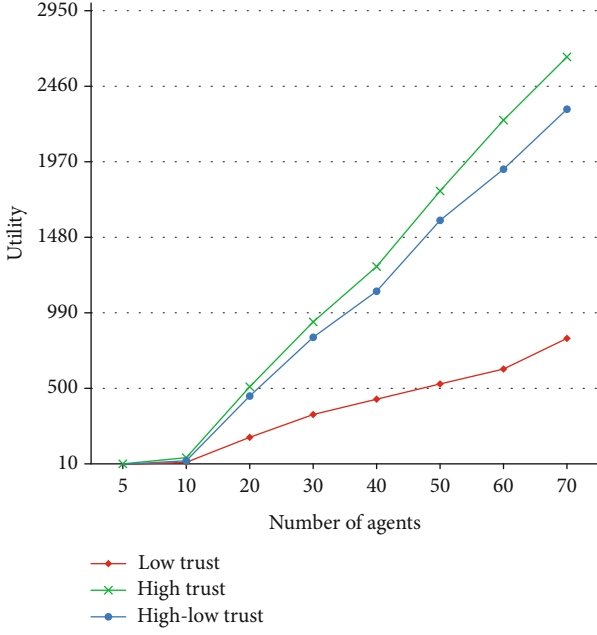


FIGURE 6: The influence of trust on utility.

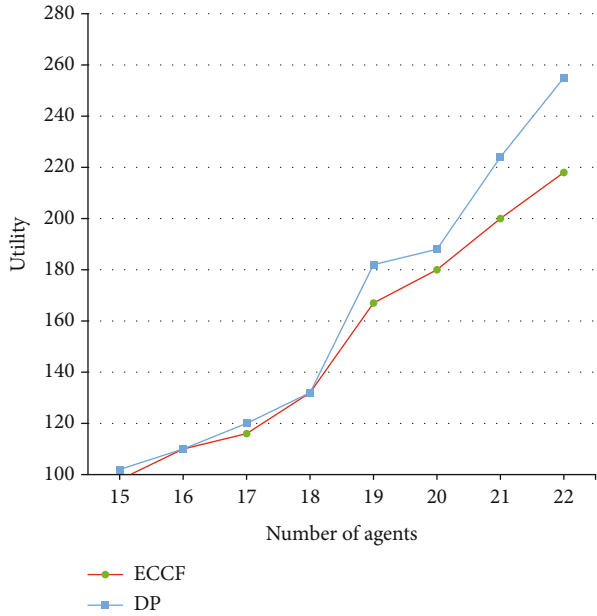


FIGURE 7: The influence of  $n$  on utility.

solution, and backtrack only hopes to get a more satisfactory solution quickly and makes the optimal selection based on the current situation, without considering all possible overall situations. It could save a lot of time to find the best solution.

**4.1.1. Time Complexity Analysis.** In the ECCF algorithm, the first step uses the trust transfer formula to obtain the trust value and utility value between agents. The time complexity is  $O(e)$ . In the second step, the trust-utility function formula is used to coherently consider calculate the trust-utility relationship between agents. The time complexity is  $O(n^2)$ . In the third step, the *for* loop in the algorithm performs the con-

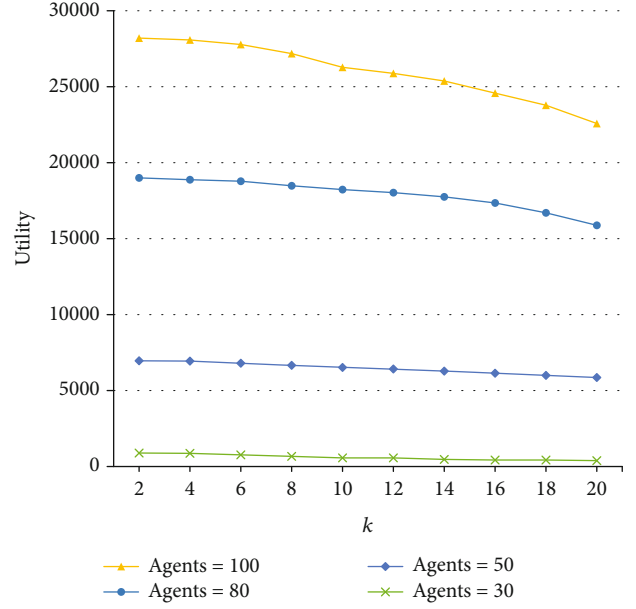


FIGURE 8: The influence of  $n$  and  $k$  on utility.

tract operation  $n - k$  times. In each contract iteration, the ECCF algorithm is used to traverse all edges in the graph and find the largest weight edge. The time complexity is  $O(e)$ . Finally, merge parallel edges and the corresponding weights of parallel edges. The complexity is at most  $O(2n)$ , so the complexity of the third step is  $O(n^2 e)$ . The total time complexity of the algorithm is  $O(n^2 e)$ .

**4.1.2. Example of Algorithm Solving Process.** Figure 4 shows an ECCF algorithm for a given simple trust network.

According to the ECCF algorithm, the solution process is explained as follows:

In the first step, the settings are initialized, which are shown in Table 1.

In the second step, according to the transfer formula, the trust and utility after processing are shown in Table 2.

In the third step, we use the trust-utility function  $f_{\rho,\omega}(a_i, a_j) = (P_{i,j}\Omega_{i,j} + P_{j,i}\Omega_{j,i})/2$  to get the relationship shown in Table 3.

The ECCF algorithm coherently considers the impact of the trust relationship and utility relationship between agents, which is consistent with the real social situation, and obtains the trust-utility relationship graph shown by Figure 5.

In the fourth step, based on the ECCF algorithm, let  $k = 2$ ; we contract the maximum edge (i.e., 6.4) and delete the remaining trust-utility relationship (i.e., 2.52) to get the coalition structure  $\{\{a_1\}, \{a_2, a_3\}\}$ ; then the utility is 12.8.

## 5. Simulation Experiments and Analysis

**5.1. Experimental Environment and Comparison Algorithm.** In this section, the simulation experiments are designed to demonstrate the effectiveness of the ECCF algorithm. We use a random graph for experimental data. The experimental environment is Windows 7 with 64 bit operating system,

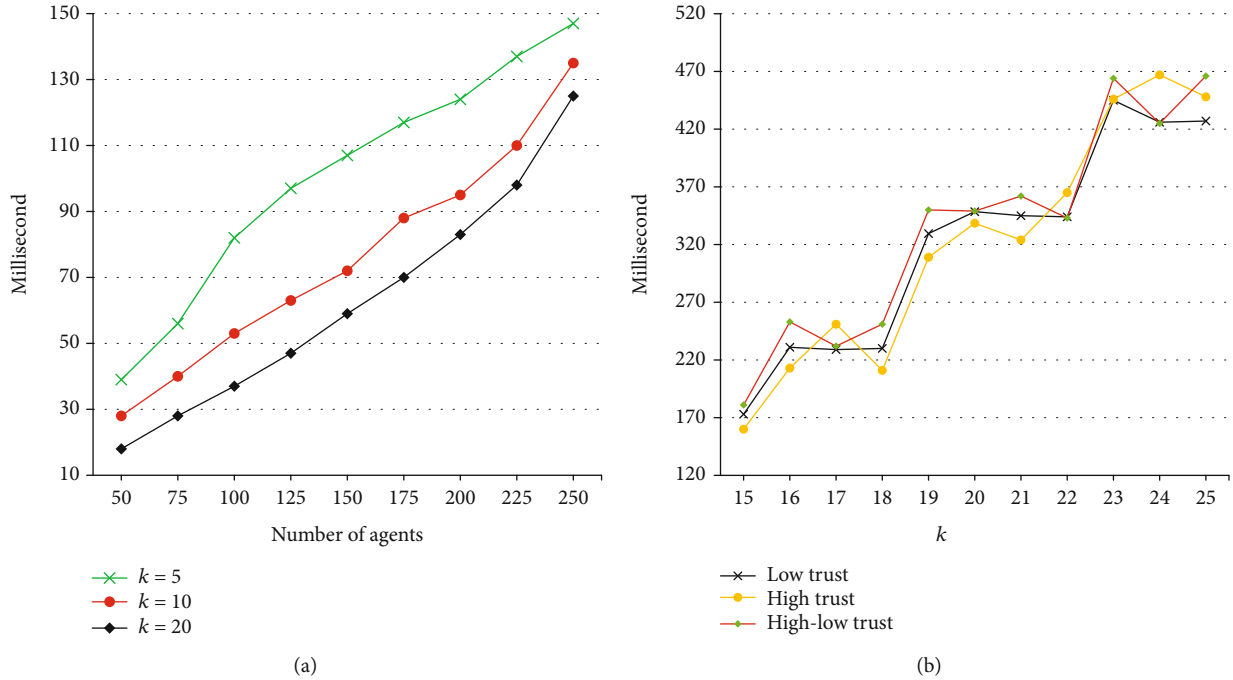


FIGURE 9: Relationship between the number of agents and runtime.

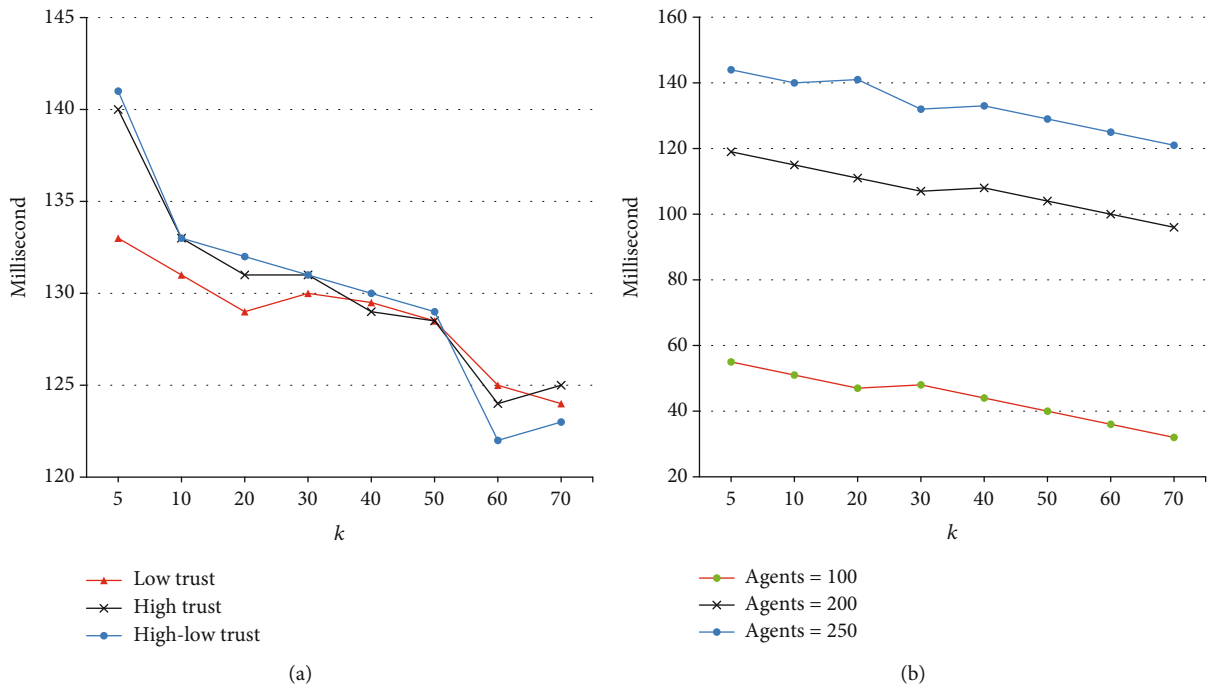


FIGURE 10: Relationship between the number of coalitions and runtime.

8GB memory, 3.2GHz main frequency, and i5-6500 processor. The programming language is java, which runs on eclipse.

We chose the classic CSG algorithm as the comparison algorithm, with the DP algorithm and ODP-IP algorithm. In addition, we also implement the efficient algorithm of Saran and Vazirani [34] which is called SV (Saran and Vazirani) algorithm. The SV algorithm is an approximate algorithm for solving min- $k$ -cut problem. The algorithm

recursively selects the edge with the smallest weight to cut until the graph is divided into  $k$  parts.

We use utility and runtime as indexes to evaluate the coalition structure. On the one hand, we study the influence of trust, the number of agents is  $n$ , and the number of coalitions is  $k$  on the utility of CS. On another hand, we study the influence of the number of agents  $n$  and the number of coalitions  $k$  on runtime.

5.2. Experimental Results and Performance Analysis

5.2.1. *Effect of  $\rho$ ,  $n$ ,  $k$  on Utility.* This section mainly discusses the influence of trust, the number of agents  $n$ , and the number of coalitions  $k$  for the ECCF algorithm on the utility of the coalition structure. Among them,  $0 < \rho < 0.5$  is a low trust relationship,  $0.5 < \rho < 1$  is a high trust relationship, and  $0 < \rho < 1$  is a high-low trust relationship.

Figure 6 shows the relationship between trust and utility among agents in the trust-utility graph of the ECCF algorithm, where the  $x$ -coordinate represents the number of agents and the  $y$ -coordinate represents the utility of the ECCF algorithm. It is easy to know that trust and utility between agents show a linear relationship. The utility generated by the agent increases with the increase of the trust relationship. The greater the trust value between agents, the greater the utility of generating the coalition structure. In addition, the utility of the coalition structure is related to the number of agents. Figure 7 shows the relationship between agents and the utility of the coalition structure in ECCF algorithm. In general, the utility of the coalition structure increases with the number of agents. Through the analysis of Figure 7, it is easy to know that the utilities of the ECCF algorithm and the DP algorithm increase with the number of agents; the increase range is basically the same. Figure 8 shows the relationship among  $k$  coalitions in CS and the utility of coalition structure. The  $x$ -coordinate shows the number of coalitions in CS, and the  $y$ -coordinate shows the utility of CS. With the increase of  $k$ , the utility of ECCF algorithm decreases correspondingly, because the number of edges that need to be contracted decreases correspondingly and the number of edges that need to be deleted finally increases, so the final remaining utility decreases correspondingly.

5.2.2. *Effect of  $n$  and  $k$  on Runtime.* This section mainly discusses the effect of the number of agents  $n$  and the number of tasks  $k$  on runtime of the generated coalition structure. Through the trust-utility relationship graph, the coalition structure of different agent sets is solved, and the runtime required to generate the coalition structure is recorded. The results are shown in Figure 9. The  $x$ -coordinate is the number of agents, and the  $y$ -coordinate is the runtime of the Coalition Structure Generation; the unit is millisecond.

Through the analysis of Figure 9, it is easy to know that with the increase of the number of agents, the runtime of ECCF algorithm increases correspondingly and the runtime is not affected by the trust relationship. According to Figure 10, it is easy to know that with the increase of the number of  $k$ , the runtime of the ECCF algorithm decreases correspondingly, because the number of edges that need to be contracted decreases correspondingly, so the calculation amount decreases correspondingly. Similarly, it is not affected by the trust relationship; that is, the trust between agents only affects the utility of the formed coalition and does not affect the runtime of the coalition.

5.2.3. *Algorithm Runtime Comparison.* This section mainly discusses the changes on the runtime of the ECCF algorithm,

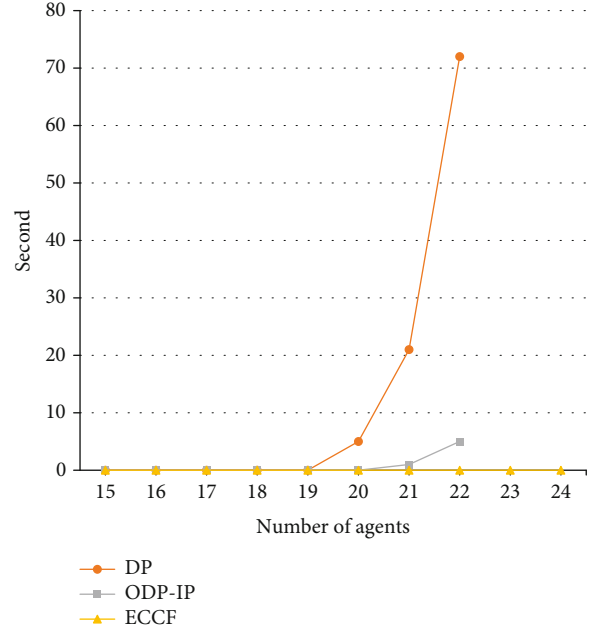


FIGURE 11: Runtime comparison of three algorithms.

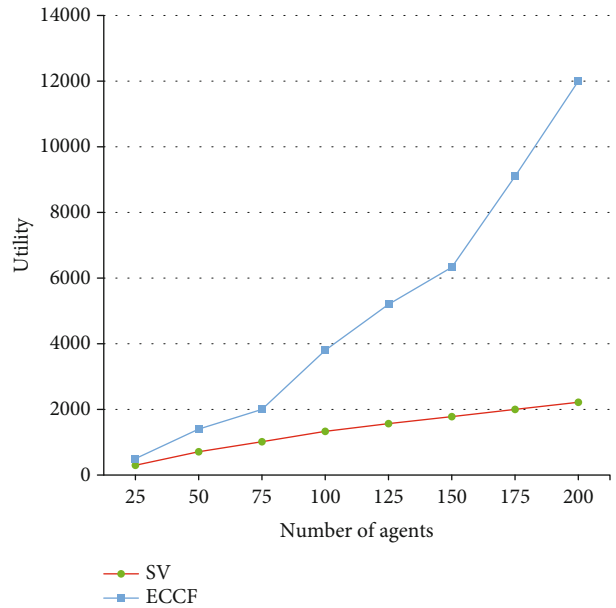


FIGURE 12: Utility comparison of two algorithms.

DP algorithm, and ODP-IP algorithm with the increase of the number of agents. The comparison results are shown in Figure 11. The  $x$ -coordinate represents the number of agents, and the  $y$ -coordinate represents the runtime.

In Figure 11, it can be seen that the runtime of the ECCF algorithm is far less than those of the DP algorithm and ODP-IP algorithm. Since the DP algorithm gradually decomposes the current solution into an optimal partition and records the intermediate solution to find the optimal coalition structure, the running time is relatively high. In addition, with the increase of the number of agents, the effect of time comparison is more obvious.

**5.2.4. Algorithm Utility Comparison.** From Figure 12, we can see that the utility of the coalition structure obtained by the ECCF algorithm is greater than that by the SV algorithm. In addition, with the increase of the number of agents, the effect of utility is more obvious. Because the SV algorithm recursively deletes the edge of the minimum trust-utility relationship until  $k$  coalitions are obtained. Then, the minimum trust-utility relationship among agents in the generated  $k$  coalitions will also be deleted, which will cause the waste of resources. The utility generated by the SV algorithm is less than that by the ECCF algorithm in most cases.

## 6. Conclusions

In this paper, we use a given number of coalitions to analyze the trust and utility of CSG. This paper proposed an ECCF algorithm based on edge contraction, which can solve the GCSG problem well. Firstly, trust is transitive, and the trust and utility are transferred according to the trust transfer. Then, the image of the transferred trust network is segmented to obtain a feasible coalition structure. Finally, the ECCF algorithm can be applied to large-scale systems, and the feasible solution can be obtained quickly.

In future works, we will analyze the search problem of the coalition core in the coalition structure. Another potential research direction is the study of the number of agents in coalition. In order to remove the coalition with a small number of agents in the solution, the size of the coalition can be limited.

## Data Availability

Data are available on request through contacting long984384142@163.com.

## Conflicts of Interest

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

## Acknowledgments

The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article. This research was financially supported by the National Natural Science Foundation of China under Grant No. 62072392 and the China Postdoctoral Science Foundation under Grant No. 2019T120732 and Grant No. 2017M622691.

## References

- [1] M. Rogna, "Coalition formation and bargaining protocols: a review of the literature," *Journal of Economic Surveys*, vol. 33, no. 1, pp. 26–251, 2019.
- [2] E. Elkind, T. Rahwan, and N. R. Jennings, "Computational coalition formation," *Annals of Nuclear Energy*, vol. 6, no. 3, pp. 329–380, 2013.
- [3] T. Sandholm, K. Larson, M. Andersson, O. Shehory, and F. Tohmé, "Coalition structure generation with worst case guarantees," *Artificial Intelligence*, vol. 111, no. 1-2, pp. 209–238, 1999.
- [4] T. Rahwan, S. D. Ramchurn, N. R. Jennings, and A. Giovannucci, "An anytime algorithm for optimal coalition structure generation," *Journal of Artificial Intelligence Research*, vol. 34, no. 1, pp. 521–567, 2009.
- [5] R. B. Myerson, "Graphs and cooperation in games," *Mathematics of Operations Research*, vol. 2, no. 3, pp. 225–229, 1977.
- [6] Y. Huang, M. Chen, Z. Cai, X. Guan, T. Ohtsuki, and Y. Zhang, "Graph theory based capacity analysis for vehicular ad hoc networks," in *IEEE Global Telecommunications Conference (GLOBECOM 2015)*, pp. 1–5, San Diego, CA, USA, December 2015.
- [7] D. Miao, Z. Cai, J. Yu, and Y. Li, "Triangle edge deletion on planar glasses-free RGB-digraphs," *Theoretical Computer Science*, vol. 788, pp. 2–11, 2019.
- [8] 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.
- [9] K. Bakht, F. Jameel, Z. Ali et al., "Power allocation and user assignment scheme for beyond 5g heterogeneous networks," *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 2472783, 11 pages, 2019.
- [10] 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.
- [11] 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*, 2021.
- [12] 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.
- [13] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, Dallas, TX, USA, July 2019.
- [14] 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.
- [15] B. Huang, Z. Li, Y. Xu et al., "Deep reinforcement learning for performance-aware adaptive resource allocation in mobile edge computing," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 2765491, 17 pages, 2020.
- [16] 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.
- [17] D. Y. Yeh, "A dynamic programming approach to the complete set partitioning problem," *BIT Numerical Mathematics*, vol. 26, no. 4, pp. 467–474, 1986.
- [18] T. P. Michalak, T. Rahwan, E. Elkind, M. Wooldridge, and N. R. Jennings, "A hybrid exact algorithm for complete set partitioning," *Artificial Intelligence*, vol. 230, pp. 14–50, 2016.
- [19] S. Ueda, A. Iwasaki, V. Conitzer, N. Ohta, Y. Sakurai, and M. Yokoo, "Coalition structure generation in cooperative games with compact representations," *Autonomous Agents and Multi-Agent Systems*, vol. 32, no. 4, pp. 503–533, 2018.



- [20] F. Wu and S. D. Ramchurn, "Monte-Carlo tree search for scalable coalition formation," in *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, pp. 407–413, Yokohama, Japan, July 2020.
- [21] S. Arib, S. Aknine, and T. Cazenave, "Nested Monte-Carlo search for multi-agent coalitions mechanism with constraints," in *International Workshop on Multi-disciplinary Trends in Artificial Intelligence*, pp. 80–88, Kuala Lumpur, Malaysia, November 2015.
- [22] Z. Cai, G. Lin, and G. Xue, "Improved approximation algorithms for the capacitated multicast routing problem," in *Proceedings of the 11th International Computing and Combinatorics Conference (COCOON 2005)*, pp. 136–145, Kunming, China, August 2005.
- [23] Z. Cai, Z.-Z. Chen, and G. Lin, "A 3.4713-approximation algorithm for the capacitated multicast tree routing problem," *Theoretical Computer Science*, vol. 410, no. 52, pp. 5415–5424, 2009.
- [24] Z. Cai, R. Goebel, and G. Lin, "Size-constrained tree partitioning: a story on approximation algorithm design for the multicast k-tree routing problem," in *Combinatorial Optimization and Applications. COCOA 2009. Lecture Notes in Computer Science*, vol. 5573pp. 363–374, Springer, Berlin, Heidelberg.
- [25] Z. Cai, R. Goebel, G. Lin et al., "Size-constrained tree partitioning: approximating the multicast k-tree routing problem," *Theoretical Computer Science*, vol. 412, no. 3, pp. 240–245, 2011.
- [26] T. D. Voice, S. D. Ramchurn, and N. R. Jennings, "On coalition formation with sparse synergies," in *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, pp. 223–230, Valencia, Spain, June 2012.
- [27] B. Filippo, F. Alessandro, C. Jesus, J. Rodríguez-Aguilar, and S. D. Ramchurn, "Algorithms for graph-constrained coalition formation in the real world," *ACM Transactions on Intelligent Systems and Technology*, vol. 8, no. 4, pp. 1–24, 2017.
- [28] M. Ghaffari, K. Nowicki, and M. Thorup, "Faster algorithms for edge connectivity via random 2-out contractions," in *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms*, pp. 1260–1279, Salt Lake City, UT, USA, January 2020.
- [29] L. Sless, N. Hazon, S. Kraus, and M. Wooldridge, "Forming  $k$  coalitions and facilitating relationships in social networks," *Artificial Intelligence*, vol. 259, pp. 217–245, 2018.
- [30] G. Han, J. Du, C. Lin, H. Wu, and M. Guizani, "An energy-balanced trust cloud migration scheme for underwater acoustic sensor networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 1636–1649, 2020.
- [31] C. Koliass, W. Meng, G. Kambourakis, and J. Chen, "Security, privacy, and trust on internet of things," *Wireless Communications and Mobile Computing*, vol. 2019, 3 pages, 2019.
- [32] Y. Wang, G. Yin, Z. Cai, Y. Dong, and H. Dong, "A trust-based probabilistic recommendation model for social networks," *Journal Network Computer Applications*, vol. 55, pp. 59–67, 2015.
- [33] S. Brânzei and K. Larson, "Coalitional affinity games," in *Proceedings of the 8th International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2009)*, pp. 1319–1320, Budapest, Hungary, May 2009.
- [34] H. Saran and V. V. Vijay Vazirani, "Finding  $k$  cuts within twice the optimal," *SIAM Journal Computing*, vol. 24, no. 1, pp. 101–108, 1995.

## Research Article

# SDRM-LDP: A Recommendation Model Based on Local Differential Privacy

Gesu Li <sup>1</sup>, Guisheng Yin, <sup>1</sup> Jishen Yang <sup>2</sup>, and Fukun Chen<sup>1</sup>

<sup>1</sup>College of Computer Science and Technology, Harbin Engineering University, Heilongjiang, China

<sup>2</sup>Department of Computer Science, Georgia State University, Georgia, USA

Correspondence should be addressed to Gesu Li; [lgs7788@hrbeu.edu.cn](mailto:lgs7788@hrbeu.edu.cn)

Received 4 December 2020; Revised 7 January 2021; Accepted 19 February 2021; Published 18 March 2021

Academic Editor: Jinbao Wang

Copyright © 2021 Gesu Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The development of 5G technology has driven the rise of e-commerce, social networking, and the Internet of Things. Under the high-speed transmission, the data volume increases, and the user demand also changes. Personalized customization has become the mainstream trend of network development. However, as the speed of the Internet increases, a series of problems also arise. The increase in data volume results in a reduction of bandwidth, a growth of the central processor's pressure, and a higher risk of data leakage. A search system and a recommendation platform are the tools to improve people's search efficiency. However, providing personalized recommendations to different users according to their needs is still an urgent problem. Simultaneously, the big data volume means that attackers can also get more information. They can use background knowledge and various reasoning methods to deduce the user's private information using nonprivate items. In this paper, the solutions to safe and reliable recommendation services are the main problem explored. Based on this idea, this paper proposed short-term dynamic recommendation model based on local differential privacy (SDRM-LDP). This model uses a small amount of user information to construct short-term user preference behaviors and provides recommendations for users based on the similarity between items. We consider that an attacker uses nonprivate items to derive privacy items. Therefore, we randomly replace the original data in the same category. At the same time, the local differential privacy (LDP) is added to the privacy item query to make the private data available and protect the privacy information. In this paper, two real-world datasets, ML-100K and ML-10M, are used for experiments. Experimental results show that the results of SDRM-LDP are superior to other models.

## 1. Introduction

The fifth-generation mobile communication system (5G) supports enhanced mobile broadband (eMBB), mMTC, and uRLLC. In high-speed transmission, the user's personalized needs are enhanced [1]. Users want to be able to get their personal customization in the shortest time. At the same time, as the mainstream of the network platform, e-commerce, social networks, and especially the Internet of Things are inseparable from personalized recommendation services. Now, the development trend of the Internet of Things is the combination of e-commerce. For example, in the scenario of driving a car [2], the navigation recommends the relevant destination to the user according to the time and the user's past behavior and habits. Also, smart appliances are connected to the mobile phone terminal and set automatically according to

the user's past behavior. Therefore, when users open the corresponding APP, relevant products are recommended to users according to their behavior habits and needs. Inevitably, the emergence of 5G accelerates the development speed of personalized recommendation. The traditional recommendation system is a mostly static recommendation. It gathers users' information and computes on a reliable device and finally outputs the recommended lists to users. There is a lag in the recommended results. As the speed of the network increases, users get more information. As a result, as the amount of information increases, users' needs change. Some regional businesses collect all the data in the cloud for processing, which commonly wastes bandwidth and increases transmission latency. After the user buys an item, the platform recommends more related items to the user, but at this time, the items are no longer needed by the user. Therefore,

transmission delay and data quantity determine the processing mode of 5G service. That is, computing is not all in the core network. Marginal computation allows the business to process a part of the data on the client side, such as data processing and encryption. By this method, the delay and load are reduced, and the efficient processing capacity is realized. This method reduces the delay and overload and improves the computing performance. Based on the background of 5G, this paper proposes a short-term dynamic personalized recommendation model on platforms such as the Internet of Things and e-commerce. The model uses marginal calculation to process a part of the user's data at the terminal to reduce bandwidth waste and latency and add the privacy protection mechanism. The privacy protection mechanism can protect the user's personal privacy and data by preventing attackers from using background knowledge and statistical knowledge to reversely deduce user privacy information.

Collaborative filtering is the main recommendation algorithm in the market. The following problems exist. (1) Due to the large data volume, the transmission process is easy to cause data leakage and loss. (2) The data is static, and there is hysteresis in the process of recommendation. (3) As the data is processed centrally, the computation speed is low for big data processing. Holbrook and Schindler [3] proposed in 1989 that users' preferences change over time. The user's preference is related to age, gender, etc. However, the traditional recommendation system mainly considers the similarity between users and items and ignores the users' preferences. With the development of the network, the promotion of intelligence, personalized service is the future mainstream. Using similarity to build a model is no longer enough to satisfy users. Therefore, we need to explore user preferences, to provide users with personalized customized services and drive the development of e-commerce and the Internet of Things.

Of course, there are always two sides to everything. 5G brings fast network experience and data transmission speed. More data are uploaded to the Internet. Attackers also can get all kinds of data more easily [4]. For example, in smart home appliances, users' daily behavior data are uploaded to the server center, which is likely to cause data leakage in the process of transmission [5]. Some attackers can use the existing background knowledge combined with statistical knowledge to infer the private information and find out the target user. For the network threats, how to provide users with safe and reliable recommendation services is the primary content of our discussion and research. We proposed SDRM-LDP. As mentioned earlier, to reduce transmission delay, the model puts some work of data processing at the user end. By filtering, the model reduces unnecessary data and provides users with more accurate recommendation services with the least user data. The model publishes the processed nonprivate data and the privacy item query results of privacy protection processing. This model uses the user's recent history to predict the user's future behavior and provides users with recommendations based on the predicted results. The retained data is protected by a random substitution combined with LDP to protect its private information and associated data to provide users with more safe and reliable fast recommendation services.

In this work, we focus on the following issues: (1) the recommendation in sparse data, (2) the prediction of short-term dynamic preferences, and (3) the privacy protection in data publication and query. The following is the summary of our contributions and improvements:

- (i) This paper uses hidden Markov chains (HMM) to predict users' short-term preferences and uses graphs to explore the relationship between categories and items. Finally, recommendation lists are built based on the above information
- (ii) The proposed model in this paper protects user privacy from two aspects: data publishing and data query. After deleting and randomly replacing non-private information, the recommendation system is constructed based on the published information. The above operations add local differential privacy to each user's privacy items and provide privacy protection for query results
- (iii) The proposed model balances the data availability and the privacy security. The balance is illustrated in quantitative indicators

This paper is organized into six sections as follows. Section 2 summarizes the current research on relevant work, which is mainly divided into two subsections as the introduction of the state-of-the-art status of recommendation system and the introduction of the research status of recommendation system based on privacy protection. In Section 3, we introduce the preliminary algorithms, including HMM, graph, and local differential privacy. Section 4 gives the demonstration of the SDRM-LDP model. Section 5 introduces the overall framework and the specific algorithms for recommendation and privacy protection. Section 6 is about the experiments including the information of datasets, experiment settings, and results. Section 7 is the conclusion of this paper and the future work plans.

## 2. Related Works

*2.1. Recommendation Systems Based on HMM.* The traditional recommendation systems construct the similarity matrix according to historical records and then sort and recommend based on the calculation. Later, more scholars [6] consider preference and context information when designing a model. Most of the researches use static data, but the preferences change over time in a more realistic setting. Therefore, some scholars proposed dynamic recommendation. At present, most of dynamic recommendations are based on HMM. There are mainly the following reasons why HMM gains such popularity. (1) HMM is dynamic and based on time series. (2) Through the study of its hidden layer and observation layer, scholars can discuss the development of user preference. Aghdam and Mobasher [7] explore the utility of user preference and introduced hierarchical HMM to capture the changes in user preferences. According to the user's feedback sequence to the model, the hierarchical HMM uses the user's current context as the hidden variable.

For known users, the model is used to infer the maximum likelihood sequence of the transformation between contexts and to predict the probability distribution of the next behavior based on this sequence. Tamayo et al. [8] study the changes of user preferences and propose a model which is a collaborative filtering recommendation system with temporary dynamics based on HMM. This model can trace the changes in user preference. Sahoo et al. [9] explain the user's item selection behavior based on HMM and provide personalized recommendations. A negative binomial-multinomial mixing model is proposed to simulate the user's choice of different items in each time period. Epure et al. [10] propose a personalized recommendation method to provide recommendations for specific users. It provides a basic degree of personalization while complying with the key characteristics of news recommendation including news popularity, recency, and the dynamics of reading behavior. All the above papers on dynamic recommendation are based on HMM. HMM illustrates good applicability in dynamic behavior prediction. This paper proposes SDRM based on HMM. This model is the same as the previous models. It is a dynamic model based on HMM. However, this model is not a dynamic prediction at the level of items, but a short-term dynamic prediction of users' internal preferences. This model pays more attention to user preferences rather than the correlation between items and users on the surface. However, the final purpose of this paper is to provide users with a list of recommendations. Therefore, on the basis of predicting the user's future behavior, we also explore the correlation between preferences and items, so as to build a personalized recommendation list in line with the user's personal preferences.

*2.2. Recommendation Systems Based on Privacy Protection.* The goal of a recommendation system is to recommend items or social content that users might be interested in. In recent years, many recommendation systems provide personalized recommendation services for users by collecting user's explicit or implicit characteristics, such as occupation, gender, age, address, and other information. Undoubtedly, the more comprehensive information, the better customized recommendations. However, improper collection, storage, and transmission of data lead to leakage of user sensitive information. Therefore, many scholars seek a more secure and reliable recommendation system, which prevents users' personal privacy from leaking while satisfying users' recommendation needs [11]. Chi et al. [12] use the location information as the research target. They proposed SRAmplified-LSH, which is amplifies LSH recommendation service based on location-sensitive information. This method can ensure the balance between accuracy, efficiency, and privacy. Garcia Clemente and He et al. [13, 14] study the privacy protection on mobile commerce recommendation. They used K-anonymity to protect user's id without the need for a trusted third party. The service is by the intelligent user selection algorithm based on grid map and Naive Bayes. They proposed a new framework to support private queries and evaluations. Al-Nazzawi et al.'s study [15] is a summary of the current research status based on LBRS. This paper explores the standards of privacy protection. Also, the paper demon-

strates the existing privacy protection methods and concerns on privacy measurement and attack. He et al. [16] proposed a protection scheme for the potential privacy of users in social networks so that the potential privacy of users can be protected without affecting the utility of data.

### 3. Preliminaries

We propose a recommendation system based on HMM and graph, which is SDRM-LDP. We use HMM to predict the changes in user preferences and use the graph to build the relationship between categories and items. Based on this information, we constructed a short-term dynamic recommendation system. We also consider the personal privacy disclosure and the attackers' use of background knowledge to infer target users and other situations that may cause harm to users' privacy. We propose a privacy protection mechanism in the recommendation system, which could protect personal privacy while providing users services. In the recommendation part, this paper addresses the cold start problem and predicts the future behavior of users who have historical records. This part mainly involves HMM's Baum-Welch algorithm [17]. There is also a brief introduction to graph theory and LDP. The details are demonstrated in the following part.

*3.1. HMM & Baum-Welch.* HMM is used to describe a Markov process with an implied position parameter. It is derived from the Markov model and is a double stochastic process. It contains a finite state Markov chain that describes the probability distribution of a transition from one state to the next. Another stochastic process represents a probabilistic correspondence between observable values and hidden states. The formula is as follows.

$$\lambda = (\pi, A, B). \quad (1)$$

$\pi$  is the initial state vector.  $A$  is the transfer matrix, and  $B$  is the emission matrix.  $\pi$  and  $A$  determine the sequence of states.  $B$  determines the sequence of observations. The process is shown in Figure 1.

The Baum-Welch (BW) algorithm is proposed by Welch in 1972 [18]. It is a special case of Expectation-Maximization (EM) algorithms. The algorithm is used to find the optimal HMM parameter  $\hat{\lambda}$ , which makes  $P(O|\lambda)$  maximum, in the case of a given observation sequence. In this paper, the BW algorithm is used to find the general optimal parameter  $\hat{\lambda}$  by combining user features and data features based on  $\hat{\lambda}$ . The model adjusts  $\hat{\lambda}$  for each user and optimizes the user's personalized parameters become  $\lambda_i$ . The model uses the personalization parameter  $\lambda_i$  to predict future preferences. The particular algorithm of BW is not specified in this paper.

*3.2. Graph.* Graph theory is the applied mathematics of graphs. It is the most commonly used modeling language and analysis tool for research and implementation. In graph theory, the direct mapping of nodes and edges shows the relationships between nodes in a network. We use a graph to simplify the complexity of the network so the relevant

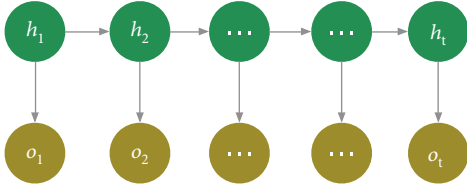


FIGURE 1: HMM structure.

information can be obtained more quickly and easily. We know that  $G$  consists of node  $V$  and edge  $E$ ; namely,  $G = (V, E)$ ,  $v_n \in V$ ,  $e_m \in E$ . The nodes and edges of a graph usually map directly to the relationships between nodes in a real network. The degree is the number of neighbors of a node as a basic index to evaluate the influence of a node. In this paper, the bipartite graph theory is used. Bipartite graphs divide nodes into two categories and represent the two related classes of nodes in a diagram. This makes it easy to analyze the interaction between nodes of different classes. In this paper, the user node and the item node are used as two types of graphs and the model uses the user as a medium to find relationships between items.

**3.3. Local Differential Privacy.** Privacy protection is a high concern. At present, scholars have put forward many methods to solve this problem, including anonymity, encryption, imnoise, and differential privacy. In this paper, local differential privacy is used to protect the recommendation system. This subsection gives a brief introduction to the basic concepts of local differential privacy. Traditional differential privacy, also known as central differential privacy, processes raw data centrally and publishes it to the public. The premise is that the data collector is a trusted third party. However, this is very difficult to do in reality because, since the birth of local differential privacy, it transfers the work to protect privacy to each user. This approach greatly reduces the likelihood of privacy leaks. The definition is set as follows.

**Definition 1.** (local differential privacy).

For any local differential privacy function  $f$ , its domain is  $\text{Dom}(f)$ . Its range is  $\text{Ran}(f)$ . For any input  $k, k' \in \text{Dom}(f)$ , its output is  $k^* \in \text{Ran}(f)$ . It satisfies the following formula.

$$P[f(k) = k^*] \leq e^\epsilon \times P[f(k') = k^*]. \quad (2)$$

## 4. Problem Statement

Before introducing the framework and algorithm of the paper, we give a clearer definition of concepts and formulas. This section is mainly composed of two parts. One is to give a clearer definition of some concepts that may cause confusion so the users can better understand this model. The other part is to give a specific definition of the formula involved in the algorithm.

**4.1. Concept Definition.** For a more explicit explanation of the research content, we give a clear definition of the concepts involved in the paper. For example, what is the preference

in the text? What part of our privacy should we protect? How do users query the statistical results of private information? We will all give clear answers by definition.

**Definition 2.** (like and preference).

Like: the user's choice of an item.

Preference: the user's choice of a category.

Like is a moment. For example, the user buys a red ball today, which may be a temporary choice. But if the user bought other balls in the same style, in a different color, then this user might have a preference for balls. It can be seen that liking exists in preference, and preference has a wider range that is why we chose to research preference. In this paper, we consider categories as preferences and study users' future behaviors based on that preference.

**Definition 3.** (privacy query).

There are two output results of this paper. One is a list of recommendations and the other is a statistical query result based on user privacy items. Privacy queries are defined as follows.

Privacy query: given the user information table  $D$ , the query function  $f(D^s)$  is the statistics of the frequency or mean or probability of some privacy  $S$ . The specific formula is as follows.

$$f(D^s): D^s \longrightarrow R. \quad (3)$$

Privacy query has some security risks. Attackers can target users based on background knowledge and privacy query results. Therefore, privacy protection is needed for this part.

### 4.2. Basic Definition

**Definition 4.** ( $G$ ).

$G = U \times L$ , where  $L$  is the link between users and items.  $U$  is defined as follows.

$$\begin{cases} u_i \in U, i = |U|, \\ C \times \text{Pr} = u_i, c_n \in C, n = |C|, \\ g_m \times R = c_n, \{1, 2, 3, 4, 5\} = R, \\ g_m \in G, m = |G|, \\ \{\text{id, sex, age, code}\} = \text{Pr}, \end{cases} \quad (4)$$

where  $u_i$  consists of historical records  $C$  and personal privacy items  $\text{Pr}$ .  $c_n$  is composed of its corresponding categories and ratings. The number of history records of users is  $(i, n)$ . The privacy item consists of the user's name, gender, age, code, and other privacy information.

**Definition 5.** ( $\Lambda$  &  $\hat{\lambda}$ ).

$\lambda_i \in \Lambda$ ,  $\lambda_i$  represented the model of  $u_i$ .  $\Lambda$  is a model set for all users.  $\hat{\lambda}$  is the optimal model based on all user data. This model is used to solve the cold start problem. The item category is taken as the hidden state.  $g_m \in G$ ,  $m \in |G|$ .  $G$  is genre

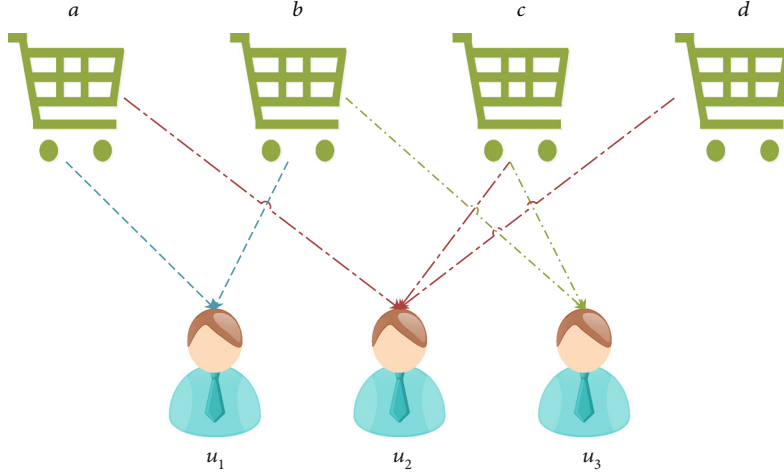


FIGURE 2: User-item relationship path diagram.

set.  $m$  is the number of genres. Each genre  $g_m$  contains  $n$  specific items. That is,  $c_n \in g_m, n \in |g_m|, c_{n \times m} \in C, |C| = n \times m$ ,  $C$  is item set. Rating as observed status  $R = (1, 2, 3, 4, 5)$ .

HMM consists of two sequences. One is a sequence of states also known as hidden sequence  $H, h_t \in H$ . The other one is the observation sequence  $O, o_t \in O$ . The genre sequence of items in the user's history is  $H$ . The sequence of corresponding ratings is  $O$ . A time threshold is set during preprocessing. The recent  $T$  items can be retained, while the others are deleted. Therefore,  $T$  can be regarded as the time threshold for filtering the history and can also be viewed as a sequence number for HMM.  $t \in T$ .

*Definition 6.* ( $\lambda_i$ ).

We present a general HMM model in Definition 5, which is specific for cold star users. This model is not suitable for the personalized recommendation of users with certain historical records. Therefore, we adjust  $\hat{\lambda}$ , according to the user's personal historical records. We add weights to the transfer matrix of the generic model to make it fit the user's personal preferences. The formula is as follows.

$$\hat{a}_{i,j} = a_{i,j} \times \sigma, \quad (5)$$

where  $a_{i,j}$  represents the transfer probability of  $u$ 's item  $i$  to  $j$ .  $i$  and  $j$  are items in  $u$ 's historical records.  $\sigma$  represents the weight added to an item in  $u$ 's history.

*Definition 7.* (relationship matrix  $E$ ).

Given an undirected bigraph  $G$ , the vertices of  $G$  are composed of users and items. We build the relationship matrix  $E$  between user and item based on  $G$ . Connect users to nonhistorical records using multilevel hops.  $E = U \times C, u_i \in U, c_n \in C$ .  $e$  is the number of paths between  $u_i$  and  $c_n$ . Examples are shown in Figure 2.

$u_1$  watches the movies  $a$  and  $b$ .  $u_2$  watches the movies  $a, c$ , and  $d$ .  $u_3$  watches the movies  $b$  and  $c$ . Then, we have  $(a \rightarrow b) = 2, (a \rightarrow c) = 2, (a \rightarrow d) = 2, (b \rightarrow c) = 2, (b \rightarrow d)$

$= 2, (c \rightarrow d) = 1$ . If we recommend a list for  $u_1$  because  $u_1$ 's historical records have  $a$  and  $b$ ,  $(\{a, b\} \rightarrow c) = 4, (\{a, b\} \rightarrow d) = 4$ . Hence, we recommend  $c$  and  $d$  for  $u_1$ .

*Definition 8.* (privacy).

The LDP in this paper is designed to address the situation where an attacker uses nonprivate items to reversely deduce user privacy information. For example, in a data list, 40% of users who have bought  $A$  are female, 60% of users who have bought  $B$  are female, and 20% of users who have bought  $C$  are female. Now,  $u$  has bought  $A$  and  $C$ , and then, there is a high probability that  $U$  is male. On the basis of formula (2) and in combination with the requirements of this paper, formula (7) is given. Meanwhile, the privacy budget is formula (12). If a user randomly changes an item in his or her history and randomly responses to his or her privacy items, the change in accuracy is no bigger than  $\epsilon$ , and the privacy is protected.

Formula (6) is a random exchange formula for user history.  $u_i$  has  $t$  records. If one is randomly picked due to the range of time threshold setting is small, there will be few records. If there are too many random substitutions, the data utility will become too low. Our purpose is to confuse the attacker, so replacing one of them is enough. The substituted item is other nonhistorical record in the same genre. The probability of that is  $1/(n_u - q)$ .  $n_u$  is the number of history records for the user.  $n_g$  is the number of items in this category.  $q$  is an item which has appeared in the historical records. Local differential privacy processes for privacy item query in this replace item. Through derivation, the privacy protection process satisfies the random disturbance mechanism and conforms to the definition of local differential privacy.

$$P(c_i \rightarrow c_j) = \frac{1}{n_u} \times \frac{1}{n_g - q}, \quad (6)$$

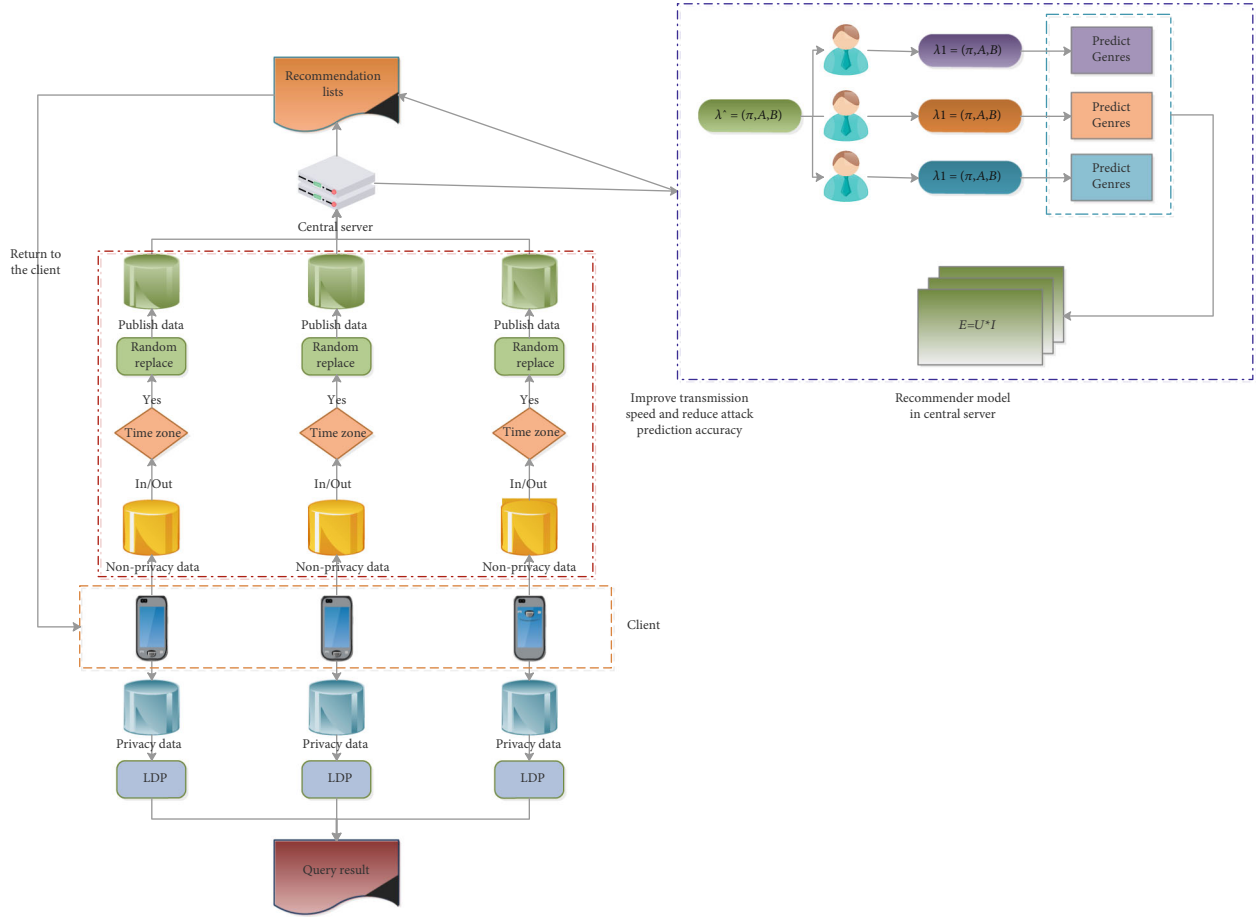


FIGURE 3: Overall framework of SDRM-LDP.

**Input:**  $U = C \times Pr; \alpha; \gamma$

**Output:** The Query Results & New History Records

1: **Initialization**

2: **for**  $i$  in  $|U|$  **do**

3: Sort  $c_{i,n}$  based on time

4: **if**  $n \leq |T|$  **then**

5: Deleting  $c_{i,n-|T|}$ .

6: Get new records  $c_{i,T}$ .

7: **Perturbing History Records**

8:  $P(c_i \rightarrow c_j) = (1/t) \times (1/(n-q))$

9: Get  $c_{i,T}$ .

10: **end if**

11: **end for**

12: **Local Differential Privacy**

13: Probability of true query result is  $\gamma$ , add noise  $\alpha$  for each user.

14:  $(P[f(k) = k^*]) / (P[P_{c_i \rightarrow c_j} \times f(k') = k^*]) \leq e^\epsilon$

15:  $\epsilon = \ln(\alpha / (1 - \alpha))$

16: Get the privacy item query result  $k^*$ .

ALGORITHM 1: Privacy protection.

**Input:**  $\pi$ ;  $O = \{o_1, o_2, \dots, o_t\}$ ;  $H = \{h_1, h_2, \dots, h_t\}$ ;  $G = (V, L)$ ,  $V = U \times C$   
**Output:** Recommendation Lists Rel.  
1: **Get**  $\hat{\lambda}$  &  $\lambda_i$   
2: **Initialization**  
3: Get  $\hat{\lambda}$  based on all dataset  
4: **for**  $i$  in  $|U|$  **do**  
5:  $A_i = A_{c_{i,n}} \times \beta$ ,  $1 \leq \beta$   
6:  $B_i = B_{c_{i,n}} \times \beta$   
7: Get  $\lambda_i$   
8: Predicting the next state item genre  $g_{i,m}$   
9: **end for**  
10: **Get Recommendation list Rel**  
11: Initialization  $E = U \times C$   
12: **for**  $i$  in  $|U|$  **do**  
13: **for**  $t$  in  $|c_i|$  **do**  
14: Find all the users that satisfy condition  $c_{i,t} \neq 0$ . Extra and get a new matrix.  
15: Sum each column separately. Get submatrix  $SE_{i,t}$   
16: **end for**  
17: Sum SE's each column separately.  
18: Get the relationship matrix  $E_i$  between the history records of user  $i$  and other items.  
19: Find all items in  $g_{i,m}$ . Sorting items. Get  $u_i$ 's recommendation list  $Re\ l_i$   
20: **end for**  
21: Get all users' recommendation list Rel.

ALGORITHM 2: Short-term dynamic recommendation model.

TABLE 1: General statistics about the two datasets.

Network property	MovieLens-100K	MovieLens-10M
Size of dataset	5 M	10 M
Number of users	943	2113
Number of movies	1682	10197
Number of genres	19	20
Range of rating	1-5	1-5
Number of privacy items	4	0

$$\frac{P[f(k) = k^*]}{P[P_{c_i \rightarrow c_j} \times f(k') = k^*]} \leq e^\epsilon. \quad (7)$$

The proof and derivation of local differential privacy after adding random replacement are as follows.

$$\begin{cases} b[\gamma\alpha + (1-\gamma)(1-\alpha)] = \frac{n_1}{n}, \\ b[(1-\gamma)\alpha + \gamma(1-\alpha)] = \frac{n_2}{n}, \\ n_1 + n_2 = n, \\ b = \frac{1}{n_u n_g - 1}. \end{cases} \quad (8)$$

$\gamma$  is the true proportion.  $\alpha$  is the added perturbation, which is the probability of random perturbation.  $n_1$  is the number of males.  $n_2$  is the number of females.  $n$  is the total.

Construct the likelihood function.

$$L(\gamma) = [b\gamma\alpha + b(1-\gamma)(1-\alpha)]^{n_1} [b\gamma(1-\alpha) + b(1-\gamma)\alpha]^{n_2}. \quad (9)$$

Get logarithmic likelihood function and derivate.

$$\begin{aligned} \ln L(\gamma) &= n_1 \ln [b\gamma(2\alpha - 1) + b(1-\alpha)] + (n - n_1) \ln [b\alpha - b\gamma(2\alpha - 1)], \\ \frac{d}{d\gamma} \ln L(\gamma) &= \frac{(2\alpha - 1)[-n_1 + n\alpha(2\gamma - 1) - n\gamma + n]}{[\alpha(2\gamma - 1) - \gamma][\alpha(2\gamma - 1) + (1-\gamma)]} = 0, \\ \hat{\gamma} &= \frac{\alpha - 1}{2\alpha - 1} + \frac{n_1}{(2\alpha - 1)n}, \\ E(\hat{\gamma}) &= \frac{1}{2(\alpha - 1)} \left[ \alpha - 1 + \frac{1}{n} \sum_{i=1}^n X_i \right] = \frac{1}{2(\alpha - 1)} [\alpha - 1 + \gamma\alpha + (1-\alpha)(1-\gamma)] = \gamma. \end{aligned} \quad (10)$$

Suppose that  $N$  represents the estimated number of men selected by a certain item statistically:

$$N = \hat{\gamma} \times n = \frac{\alpha - 1}{2\alpha - 1} n + \frac{n_1}{2\alpha - 1}. \quad (11)$$

In order to satisfy  $\epsilon$  - local differential privacy, the privacy budget  $\epsilon$  is set as

$$\epsilon = \ln \frac{\alpha}{1 - \alpha}. \quad (12)$$



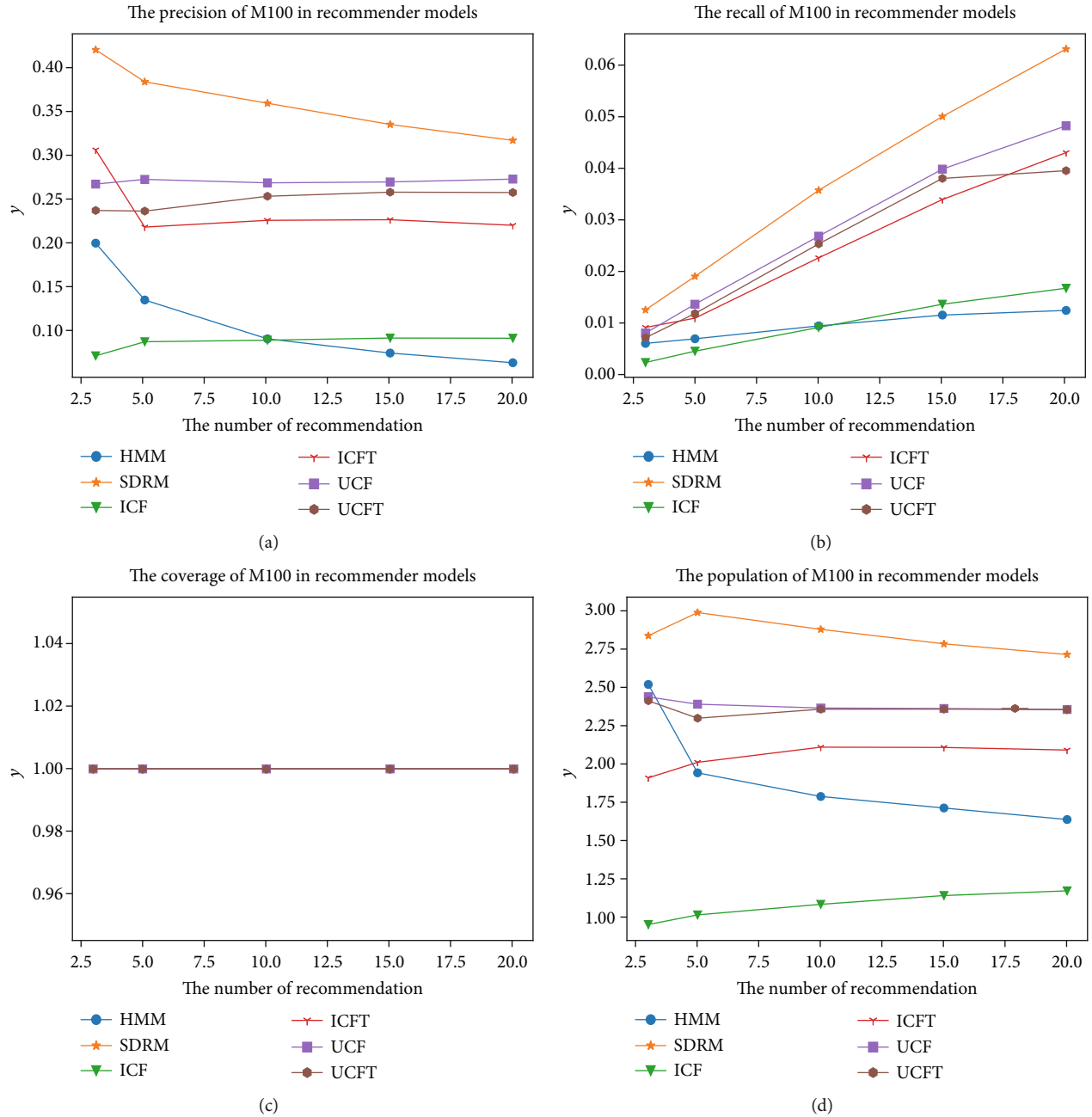


FIGURE 4: Continued.

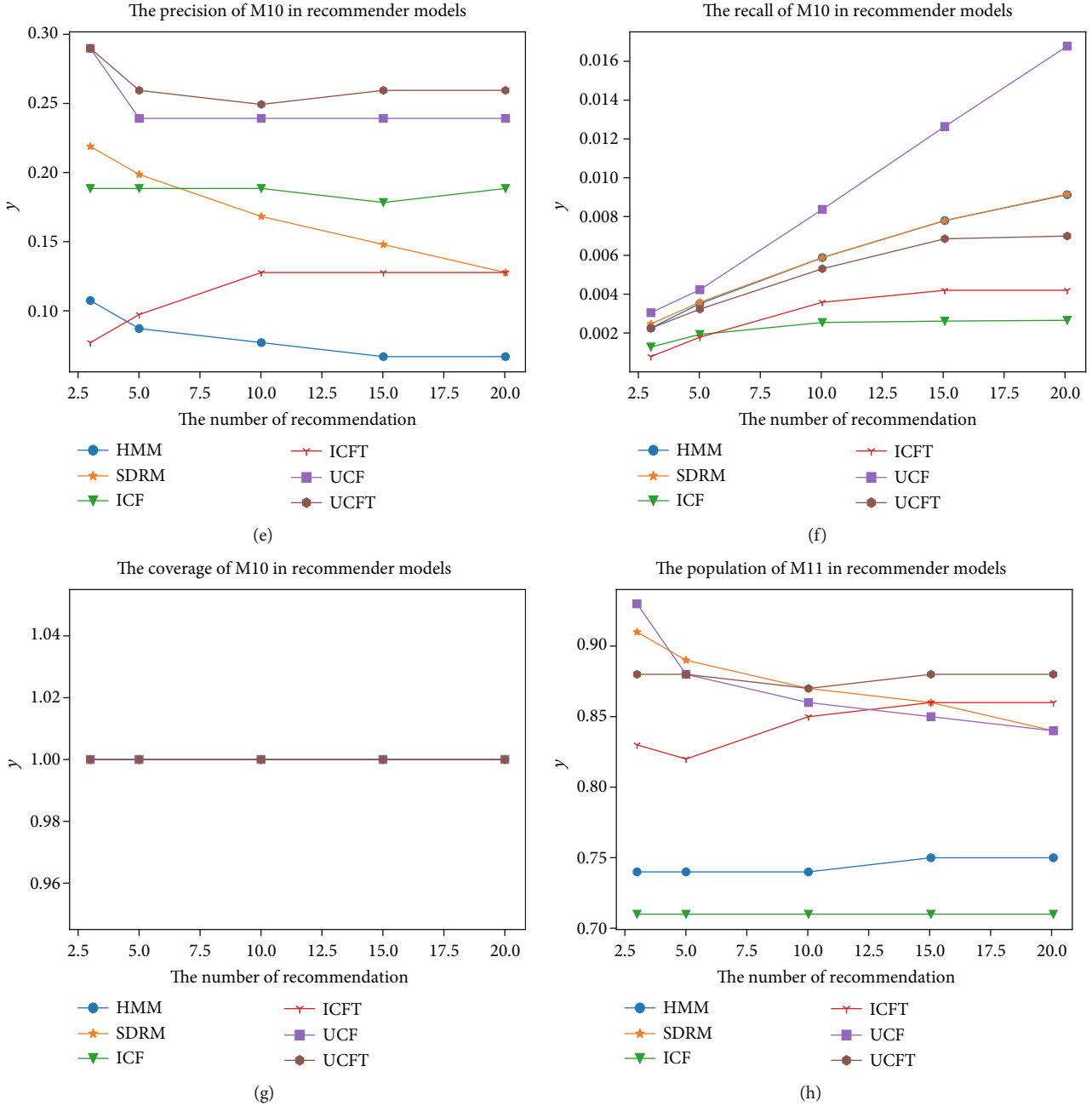


FIGURE 4: The recommendation result based on ML-100K dataset and ML-10M dataset. (a–d) Results of ML-100K dataset. (e–h) Results of ML-10M dataset.

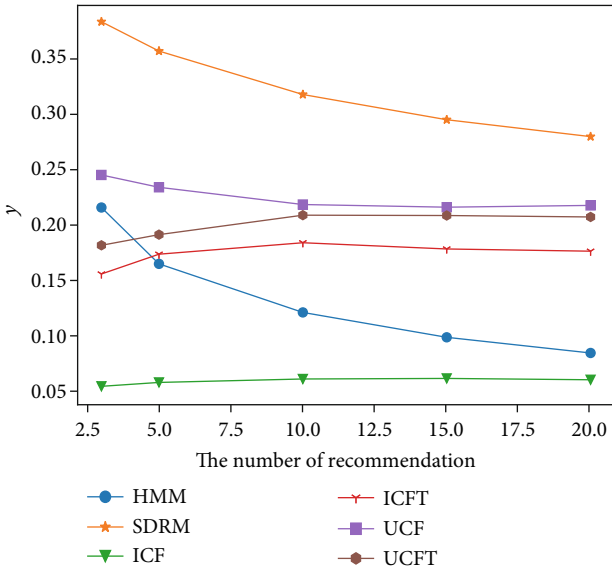
### 5. SDRM-LDP Model

This part is the core of the thesis. We introduce the framework of the model and the specific algorithm. SDRM-LDP model is mainly composed of two parts. One part is the SDRM recommendation system, and the other is the SDRM-LDP privacy protection mechanism. The two parts are presented separately. The following figure shows the overall frame of this paper (see Figure 3).

In Figure 3, we take into account the problems of user data transmission speed and user data leakage under 5G, so we delete the data based on the time threshold before upload-

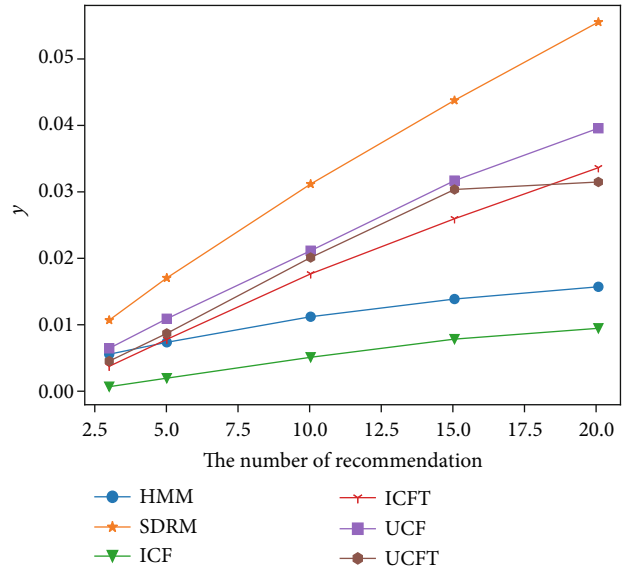
ing the data. At the same time, in order to strengthen the security of user data and prevent attackers from deducing private information by using nonprivate data, we randomly replaced the nonprivate data before localized differential privacy. The processed data is uploaded to the central processor, and the prediction of the user's future behavior and the construction of matrix  $E$  are completed in the central processor. Finally, a list of recommendations for each user is given. The other side is the bottom half of the terminal. With the addition of LDP to user privacy items, we offer statistical queries to third parties. The protection of this part is combined with the deletion and disturbance of

The population of M100 in recommender models based on random replace



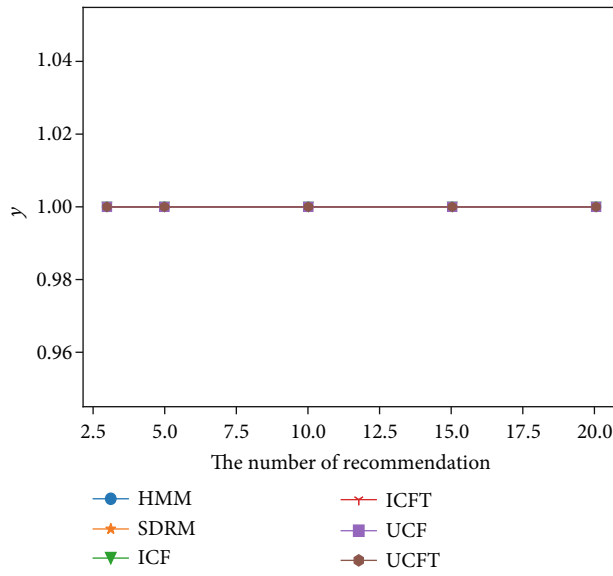
(a)

The recall of M100 in recommender models based on random replace



(b)

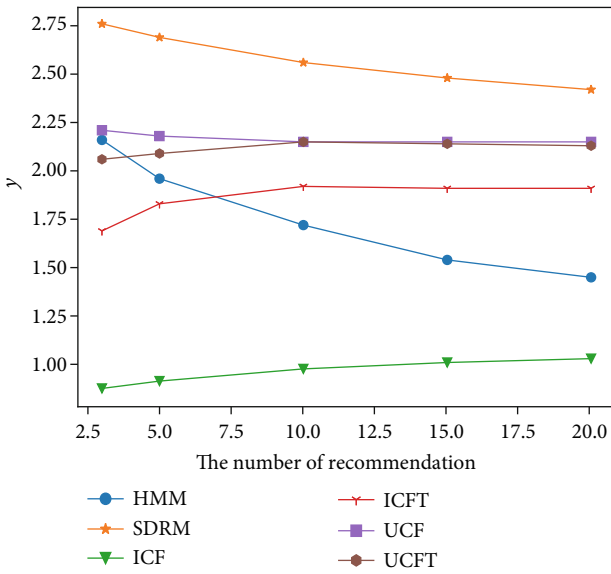
The coverage of M100 in recommender models based on random replace



(c)

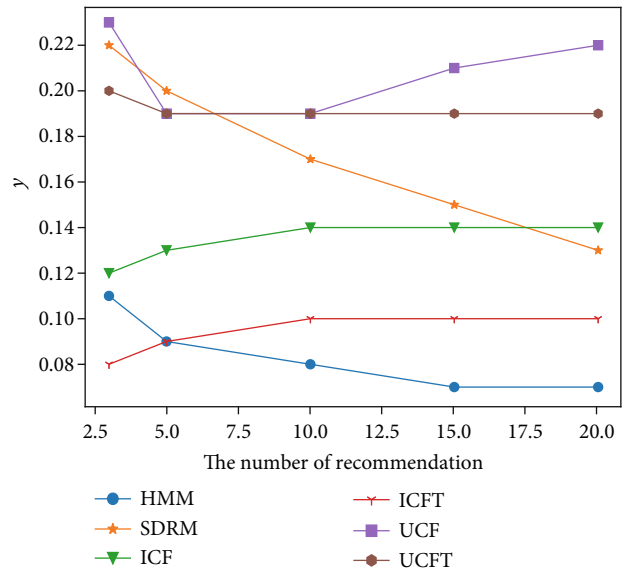
FIGURE 5: Continued.

The population of M100 in recommender models based on random replace



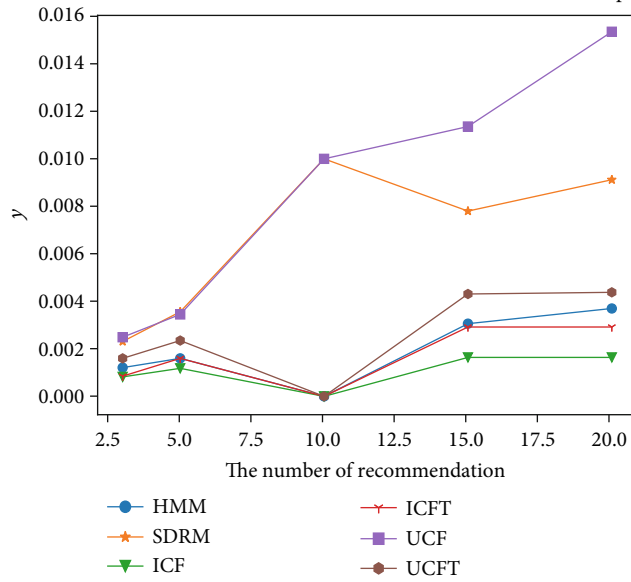
(d)

The precision of M10 in recommender models based on random replace



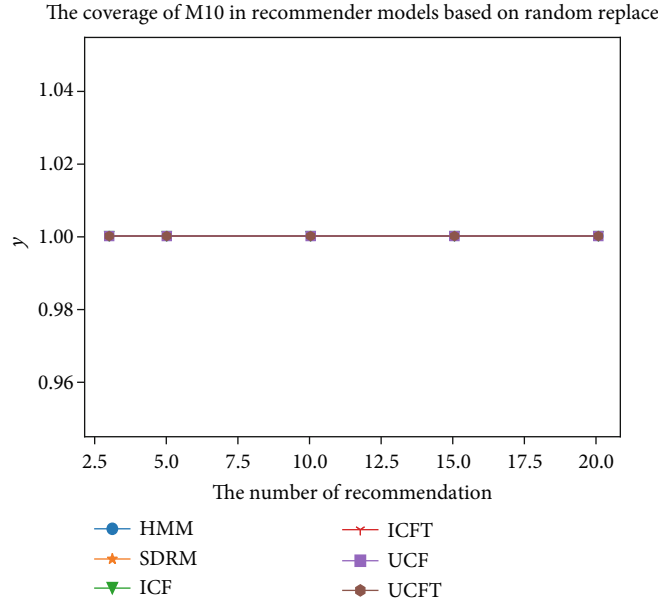
(e)

The Recall of M10 in recommender models based on random replace

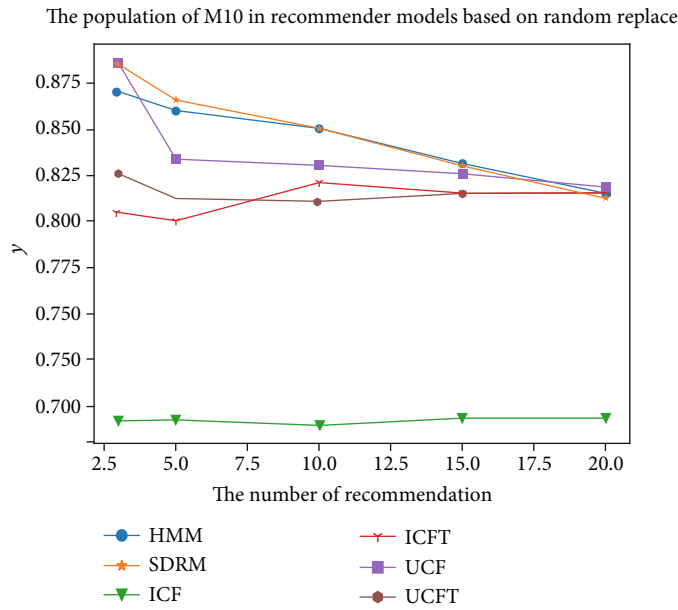


(f)

FIGURE 5: Continued.



(g)



(h)

FIGURE 5: The replaced recommendation result based on ML-100K dataset and ML-10M dataset. (a–d) Results of ML-100K dataset. (e–h) Results of ML-10M dataset.

the upper part. See Definition 8 for a detailed explanation of why this is considered.

**5.1. Privacy Protection Mechanism.** The data are processed from three perspectives: data source, data publication, and privacy query. The process includes data cleaning, filtering, replacement, and local differential privacy to protect the security of user data from all aspects. First, we process the raw data according to the time threshold and delete the data beyond the time threshold. There are two reasons. (1) Reduce user information from the raw data and reduce the risk of data leakage. (2) This model is the short-term behavior pre-

diction of users. Too much history can lead to overfitting the results. The total number of dataset after processing becomes  $t \times i$ , which is the number of users times the number of retained historical records. The second is to randomly replace items in the user's history with random nonhistorical items of the same genre. The purpose of this operation is to interfere with the attacker to accurately obtain nonsensitive information. The third step is to query the statistical results of privacy for the third party. Based on the above two steps, we add random disturbance to the user's privacy. It satisfies local differential privacy, which has been proved in the previous section. This step is intended to protect users'

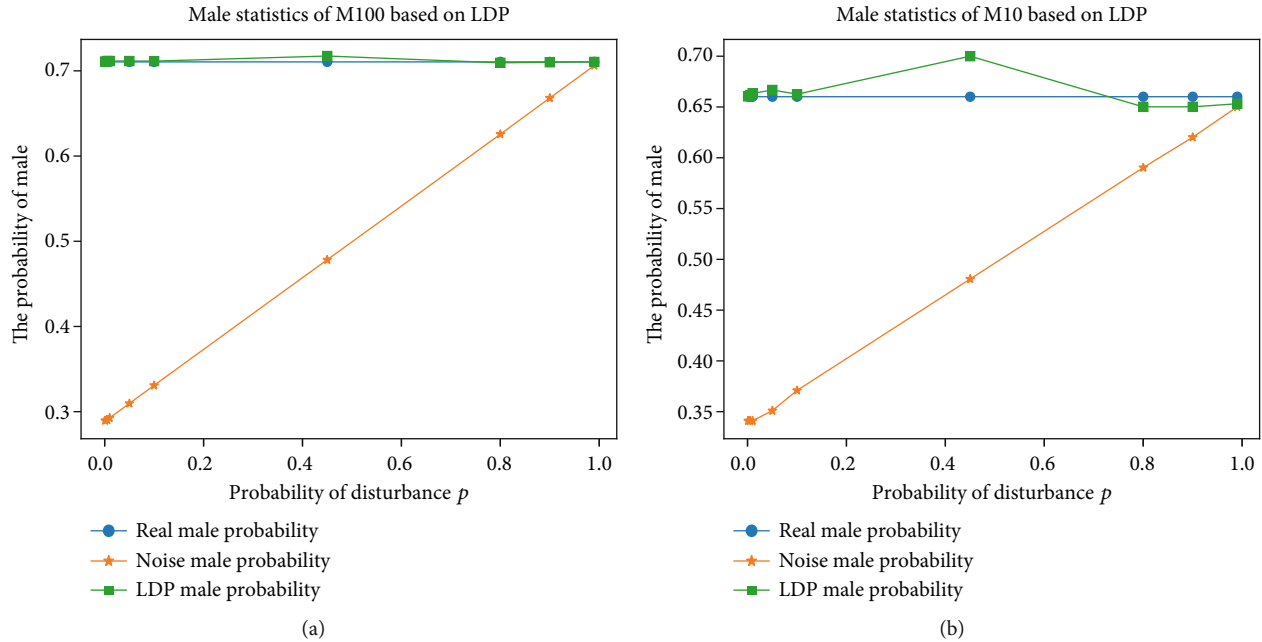


FIGURE 6: The results based on LDP replace on ML-100K and ML-10M.

information during private queries and prevent reverse inferences by attackers. The algorithm is as follows (see Algorithm 1).

5.2. *SDRM*. This paper explores users' future preferences based on their short-term behaviors. The relationship matrix between items and users is constructed based on user history to solve the relationship between items and genres in the future. First, according to Definition 5, we construct the general  $\hat{\lambda}$ .  $\hat{\lambda}$  could solve cold start for the new users. Based on  $\hat{\lambda}$ , we adjust each user based on the retained historical records and get  $\lambda_i$  for  $u_i$ .  $\lambda_i$  is used to predict the user's future preferences, which are the genres the user's most likely to choose in the next state and find all items in these genres in  $E$  and sort them to get the recommendation list for  $u_i$ . The recommendation algorithm is as follows (see Algorithm 2).

## 6. Evaluation

6.1. *Datasets*. The experimental datasets in this paper are all from the real world. One is the MovieLens-100K dataset and the other is the MovieLens-10M dataset. Because the experiment involves user privacy information, but MovieLens-10M lacks this part, we fill the part with random data. The information involved in the two datasets is collated in Table 1.

6.1.1. *MovieLens-100K*. The dataset is a virtual community site that recommends movies to users based on their preferences. It is a project of GroupLens Research LABS. The MovieLens dataset in this paper is the MovieLens-100K dataset from GroupLens [19]. The dataset is 5 MB in size and contains 100,000 comments, 943 users, 1682 movies, and at least 20 movies per user. The rating range is 1-5. Only integer rating is allowed. The dataset also includes some privacy of users, such as age, sex, occupation, and zip code. The data

are collected from September 19, 1997, to April 22, 1998. The dataset is divided into 5 cross-validation sets on a scale of 8:2.

6.1.2. *MovieLens-10M* [20]. In this dataset, users can comment on items they have purchased or used in the past. Other users can rate comments based on how useful the comments are. This part of the data is not required for this paper. The crawling period of this dataset is from May 2011. The dataset size is 10 M and has 2113 users and 10,197 films. The features include user ID, movie ID, genre, rating, comments, and time. The dataset lacks user privacy information. We use gender as the target information for privacy queries. Therefore, gender is randomly generated for users to complete the privacy inquiry part of the experiment. The experimental results in this part are for reference only. In fact, the randomly generated gender data has little effect on the local differential privacy experiment because this part of local differential privacy in the derivation is consistent with its original definition. If the ML-100K's results are consistent with our definition of privacy protection, we can assume that the ML-10M results are consistent in principle. Privacy information such as gender has no effect on other parts of the experiment except for the statistical query.

6.2. *Experiment Settings*. In this part, relevant settings are given for the experiment. The setting includes parameters, division ratio of training set and test set, evaluation metrics of recommendation system, and privacy protection.

6.2.1. *Training Dataset and Test Dataset*. The MovieLens-100K has 943 users and the MovieLens-10M has 2113 users. Each dataset is based on the real users. The users are randomly selected in a ratio of 8:2 to constitute the training set and the test set. The data is sorted by each user's history

time. Because each user has at least 20 records, the chronological order is separated by the number of time thresholds. For example, if the time threshold is set to 5, 3-5 historical records are kept to evaluate the final prediction results. So, we select the latest 5 as the training dataset of the model.

**6.2.2. Basic Parameter Settings.** Because each movie in the dataset corresponds to multiple categories, to reduce the computational complexity, we integrate multiple categories into one and rearrange them to form new categories. ML-100K is sorted into 218 new categories and ML-10M into 718 new categories.  $t_0$  is the latest date, which is the last date of data sorted by times as a reference data. The number of CF neighbors is 5. The kept states in SDRM are 5.  $\beta$  is 2.  $\sigma$  is 1.6. The number of recommended result is 3, 5, 10, 15, and 20.  $\alpha$  in the privacy query based on LDP is 0.001, 0.005, 0.01, 0.05, 0.1, 0.45, 0.8, 0.9, and 0.99.

**6.2.3. Evaluation Metrics.** The overall structure of this paper is divided into two parts: one is the recommendation system, and the other is the privacy protection algorithm. Therefore, corresponding evaluation indexes are given for different parts of the evaluation. The evaluation of a recommendation system is mainly divided into online evaluation and offline evaluation. Online evaluation is generally to evaluate the problems involved in the actual scenario, such as click rate and conversion rate. Obviously, this is not appropriate for this paper, so we choose to use offline evaluation. A total of 4 offline evaluation indexes are selected: accuracy rate, recall rate, coverage rate, and novelty. Accuracy is the percentage of the predicted results that are correct. The recall rate measures how many of all the correct outcomes are predicted. Accuracy and recall rates measure the accuracy of a recommendation algorithm. Coverage is a description of a recommendation system's ability to find the long tail of an item. The higher the coverage rate, the more long-tail items are recommended to users. Novelty refers to the proportion of items recommended to users that they have not heard of. A new method for measuring gluttenness is to use the average popularity of the recommended results, because the less popular the item, the more novel the user thinks it is. The specific formulas for accuracy, recall rate, and coverage are as follows:

$$\begin{aligned} \text{Precision} &= \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |R(u)|}, \\ \text{Recall} &= \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |T(u)|}, \\ \text{Coverage} &= \frac{U_{u \in U} R(u)}{I}. \end{aligned} \quad (13)$$

**6.3. SDRM Results.** We compared SDRM with general HMM-RS, item-based collaborative filtering, user-based collaborative filtering, item-based collaborative filtering with time factor, and user-based collaborative filtering with time context. Item-based collaborative filtering and user-based collaborative filtering can be regarded as the baseline. SDRM was compared to two models with added time context, because both of them and our algorithms take time into

account. In relevant work, we mentioned that some scholars also use HMM to make a recommendation system, but our HMM is an improvement, and we compare the two methods. We use four indicators of accuracy, recall rate, coverage rate, and population to evaluate the recommendation models. We use two real network datasets ML-100K and ML-10M for the experiment. The results are shown in Figure 4. Figures 4(a)–4(d) are the comparison results on the ML-100K dataset. The results show that SDRM is superior to other algorithms in the precision rate. The recall rates are similar except for the ICF and ICFT. The coverage of each algorithm is basically the same. SDRM is superior to other algorithms in terms of popularity, which is more important to the recommended. SDRM is superior to other algorithms in terms of overall performance.

Figures 4(e)–4(h) are the recommended results of ML-10M. The results show that UCF results are relatively high among these and SDRM is only inferior to UCF. The main reason for this is that ML-100K and ML-10M use the same parameter setting but ML-10M is much larger than ML-100K in both the number of movies and the number of movie genres. With no more parameters, the results of SDRM are slightly lower than UCF but higher than HMM and ICF.

The results shown in Figure 5 are random substitutions of some items in the original data within the same genre. Figures 5(a)–5(d) are the prediction results of randomly replacing one item in the training data of each user in ML-100K. In the figure below, SDRM is far superior to other recommended models in accuracy, recall rate, and popularity. Figures 5(e)–5(h) are the results of random replacement based on dataset ML-10M. In Figures 5(e) and 5(f), when the number of recommendations is relatively small, the accuracy and recall rate of SDRM are higher than other recommended models. SDRM's popularity is better than other models. The interpretation for this result is basically the same as the original data recommendation that the parameters are not adjusted according to the dataset to be optimal for SDRM. However, SDRM is more stable when the data part is changed, which also indicates that the model is indeed a prediction based on user preferences.

**6.4. SDRM-LDP Results.** Figure 6 is the comparison result of LDP privacy query based on random replacement under ML-100K and ML-10M datasets, respectively. From the results, the LDP-based query results are pretty much the same as the real result. This evidence shows that while users' privacy is protected, the utility availability of the data is still relatively high, with little loss. The start line is the result of uncorrected, only after adding noise. The LDP-based privacy protection works better according to the experiments.

## 7. Conclusions

In this paper, we discuss the following issues: (1) user preference prediction and recommendation in dynamic conditions and (2) how to provide recommendations while protecting user privacy. For the first issue, we propose the SDRM model. The model is not a traditional recommendation system based on similarity. This model uses HMM and graph to find the

dynamic preference model belonging to each user. Genre-based forecasting reduces the amount of computation required. At the same time, with changes to certain data, the prediction result is more stable. In general, the traditional recommendation model is too superficial, and every data has an impact on future predictions. The SDRM explores the selection of users' internal preferences. Therefore, a fraction of data changed has little impact on the predicted results. For the second issue, we query the probability of each user's privacy item based on random replacement items. The process is based on LDP, and the experiment results show that the noise adding does not change the performance in significant value, so the privacy of users is protected without affecting the utility of the data. After the random replacement, the data and the privacy query results are all the data we released to the public. SDRM-LDP guarantees the information security of users and reduces the risk of attacks. Both the final results of the recommendation model and the privacy protection model are satisfactory.

## Data Availability

The data used to support the findings is generally unavailable due to public releasability constraints. Please contact the corresponding author for special release consideration.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

Our research fund is funded by the Fundamental Research Funds for the Central Universities (3072020CFQ0602, 3072020CF0604, and 3072020CFP0601) and 2019 Industrial Internet Innovation and Development Engineering (KY10600200021 and KY10600200008).

## References

- [1] 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.
- [2] 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.
- [3] M. B. Holbrook and R. M. Schindler, "Some exploratory findings on the development of musical tastes," *Journal of Consumer Research*, vol. 16, no. 1, pp. 119–124, 1989.
- [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*, p. 1, 2016.
- [5] Z. Cai and Z. He, "Trading private range counting over big IoT data," *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, 2019, Dallas, TX, USA, 2019.
- [6] Y. Wang, G. Yin, Z. Cai, Y. Dong, and H. Dong, "A trust-based probabilistic recommendation model for social networks," *Journal of Network and Computer Applications*, vol. 55, pp. 59–67, 2015.
- [7] N. H. M. H. Aghdam and B. Mobasher, "Adapting recommendations to contextual changes using hierarchical hidden Markov models," *RecSys '15: Proceedings of the 9th ACM Conference on Recommender Systems*, pp. , 2015241–244, 2015.
- [8] S. Cleger Tamayo, L. A. Hernandez Leyva, L. W. Osorio Gamez, and A. L. Scull Pupo, "Temporal dynamic study in personalization digital newspaper ahora!," *IEEE Latin America Transactions*, vol. 13, no. 8, pp. 2792–2797, 2015.
- [9] N. Sahoo, P. V. Singh, and T. Mukhopadhyay, "A hidden Markov model for collaborative filtering," *MIS Quarterly*, vol. 36, no. 4, pp. 1329–1356, 2012.
- [10] V. E. Epure, B. Kille, E. J. Ingvaldsen, R. Deneckere, C. Salinesi, and S. Albayrak, "Recommending personalized news in short user sessions," *RecSys*, pp. , 2017121–129, 2017.
- [11] W. Huang, B. Liu, and H. Tang, "Privacy protection for recommendation system: a survey," *Journal of Physics: Conference Series*, vol. 1325, p. 012087, 2019.
- [12] X. Chi, C. Yan, H. Wang, W. Rafique, and L. Qi, "Amplified locality-sensitive hashing-based recommender systems with privacy protection," *Concurrency and Computation Practice and Experience*, no. 1, 2020.
- [13] F. J. Garcia Clemente, "A privacy-preserving recommender system for mobile commerce," *2015 IEEE Conference on Communications and Network Security (CNS)*, 2015, pp. 725–726, Florence, Italy, 2015.
- [14] Z. He, Z. Cai, J. Yu, X. Wang, Y. Sun, and Y. Li, "Cost-efficient strategies for restraining rumor spreading in mobile social networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 2789–2800, 2017.
- [15] T. S. Al-Nazzawi, R. M. Alotaibi, and N. Hamza, "Toward privacy protection for location based recommender systems: a survey of the state-of-the-art," in *2018 1st International Conference on Computer Applications & Information Security (ICCAIS)*, Riyadh, April 2018.
- [16] Z. He, Z. Cai, and J. Yu, "Latent-data privacy preserving with customized data utility for social network data," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 665–673, 2018.
- [17] L. R. Welch, "Hidden Markov models and the Baum-Welch algorithm," *IEEE Information Theory Society Newsletter*, vol. 53, no. 2, pp. 194–211, 2003.
- [18] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–22, 1977.
- [19] <https://grouplens.org/datasets/movielens/100k/>.
- [20] <https://grouplens.org/datasets/hetrec-2011/>.