

# Semantic Sensor Data Annotation and Integration on the Internet of Things 2021

Lead Guest Editor: Xingsi Xue

Guest Editors: Yuemin Ding, Pei-Wei Tsai, and Chin-Ling Chen





---

# **Semantic Sensor Data Annotation and Integration on the Internet of Things 2021**



Wireless Communications and Mobile Computing

---

**Semantic Sensor Data Annotation and  
Integration on the Internet of Things  
2021**

Lead Guest Editor: Xingsi Xue

Guest Editors: Yuemin Ding, Pei-Wei Tsai, and  
Chin-Ling Chen




---



Copyright © 2022 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

## **Label Propagation Clustering Algorithm Based on Adaptive Angle**

Hui Du , Manjie Zhang , Zhihe Wang , Qiaofeng Zhai , and Xuyan Cao 


Research Article (11 pages), Article ID 7535575, Volume 2022 (2022)

## **A Joint Model of Natural Language Understanding for Human-Computer Conversation in IoT**

Rui Sun , Lu Rao , and Xingfa Zhou 







Research Article (10 pages), Article ID 2074035, Volume 2022 (2022)

## **Multiobjective Optimization Model and Algorithm Based on Differential Brain Storm for Service Path Constructing**

Xiaoqiang Jia , Li Ge, Yunfei Li, Jun Liu, and Biying Zhou






Research Article (12 pages), Article ID 2136933, Volume 2022 (2022)

## **Study of Water Resources Optimal Operation Model of Multireservoir: A Case Study of Kuitun River Basin in Northwestern China**

Changlu Qiao , Yan Wang , Yanxue Liu , Junfeng Li , Heping Zhang , and Jiangang Lu 

Research Article (17 pages), Article ID 7715398, Volume 2022 (2022)

## **Energy-Efficient Data Transmission in Mobility-Aware Wireless Networks**

Mengmeng Xu , Guixiang Zhang , Xinpei Liang , Hai Zhu , and Juanjuan Wang 

Research Article (11 pages), Article ID 8683854, Volume 2022 (2022)

## **Research on Distributed Multisensor Spectrum Semantic Sensing and Recognition in Internet of Things Environment**

Xiguo Liu , Jing Zhang, Min Liu, Zhongyang Mao , and Changbo Hou


Research Article (15 pages), Article ID 2608885, Volume 2022 (2022)

## **A Load-Aware Multistripe Concurrent Update Scheme in Erasure-Coded Storage System**

Junqi Chen , Yong Wang , Miao Ye , Qinghao Zhang , and Wenlong Ke 



Research Article (15 pages), Article ID 5392474, Volume 2022 (2022)

## **Lightweight Security Wear Detection Method Based on YOLOv5**

Sitong Liu, Nannan Zhang, and Guo Yu 


Research Article (14 pages), Article ID 1319029, Volume 2022 (2022)

## **A Robust Pupil Localization via a Novel Parameter Optimization Strategy**

Wenjun Zhou , Xiaoyi Lu, and Yang Wang 



Research Article (12 pages), Article ID 2378911, Volume 2022 (2022)

## **Prediction of Air Leakage Rate of Sintering Furnace Based on BP Neural Network Optimized by PSO**

Xiaokai Quan, Nannan Zhang, Guo Yu , Qunfeng Liu, and Lianbo Ma





Research Article (9 pages), Article ID 5631787, Volume 2022 (2022)

## **Label Propagation Community Detection Algorithm Based on Density Peak Optimization**

Ma Yan  and Chen Guoqiang 

Research Article (12 pages), Article ID 6523363, Volume 2022 (2022)

### **IOT Automation with Segmentation Techniques for Detection of Plant Seedlings in Agriculture**

Shoab Kamal , K. R. Shobha, Flory Francis, Rashmita Khilar, Vikas Tripathi, M. Lakshminarayana , B. Kannadasan , and Kibebe Sahile 



Research Article (10 pages), Article ID 6466555, Volume 2022 (2022)

### **Neural Network-Based Ultra-High-Definition Video Live Streaming Optimization Algorithm**

Yunning Feng, Nan Hu , and Xiaosheng Yu

Research Article (10 pages), Article ID 2509209, Volume 2022 (2022)

### **Anomaly Detection and Restoration for AIS Raw Data**

Shuguang Chen , Yikun Huang , and Wei Lu

Research Article (11 pages), Article ID 5954483, Volume 2022 (2022)

### **UAV Path Planning Based on Multicritic-Delayed Deep Deterministic Policy Gradient**

Runjia Wu, Fangqing Gu , Hai-lin Liu, and Hongjian Shi



Research Article (12 pages), Article ID 9017079, Volume 2022 (2022)

### **Solving Sensor Ontology Metamatching Problem with Compact Flower Pollination Algorithm**

Wenwu Lian , Lingling Fu , Xishuan Niu , Junhong Feng , and Jian-Hong Wang 





Research Article (7 pages), Article ID 9662517, Volume 2022 (2022)

### **Improved Optimal Path Finding Algorithm of Multistorey Car Park Based on MCP Protocol**

Zhendong Liu , Dongyan Li , Xi Chen, Xinrong Lv, Yurong Yang, Ke Bai, Mengying Qin, Zhiqiang He, Xiaofeng Li, and Qionghai Dai






Research Article (9 pages), Article ID 3821414, Volume 2022 (2022)

### **A Secured and Efficient Anonymous Roaming Scheme of Mobile Internet**

Yao Cheng , Li Yue , Naixia Duan , and Chenglong Li 




Research Article (9 pages), Article ID 5426288, Volume 2022 (2022)

### **Online Missing Data Imputation Using Virtual Temporal Neighbor in Wireless Sensor Networks**

Yulong Deng , Chong Han , Jian Guo , Linguo Li , and Lijuan Sun 






Research Article (20 pages), Article ID 4909476, Volume 2022 (2022)

### **Basic Research on Ancient Bai Character Recognition Based on Mobile APP**

Zeqing Zhang , Cuihua Lee, Zuodong Gao , and Xiaofan Li 

Research Article (7 pages), Article ID 4059784, Volume 2021 (2021)

### **Improved Joint Optimization Design for Wireless Sensor and Actuator Networks with Time Delay**

Lihan Liu , Yuehui Guo , Yang Sun , Zhuwei Wang , Enchang Sun , and Yanhua Sun

Research Article (10 pages), Article ID 3927584, Volume 2021 (2021)

### **Enterprise Financial Risk Management Using Information Fusion Technology and Big Data Mining**


Huabo Yue, Haojie Liao , Dong Li, and Ling Chen

Research Article (13 pages), Article ID 3835652, Volume 2021 (2021)

# Contents

---

**Intrusion Detection Analysis of Internet of Things considering Practical Byzantine Fault Tolerance (PBFT) Algorithm**

Leixia Li, Yong Chen, and Baojun Lin 

Research Article (9 pages), Article ID 6856284, Volume 2021 (2021)

**Medical Record Encryption Storage System Based on Internet of Things**

Yamei Zhan and Zhaopeng Xuan 

Research Article (9 pages), Article ID 2109267, Volume 2021 (2021)

## Research Article

# Label Propagation Clustering Algorithm Based on Adaptive Angle

Hui Du , Manjie Zhang , Zhihe Wang , Qiaofeng Zhai , and Xuyan Cao 

The School of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

Correspondence should be addressed to Manjie Zhang; 2020222042@nwnu.edu.cn

Received 13 March 2022; Accepted 21 July 2022; Published 25 August 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Hui Du 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 direction-based label propagation clustering (DBC) algorithm needs to set the number of neighbors ( $k$ ) and the angle value (*degree*), which are highly sensitive. Moreover, DBC algorithm is not suitable for datasets with uneven neighbor density distribution. To overcome above problems, we propose an improved DBC algorithm based on adaptive angle and label redistribution (ALR-DBC). The ALR-DBC algorithm no longer input parameter *degree*, but dynamically adjusts the deviation angle through the concept of high-low density region to determine the receiving range. This flexible receiving range is no longer affected by the uneven distribution of neighbor density. Finally, those points that do not meet the expectations of the main direction are redistributed. Experiments show that the ALR-DBC algorithm performs better than DBC algorithm in most artificial datasets and real datasets. It is also superior to the classical algorithms listed. It also has good experimental results when applied to wireless sensor data annotation.

## 1. Introduction

The frontier development of computer science has focused on data mining and artificial intelligence in recent years. Cluster analysis is the most classical research in the unsupervised direction of machine learning. It is widely used in image analysis [1], knowledge discovery [2], medicine [3], pattern recognition [4], and other fields. In the case of unknown sample information, the points are divided by some similarity measurement method so that the similarity of points within clusters is high while that of points between clusters is low [5]. Many classical algorithms show good results on different datasets [6].

The basic idea of hierarchical clustering is bottom-up and merging layer by layer [7]. The clustering process starts with each point being a separate class, and then, the two classes with the highest similarity are merged and iteratively repeated. The similarity is mainly measured by distance. The closer the distance is, the higher the similarity is. The advantage of this approach is that it does not input the number of clusters and hierarchical relationship between classes can be discovered. The disadvantage is high time complexity and low efficiency [8].  $K$ -means is an early classic and widely used algorithm in the field of data mining [9]. Compared

with hierarchical clustering, it has a much lower time complexity. It has high scalability and compressibility when dealing with big datasets. It is highly dependent on the selection of the initial centroid, so it often takes several iterations to achieve better clustering results. And  $K$ -means algorithm cannot cluster nonspherical datasets [10]. Compared with  $K$ -means clustering method, affinity propagation clustering algorithm (AP) is more robust and accurate [11]. It starts with initializing two preset matrices and then iteratively updates them. The “responsibility”  $r(p, z)$  represents the suitability of point  $z$  as the clustering center of point  $p$ . The “availability”  $a(p, z)$  represents the suitability of point  $p$  to select point  $z$  as its cluster center in the current round. These two matrices are interrelated and determine the final clustering result. When both are large, it means that point  $z$  has strong competitiveness and is more likely to be selected as the clustering center. AP algorithm has good performance and efficiency. Different from the clustering centers in other algorithms, the exemplar (center of clustering) in AP algorithm is an exact data point in the original data. And it is started by inputting the similarity matrix, so the data are allowed to be asymmetric and the sum of squares of error is low. However, the complexity of AP algorithm is high and the running time is long when the data is large [12].



In order to solve the clustering problem of irregular shapes, density-based spatial clustering of applications with noise algorithm (DBSCAN) was proposed [13]. It uses density reachability and density connection to cluster by establishing the definition of a core point, boundary point, and noise point [14]. DBSCAN algorithm can achieve adaptive clustering. There is no need to give the parameter of expected cluster number, and the advantage of insensitivity to noise points is conducive to clustering. The disadvantage is that the parameter sensitivity is high, and small parameter changes may lead to large differences in the results. And the DBSCAN algorithm must specify a density threshold to remove noise points below this density threshold. Based on the above analysis, the clustering by fast search and find of density peaks (DPC) [15] is put forward on the premise that the two assumptions are true: the cluster center is surrounded by points with lower density than it, and the distance between these points and the cluster center is closest compared with other cluster centers. And it has a relatively far distance from the point where density is higher than itself. Only meeting these conditions at the same time can it be possible to become the clustering center. Its disadvantage is that it needs to calculate the distance between all points. The DPC algorithm also does not cluster well those sample sets with multidensity peaks [16].

In order to better cluster samples with uneven density distribution, scholars continue to explore. The reference [17] uses two parameters. The parameter  $k$  is used to determine the receiving direction, and the parameter *degree* is used to define the receiving range that can deviate from the receiving direction. When the density distribution is not uniform, the clustering effect of DBC algorithm will not be greatly affected. After a large number of experiments, it is known that if the first parameter  $k$  is not appropriate, the second parameter *degree* will be adjusted many times to achieve the desired clustering effect. After determining the  $k$ -nearest neighbor value on some datasets, the value of degree can only change in a small range; otherwise, the expected value cannot be achieved. Therefore, the DBC algorithm is improved. We reduce the parameter sensitivity by selecting the method of dynamically adjusting the deviation angle to determine the receiving range of each point and using the DBC algorithm to cluster the points. It can be seen from the evaluation index of the experimental part that the improved algorithm holds higher NMI and ARI on most of the tested datasets. In this paper, the new improved algorithm is named ALR-DBC algorithm.

Clustering also has outstanding performance in practical applications, such as applying it to the Internet of Things. The Internet of Things came into being with the vigorous development of the information technology industry. It connects the object with the network through the information sensing device according to the agreement. As the core of the Internet of Things, the perception layer can not only sense signals and identify objects but also has the function of processing and controlling. The wireless sensor network is an important part of perception layer. It realizes the data collection, processing, and transmission and sends the information to the network owner. To enhance the communica-

tion between sensor networks, it is essential to establish semantic connections between sensor ontologies in this field. Literature [18] proposed a new sensor ontology integration technology, which utilizes the debate mechanism to extract sensor ontology alignment, greatly improving the effectiveness of the whole wireless sensor network. It enhances the communication ability between wireless sensors and improves the performance of the whole network. At the same time, we also read the relevant literature on matching ontology [19, 20] to better understand the important role and help of these methods in this field. Inspired by this, this paper applies ALR-DBC algorithm to wireless sensor networks and conducts performance comparison experiments.

## 2. Related Work

In this chapter, we will explain the label propagation algorithm (LP) [21] and the direction-based label propagation algorithm (DBC). The basic idea of these two algorithms is clustering through the similarity between the sample points. The DBC algorithm is an improvement of LP algorithm by adding parameters of angle value. Through the description and comparison of these two algorithms, it is more helpful to understand the improved DBC algorithm.

*2.1. LP Algorithm.* It is assumed that each point can find  $k$  neighbors closest to it, which is the basic assumption of LP algorithm. These neighbors all have unique class tags. Then, the update of the cluster label is determined by the neighbor's label. Among the labels to which the neighbor belongs, the label that occupies the largest number is the new cluster label of the point. Repeat the process; when the label of all points do not change, the iteration ends [22]. In the LP algorithm, the number of iterations needs to be set to avoid being overcalculated affecting the final result. In the running process of the algorithm, there is no need to calculate any clustering index, nor to input the number of clustering. The disadvantage of LP algorithm is that the randomness of the sequence in the iterative process will lead to the same initial label setting, but the clustering results are very different. It may also be affected by the  $k$  nearest neighbors set, the maximum values of neighbor labels of some points are the same, so random selection is adopted to update the cluster labels. Based on this, scholars also put forward improvement strategies, such as introducing potential function [23] and LeaderRank value [24] to increase the weight of nodes or edges. In this way, the centroid effect can be preliminarily determined and the classification accuracy can be improved. The label entropy attribute [25] can also be added so that it can be sorted according to the label entropy of each point and avoid the random influence in the iteration process.

*2.2. DBC Algorithm.* DBC is a direction-based cluster label propagation clustering algorithm. It need not enter the number of classes that will eventually be generated and can cluster any shape of clusters with stable results. Distance and direction are the two basic physical metrics, which are helpful for clustering. The major difference between LP and DBC algorithm is that the DBC algorithm considers

the orientation relationship between sample points, while the LP algorithm considers the relationship between the numbers of labels to which neighboring points belong. The DBC introduces the second parameter *degree* and finds the direction and receiving range with the greatest density in each point's neighborhood. There is no feedback or update during tag propagation, so the selection of receiving range is very important. The general process of the algorithm is to find the receiver of each point according to the parameters set and select the point with the largest number of receivers as the starting point of this round. After it is assigned an initial label, the clustering starts and the points in the receiver list are classified into one category. These points continue to pass their labels to their respective receivers as new senders until a round of clustering ends without new senders. Next, the remaining unlabeled points continue the next round of clustering until all points have class labels, and the clustering is over. We assume that point  $P$  is an arbitrary sample point in the dataset. The mathematical concept shows that point multiplication reflects the "similarity" of two vectors. The more similar the two vectors are, the greater the point multiplication is. Therefore, the formula of receiving direction of point  $P$  is shown in

$$\vec{v} = \arg \max_{\vec{v}_i \in V} \sum_{\vec{v}_j \in V: \vec{v}_i \cdot \vec{v}_j \geq 0.5} \vec{v}_i \cdot \vec{v}_j. \quad (1)$$

The set  $V$  stores all the neighbor vectors of point  $P$ .  $\vec{v}$  is the most densely distributed direction of the neighbors of point  $P$ .

After determining the receiving direction, the DBC algorithm also sets the maximum deviation angle to determine the receiving range of cluster labels. It refers to the maximum angle that a neighbor vector can deviate from the receiving direction, and points within this angle range can pass the cluster label to point  $P$ . The DBC algorithm also defines the concepts of sender and receiver. After the parameters are determined, each point can obtain its sender and receiver according to its receiving direction and maximum deviation angle. Then, the label propagation of DBC algorithm begins. Algorithm 1 describes the DBC algorithm.

We default that each time the tag number is updated and a new pass is started. The SenderList keeps storing the sample that currently has no tags and has the most receivers. ID represents the number of samples. NewSenderList is a list of new senders. Line 21 of Algorithm 1 is to find the number of samples that are not currently labeled.

### 3. ALR-DBC Algorithm

**3.1. Basic Idea.** In describing the DBC algorithm, we mention that it introduces the concept of direction and angle as a reference condition based on the basic method of cluster label transfer. After determining the range of label transmission, the clustering process will begin. We consider that the number of neighbors and the angle value are global parameters. In fact, the neighbor density around each point is unevenly distributed. If the receiving range of each point

```

1: Num = Dataset.size
2: ID = 0
3: //Traverse SenderList to get its recipients. Assign
//labels to unlabelled recipients and store them
//in NewSenderList. Until all points have class
//labels
4: while Num > 0 do
5:   ID = ID + 1
6:   while SenderList.size > 0 do
7:     NewSenderList = []
8:     for i = 0; i < SenderList.size; i++ do
9:       Receivers = getReceivers(SenderList[i])
10:      for j = 0; j < Receivers.size; j++ do
11:        Receiver = receivers[j]
12:        if Receiver.label == 0 then
13:          Receiver.label = ID
14:          NewSenderList.append(receiver)
15:        end if
16:      end for
17:    end for
18:    SenderList.clear()
19:    SenderList = NewSenderList.copy()
20:  end while
21:  Num = NumberOfUnassignedPoints(dataset)
22: end while

```

ALGORITHM 1: Label transfer process of DBC.

can be dynamically changed according to its neighbor density, the accuracy of clustering results can be improved. We use Figure 1 to understand this hypothesis.

In Figure 1(a), the receiving range is all samples within the angle formed by  $P1a$  and  $P1b$ . In Figure 1(b), the receiving range is all samples within the angle formed by  $P2c$  and  $P2d$ . These points within the receiving range of point  $P$  can pass their labels to point  $P$ . It is concluded that when the distribution of neighbor points on both sides of the main direction is dense, the receiving range should be reduced, and when the distribution of neighbor points on both sides is sparse, the receiving range should be enlarged.

**3.2. Adaptive Angle.** In the previous section, we mentioned that DBC algorithm sets the *angle* as a global parameter. For different datasets, it can also achieve the ideal clustering effect after adjusting the parameter many times. However, there are also problems in the experimental process. The two parameters  $k$  and *degree* will affect each other. If  $k$  at the beginning is too large or too small for the current test dataset, it may require multiple attempts to achieve the expectation when setting the second parameter *degree*. So it will increase the uncertainty of clustering time. For this shortcoming, we introduce the concepts of high-density and low-density regions to explain the ALR-DBC algorithm.

In the sample set with class labels, we can find the obvious rule that the cluster distribution divided into one category is relatively dense, which can be represented by high-density regions. There will be sparsely distributed sample points between the two clusters, that is, the low-density region we want to refer to. In this way, we define the

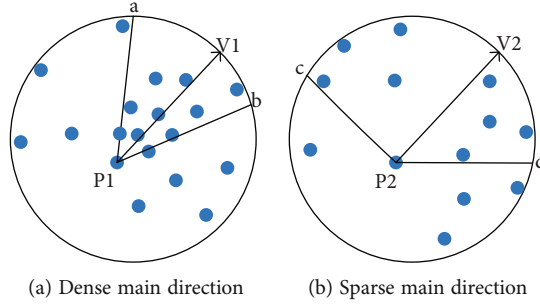


FIGURE 1: The receiving range for the dynamic selection of  $P1$  and  $P2$ .

distribution states of high-low-high and divide clusters. As shown in Figure 2, our goal is to find a continuous high-density area as a receiving area for point  $P$ .

The next question we need to think about is how to find the criteria for distinguishing high- and low-density regions. From the above, it can be seen that the main receiving direction is easily determined by formula (1). Then, based on this direction, it is feasible to use the ratio definition method to divide regions. When the condition of  $A/B \geq M$  is satisfied, it is included in the receiving range of this point.

$$A = \sum_{\vec{v}_i \cdot \vec{v}_j \geq 0.5} \vec{v}_i \cdot \vec{v}_j. \quad (2)$$

All neighbor vectors of point  $P$  are stored in list  $V$ . As shown in formula (2), we traverse each vector  $V_i$  in list  $V$ , taking the dot product of each vector with the rest of its neighbors and summing them up. So each neighbor vector will get the corresponding  $A$  value. The value 0.5 in the constraint condition is consistent with the DBC algorithm.  $B$  is the value  $A$  of vector  $PV_1$  in Figure 2. Because each point is receiving in a unique direction, the value  $B$  is fixed after locking a point  $P$ .  $M$  is a receiving threshold we set artificially.

We analyzed the scanning process according to the discriminant conditions. The idea of ALR-DBC algorithm is to scan the neighbor vectors from the main direction vector of point  $P$  to both sides and judge them in turn. The points that meet the receiving threshold on both sides of the main direction are put into the receiving range list of point  $P$ . Because the density distribution of each dataset is different, the advantage is that we will judge whether the direction currently scanned meets the receiving condition, rather than using fixed angle to define the range. Here is the dynamic adjustment process of the algorithm. The final receiving range includes all high-density direction vectors without crossing the low-density region. The specific scan is shown in Figures 3 and 4, showing the neighbor distribution of point  $P$  and the dynamic change process of one side of the main direction.

To help understand how the ALR-DBC algorithm dynamically selects the receiving angle of each point, a partial scanning process is drawn. As shown in Figure 4, point  $P$  finally finds a receiving boundary on the clockwise side that is a vector from  $p$  to  $c$ . Judge from vector  $a$  with the

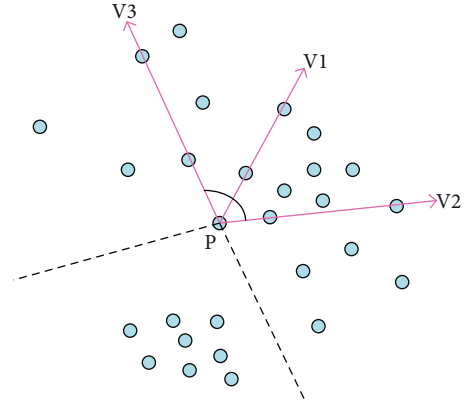


FIGURE 2: Cluster partition at high and low density.

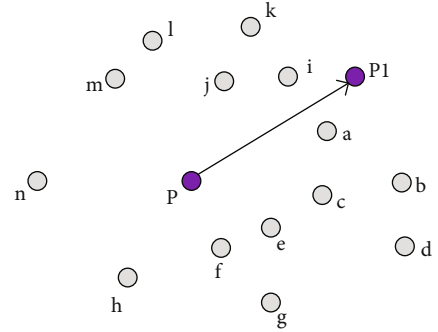


FIGURE 3: The main receiving direction and neighbor distribution of point  $P$ .

minimum angle away from the main direction. Since the vector meets the conditions that  $A/B \geq M$ , it is added to the receiving list of point  $P$ . Then continue scanning to point  $i$  in Figure 4(b) with the second smaller angle from the main direction. Repeat the above discrimination steps. Until the unqualified vector is found as shown in Figure 4(d), the vector is saved and recorded as the boundary vector of one side (assumed as the  $X$  side) of point  $P$ .

The focus of the next algorithm is to find the boundary vector on the other side of point  $P$ . We continue to scan the neighbor vector outward and judge it by inequality. If the condition is met and the vector is not on the side of  $X$ , it is included in the receiving range of point  $P$ ; otherwise, it is not included and continues to scan. That is because if

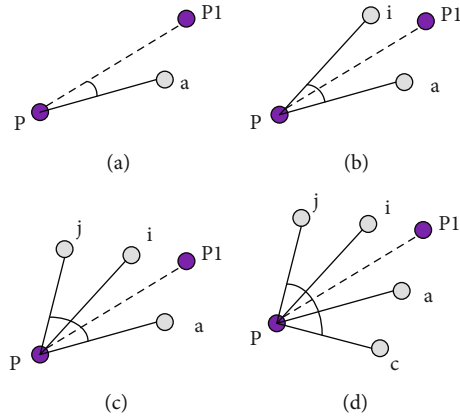


FIGURE 4: The process diagram of point  $P$  finding the receiving boundary on one side of the main direction.

this vector is on  $X$  side, it is already outside the reception boundary. Although it meets the density required by the receiving condition, it does not meet the continuous high-density area mentioned above. Then, when the scanned neighbor vector does not meet the density required by the receiving condition, if it is not on the side of  $X$ , we find the boundary vector on the other side of point  $P$  (assumed as the  $Y$  side), that is, the low-density region after the continuous high-density region. So far, we have set the receiving range of point  $P$ .

**3.3. Label Redistribution.** In describing the idea of the DBC algorithm, we mention that it can determine an edge of the receiving range by finding the receiving direction at point  $P$ . The other side is determined by the angle value, which allows neighbor vectors within a certain deviation angle to pass their class labels to point  $P$ . When the program runs, the ideal clustering results can be achieved by adjusting the two parameters. However, we need to make it clear that point  $P$  should best be grouped with its principal direction vector. In the original DBC algorithm, when a point becomes a new cluster label sender, it will immediately pass its class label to all its receivers. Assuming that the receiver's receiving direction label is different from the tag, the final clustering effect will be worse than expected. We illustrate this by using a spiral dataset.

As shown in Figure 5, it is the clustering diagram of the spiral dataset when only the receiving angle is dynamically adjusted. We observed that the points were divided into three categories and were spirally distributed. At the top of the figure, there is a blue sample point  $F$  next to the green sample point. Obviously, the sample point is wrongly classified at this time, and it should actually be classified as a green class. The reason for this result is that the receiving range of a blue point in the dataset includes  $F$ . At the beginning of each new round of clustering, the transfer of clustering labels starts from the unlabeled samples with the largest number of receivers. It can be found from Figure 5 that the closer to the center, the closer the spiral is, that is, the higher the density of points is. Therefore, the sample points classified as the blue class have labels earlier than the green class. Then,  $F$  is given the wrong class label in this round. When  $F$  has a

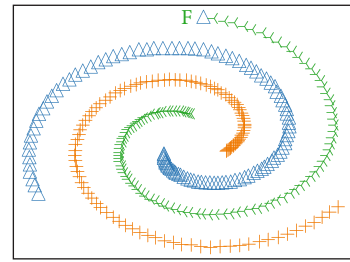


FIGURE 5: Clustering graph with only improved receiving angle ( $k = 8$ ).

label, the subsequent clustering process is no longer assigned, considering only the data points that are not currently labeled. We consider whether we can change the clustering order so that the points of the green class get the class label first. Although this ensures the correct classification of  $F$ , there are few receivers of this part of the points, which will force the points that should be classified into one category to be classified into multiple categories because they cannot be reached in a round of label transmission. Therefore, we adopt another improvement idea to solve this problem. After all points have class labels, all points are traversed to determine whether the current label is consistent with the label in the receiving direction. If not, the cluster label on the receiving direction is redistributed to that point. This is a search lookup and redistribution process. The class label of sample points is most likely to be consistent with its receiving direction, because the receiving direction represents the most densely distributed region of neighbors. These regions are also most likely to be clustered eventually.

**3.4. Algorithm Description.** Through the determination of the above adaptive angle, the receiving range of each point is obtained, and then, the DBC algorithm is used for clustering. After the first clustering, judge whether there are samples that have not been assigned labels. If so, start the second round of clustering. From the remaining samples without labels, continue to select the most current recipients as a new starting point to start a new round of label delivery. Continue the iteration until all points have class tags. To



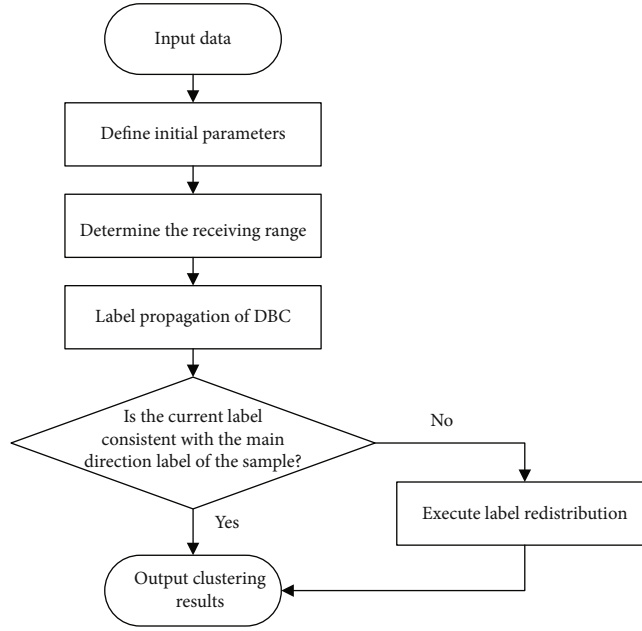


FIGURE 6: Flowchart of ALR-DBC.

have a clearer explanation, we provide the flowchart of the whole process as shown in Figure 6.

Algorithm 2 describes the process of determining the receiving angle of data point  $i$ . VecList stores all neighbor vectors of data point  $i$ . sList is the value computed by the neighbor vector of data point  $i$  according to formula (2). The dictionary  $a$  is arranged in descending order to represent the points that are scanned outward from the principal direction vector according to the increment of the angle value. MaxV takes the maximum value in sList. The Vector list stores the receiving direction vector of each point.  $M$  is the acceptance threshold mentioned in Section 3.2. The range list stores the sender within the current deviation range of point  $i$ .

In the fifth line, we determine the receiving direction of data point  $i$ . Lines 6 to 9 compute the dot product of the received direction vector of data point  $i$  and all its neighbor vectors and store it in dictionary  $a$ . Lines 12 to 19 determine the receiving range of one side. Lines 20 to 28 determine the receiving range on the other side.

Then, we use the DBC algorithm for clustering. The obtained clustering results are stored in the Result list. At this point, we begin to redistribute cluster labels. The existing label of each point is compared with the label of its receiving direction. If the labels do not match, the existing label for that point is modified.

**3.5. Parameter Selection.** The first parameter to be set in the algorithm is  $k$ . It is the number of  $k$  neighbors of the sample point that are closest to it. In general,  $k$  is between 5 and 30, and some datasets need larger  $k$ . At this time,  $k$  is between 30 and 60.  $M$  is a low-sensitivity parameter. When  $M$  takes an increasing value, it makes the receiving angle smaller, and then, the final number of clusters becomes larger. In this paper, a large number of experiments show that the cluster-

ing effect for most datasets is ideal when  $M$  is between 0.6 and 0.7. It is easier to determine than the angle parameter of DBC algorithm.

## 4. Experimental Result

This chapter evaluates the clustering effect of ALR-DBC algorithm. Comparison algorithms are DBC algorithm, FCM algorithm, and DBSCAN algorithm. Since FCM algorithm is a popular fuzzy clustering algorithm, DBSCAN algorithm is a classical density-based clustering algorithm. Therefore, as a comparison, it can prove the superiority of ALR-DBC algorithm. Test datasets include artificial datasets and real datasets. Artificial datasets are Flame, Threecircles, Twomoons, Aggregation, Lsun, and Hard [17]. Real datasets include Iris [26], Dermatology [27], Balance, Vote, and Vowel. The introduction to these artificial datasets is listed in Table 1.

The experimental environment is AMD Ryzen 5 4600 H @ 3.00 GHz. The memory is 16 GB. The programming environment is Python 3.8 and the compiler is PyCharm.

Normalized interactive information (NMI) [28], adjusted Rand index (ARI) [29], and Homogeneity are selected as the evaluation indexes of clustering.

**4.1. Artificial Dataset.** In Table 1, Flame is a dataset with overlapping regions, which can test whether the improved algorithm can have good clustering results for such datasets. Threecircles is a typical representative of the nonconvex dataset, and our algorithm can cluster well. The Hard dataset is characterized by large density differences between clusters. Aggregation and Twomoons datasets are composed of irregular clusters. The Lsun dataset is composed of clusters with uneven density distribution. In the original DBC algorithm, through the continuous adjustment of two parameters, we

```

1: //All points in the sample set are traversed to
   //dynamically determine their receiving range
2: for  $i$  from 1 to the maximum number of dataset do
3:   Compute sList
4:    $MaxV = \text{Max}(sList)$ 
5:    $l = \text{Max}(sList).label$ 
6:    $Vector[i] = \text{VecList}[l]$ 
7:   for  $j = 0; j < \text{VecList.size}; j++$  do
8:      $a1.append(\text{Vector}[i] \cdot \text{VecList}[j])$ 
9:      $a = a1.sorted(\text{reverse} = \text{true})$ 
10:  end for
11:  for each  $u \in a$  do
12:    if  $sList[u]/MaxV > M$  then
13:       $\text{Range}[i].append(u)$ 
14:    else
15:       $\text{Boundary} = \text{VecList}[u]$ 
16:       $R = u$  and exit this cycle
17:    end if
18:  end for
19:  for  $n$  from  $R$  to the remaining points do
20:     $D = sList[n]/MaxV$ 
21:    if  $D/MaxV > M$  and  $n$  not in the side of boundary then
22:       $\text{Range}[i].append(n)$ 
23:    else if  $D/MaxV < M$  and  $n$  not in the side of boundary then
24:      Break
25:    else
26:      Continue
27:    end if
28:  end for
29: end for

```

ALGORITHM 2: Receiving angle.

TABLE 1: Artificial datasets for testing.

Name	Instances	Clusters
Flame	240	2
Threecircles	299	3
Twomoons	1502	2
Aggregation	788	7
Lsun	400	3
Hard	1501	3

can obtain better clustering results of these six datasets. Figure 7 shows the clustering effect of the ALR-DBC algorithm when only one parameter is adjusted.

When the DBC algorithm executes the parameter set in Table 2, the optimal NMI values can be achieved. On these artificial datasets, the ALR-DBC algorithm achieves the best NMI of DBC algorithm under the condition that  $M$  is 0.6. For some datasets, even a small range of fluctuations in the angular parameters of the DBC algorithm can affect the final clustering results. We can analyze it in Table 3.

In Table 3, we use the Compound dataset as the experimental subject. The NMI changes are observed by adjusting the parameters. In the DBC algorithm, when  $k$  is 9 and *degree* is 90, NMI can reach 0.8812. However, when the angle value is 86 degrees, it can be seen that the NMI value

is reduced to 0.8170. The number of points classified wrong increases, and their density distribution is uniform. At the same angle value, the decrease of  $k$  from 9 to 8 also leads to a decrease in NMI. Since the dataset is two-dimensional, the distance to observe these points is also very close. It is precisely because the change of angle value leads to the deviation of experimental results.

To better illustrate the influence of parameters in the ALR-DBC algorithm, we performed a parameter sensitivity test on the Compound dataset. NMI was taken as the evaluation index, as shown in Table 4. We can find that when  $M$  gradually increases from 0.60, NMI does not fluctuate significantly.

**4.2. Real Dataset.** The ALR-DBC algorithm shows good clustering effect on the artificial dataset in the previous section. In this section, we use real high-dimensional datasets to further test its clustering effect and compare with other algorithms. In the experiment, NMI, ARI, and Homogeneity are used to evaluate the performance. Their ranges are all  $[0, 1]$ . The larger the value in this interval is, the closer the clustering result is to the real label, and the better the effect is. The UCI dataset used in the experiment is shown in Table 5.

For different algorithms, it is necessary to set its iterative method according to the characteristics of parameters to find the best evaluation index. For FCM algorithm, input the number of clusters and run the program many times.

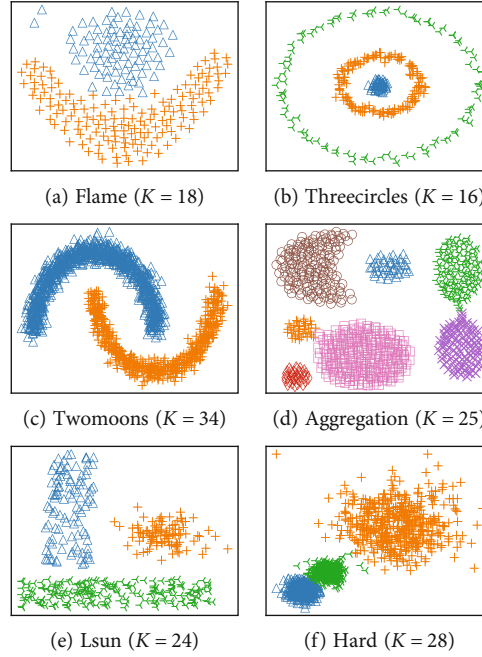


FIGURE 7: Clustering results obtained by ALR-DBC on six datasets (the experimental parameter  $k$  values for each dataset are given, and  $M$  is 0.6 for all).

TABLE 2: The best experimental results of DBC algorithm.

Name	$k$	Degree	NMI
Flame	18	35	0.9269
Threecircles	16	60	1.0
Twomoons	34	80	1.0
Aggregation	20	60	0.9956

TABLE 3: Experimental results of DBC on Compound.

$k$	Degree	NMI
9	90	0.8812
9	86	0.8170
8	90	0.8285
10	90	0.8768
10	80	0.8051

TABLE 4: Experimental results of ALR-DBC on Compound.

$k$	$M$	NMI
9	0.60	0.8977
9	0.62	0.8977
9	0.64	0.8977
9	0.66	0.8475
9	0.68	0.8475

When the clustering results do not reach better within at least 20 times, it proves that we have obtained the desired experimental results. For DBSCAN, we set a reasonable iteration range for two parameters  $e$  and MinPts, respectively.

TABLE 5: UCI datasets for testing.

Name	Instances	Attributes	Clusters
Iris	150	4	3
Dermatology	358	34	6
Balance	625	4	3
Vote	435	16	2
Vowel	990	13	11
Landsat	2000	36	6
Ecoli	336	7	8
WDBC	569	30	2

In our experiment,  $e$  is traversed from 0.01 to 1, and MinPts traverses from 1 to 30. For the improved algorithm ALR-DBC, the adjustment range of parameter  $k$  is 3 to 50. Then, we get three groups of evaluation index of each algorithm as shown in Table 6.

In general, the clustering effect of the DBC algorithm and ALR-DBC algorithm in these eight sets of data is better. The ALR-DBC algorithm uses the advantage of adaptive angle to divide the receiving range of sample points under different distributions. It works best in Iris, Dermatology, and Balance datasets. The two evaluation indexes of ALR-DBC algorithm and DBC algorithm are consistent and optimal on Vote dataset and close to the optimal clustering effect on Landsat dataset. Compared with the DBC algorithm, the ALR-DBC algorithm performs better in all indicators on five datasets. The FCM algorithm performs better on WDBC dataset because it has better adaptability for datasets with large density difference and overlapping between clusters. This is the advantage that other comparison algorithms do not have.

TABLE 6: Comparison of clustering effects on the UCI datasets.

Dataset	Criteria	DBSCAN	FCM	DBC	ALR-DBC
Iris	NMI	0.7336	0.7433	0.8705	0.8980
	ARI	0.5681	0.7287	0.8857	0.9221
	Homogeneity	0.5793	0.7404	0.8696	0.8980
Dermatology	NMI	0.6205	0.7644	0.8102	0.8486
	ARI	0.4151	0.6866	0.7307	0.7871
	Homogeneity	0.6484	0.6878	0.7859	0.7879
Balance	NMI	0.0178	0.0093	0.2082	0.2104
	ARI	0.0200	0.0075	0.0424	0.0440
	Homogeneity	0.0101	0.0083	0.5125	0.5181
Vote	NMI	0.3977	0.4547	0.4942	0.4942
	ARI	0.4480	0.5232	0.5709	0.5709
	Homogeneity	0.5034	0.4633	0.5030	0.5030
Vowel	NMI	0.5317	0.5420	0.5576	0.5693
	ARI	0.4170	0.4004	0.4221	0.4579
	Homogeneity	0.4902	0.5503	0.5242	0.5199
Landsat	NMI	0.5768	0.6054	0.6415	0.6365
	ARI	0.4153	0.5216	0.5966	0.6306
	Homogeneity	0.5026	0.6091	0.6667	0.6395
Ecoli	NMI	0.6161	0.5448	0.6550	0.6571
	ARI	0.1459	0.3620	0.7009	0.7165
	Homogeneity	0.6285	0.6431	0.5558	0.5580
WDBC	NMI	0.3790	0.6151	0.5203	0.5258
	ARI	0.4661	0.7304	0.5189	0.5883
	Homogeneity	0.3868	0.6080	0.6528	0.6656

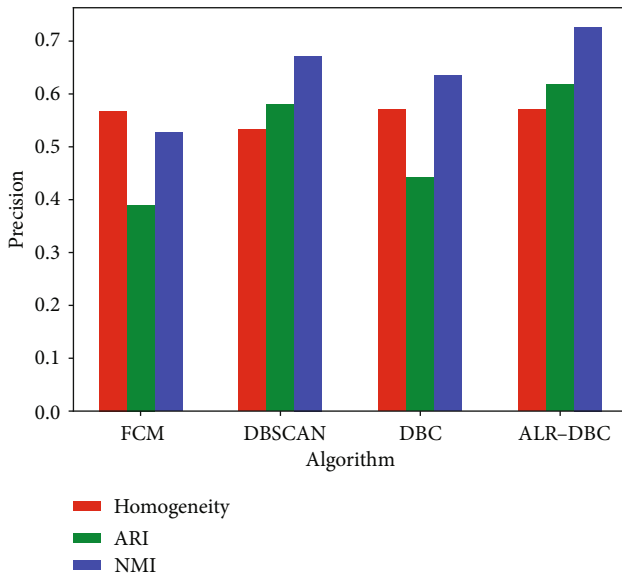


FIGURE 8: Performance comparison of different algorithms.

**4.3. Application.** Our improved algorithm can also be applied to wireless sensor data annotation in IoT. In this section, a set of high-risk behavior monitoring data of elderly volunteers in clinical activities is used to verify the effectiveness of ALR-DBC algorithm in practical application. The perception layer of the IoT is covered by various sensors and sensor gateways. Their function is to identify objects and collect information. The test data we used were provided by the research in literature [30] and classified by clinical activity status. Through ALR-DBC algorithm, we obtain the class labels of activity data in all monitoring time periods and compare the evaluation indicators with other algorithms mentioned above, as shown in Figure 8. Experiments show that our method has a good application effect in wireless sensor data annotation of the IoT.

## 5. Conclusions

In this paper, we reduce the number of parameters of the DBC algorithm through the strategy of adaptive angle, and the problem of misclassification caused by the order of scanning points in the clustering process is solved by the method of redistribution of cluster labels. In the experimental process, we found that it can also well separate clusters with large density difference and nonuniformity. It shows good



clustering effect on artificial datasets. In some UCI datasets, it can surpass the clustering effect of the original algorithm. We apply the improved algorithm to wireless sensor data annotation. Good application effect can be obtained through experiments. For those datasets with more overlapping regions between different clusters, although the evaluation metrics have been improved, they still cannot achieve the desired results. In future research, we intend to improve the ALR-DBC algorithm by combining the knowledge of depth measurement learning. We will also discuss its prospects in the field of IoT to make it more effective.

## Data Availability

In this paper, the experiments using real datasets are available at url = "http://archive.ics.uci.edu/ml". References have been marked at corresponding positions in the article.

## Conflicts of Interest

The authors declare that there is no conflict of interest in publishing this article.

## Acknowledgments

This research was supported by the National Natural Science Foundation of China (61962054).

## References

- [1] A. Rebiai, B. B. Seghir, H. Hemmami et al., "Clustering and discernment of Algerian bee pollen using an image analysis system," *Algerian Journal of Chemical Engineering*, vol. 1, no. 2, pp. 41–48, 2021.
- [2] K. Vantas and E. Sidiropoulos, "Knowledge discovery using clustering analysis of rainfall timeseries," in *EGU General Assembly Conference Abstracts*, pp. 21–14758, Vienna, Austria, 2021.
- [3] N. Han, S. Qiao, G. Yuan, P. Huang, D. Liu, and K. Yue, "A novel Chinese herbal medicine clustering algorithm via artificial bee colony optimization," *Artificial Intelligence in Medicine*, vol. 101, article 101760, 2019.
- [4] A. Caggiano, F. Napolitano, and R. Teti, "Hierarchical cluster analysis for pattern recognition of process conditions in die sinking edm process monitoring," *Procedia CIRP*, vol. 99, no. 2, pp. 514–519, 2021.
- [5] M. Du, S. Ding, and H. Jia, "Study on density peaks clustering based on  $k$ -nearest neighbors and principal component analysis," *Knowledge-Based Systems*, vol. 99, pp. 135–145, 2016.
- [6] F. A. Ferdous, "A conceptual review on different data clustering algorithms and a proposed insight into their applicability in the context of COVID-19," *Journal of Advances in Technology and Engineering Research*, vol. 6, no. 2, pp. 58–68, 2020.
- [7] S. C. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, no. 3, pp. 241–254, 1967.
- [8] A. Dogan and D. Birant, "K-centroid link: a novel hierarchical clustering linkage method," *Applied Intelligence*, vol. 52, no. 5, pp. 5537–5560, 2022.
- [9] J. A. Hartigan and M. A. Wong, "Algorithm as 136: a  $k$ -means clustering algorithm," *Journal of the Royal Statistical Society*, vol. 28, no. 1, pp. 100–108, 1979.
- [10] L. Y. Tseng and S. B. Yang, "A genetic clustering algorithm for data with non-spherical-shape clusters," *Pattern Recognition*, vol. 33, no. 7, pp. 1251–1259, 2000.
- [11] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.
- [12] Q. Zhang and X. Chen, "Agglomerative hierarchical clustering based on affinity propagation algorithm," in *2010 Third International Symposium on Knowledge Acquisition and Modeling*, pp. 250–253, Wuhan, 2010.
- [13] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A densitybased algorithm for discovering clusters in large spatial databases with noise," *AAAI Press*, vol. 96, no. 34, pp. 226–231, 1996.
- [14] Y. Chen, L. Zhou, N. Bouguila, C. Wang, Y. Chen, and J. du, "BLOCK-DBSCAN: fast clustering for large scale data," *Pattern Recognition*, vol. 109, article 107624, 2021.
- [15] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.
- [16] Y. Li, W. J. Zhou, and H. K. Wang, "F-DPC: fuzzy neighborhood-based density peak algorithm," *IEEE Access*, vol. 8, pp. 165963–165972, 2020.
- [17] N. Xiao, K. Li, X. Zhou, and K. Li, "A novel clustering algorithm based on directional propagation of cluster labels," in *2019 International Joint Conference on Neural Networks*, pp. 1–8, Budapest, Hungary, 2019.
- [18] X. S. Xue, X. J. Wu, C. Jiang, G. Mao, and H. Zhu, "Integrating sensor ontologies with global and local alignment extractions-financial big data based on Internet of things and wireless network communication," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 8944618, 12 pages, 2021.
- [19] X. S. Xue and Q. H. Huang, "Generative adversarial learning for optimizing ontology alignment," *Expert Systems*, vol. 39, pp. 1–12, 2022.
- [20] X. S. Xue and J. Zhang, "Matching large-scale biomedical ontologies with central concept based partitioning algorithm and adaptive compact evolutionary algorithm," *Applied Soft Computing*, vol. 106, article 107343, 2021.
- [21] J. G. Liang, X. F. Zhou, Y. Sha, P. Liu, L. Guo, and S. Bai, "Unsupervised clustering strategy based on label propagation," in *IEEE International Conference on Data Mining Workshops*, pp. 788–794, Dallas, TX, USA, 2013.
- [22] X. J. Zhu, *Semi-Supervised Learning with Graphs*, Carnegie Mellon University ProQuest Dissertations Publishing, 2005.
- [23] J. X. Shi and L. X. Zhang, "Label propagation algorithm based on potential function for community detection," *Journal of Computer Applications*, vol. 34, no. 3, p. 738, 2014.
- [24] M. Y. Shi, Y. Zhou, and Y. Xing, "Community detection by label propagation with leaderrank method," *Journal of Computer Applications*, vol. 35, no. 2, p. 448, 2015.
- [25] N. Y. Chen, Y. Liu, H. Q. Chen, and J. Cheng, "Detecting communities in social networks using label propagation with information entropy," *Physica A: Statistical Mechanics and its Applications*, vol. 471, pp. 788–798, 2017.
- [26] P. Zhong and M. Fukushima, "Regularized nonsmooth newton method for multi-class support vector machines," *Optimization Methods and Software*, vol. 22, no. 1, pp. 225–236, 2007.
- [27] N. Rajkumar and P. Jaganathan, "A new RBF kernel based learning method applied to multiclass dermatology diseases classification," in *2013 IEEE Conference on Information & Communication Technologies*, Thuckalay, India, 2013.

- [28] T. P. Q. Nguyen and R. J. Kuo, "Partition-and-merge based fuzzy genetic clustering algorithm for categorical data," *Applied Soft Computing*, vol. 75, 2018.
- [29] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: variants, properties, normalization and correction for chance," *The Journal of Machine Learning Research*, vol. 11, pp. 2837–2854, 2010.
- [30] A. P. Sample, R. S. Torres, D. C. Ranasinghe, Q. Shi, and A. P. Sample, "Sensor enabled wearable rfid technology for mitigating the risk of falls near beds," in *IEEE International Conference on RFID*, Orlando, FL, USA, 2013.

## Research Article

# A Joint Model of Natural Language Understanding for Human-Computer Conversation in IoT

Rui Sun <sup>1</sup>, Lu Rao <sup>2</sup>, and Xingfa Zhou <sup>2</sup>

<sup>1</sup>School of Artificial Intelligence, Leshan Normal University, Leshan, China

<sup>2</sup>AI Lab, Sichuan Changhong Electric Appliance Co., Ltd, Chengdu, China

Correspondence should be addressed to Lu Rao; [lu.rao@changhong.com](mailto:lu.rao@changhong.com)

Received 1 March 2022; Revised 7 July 2022; Accepted 2 August 2022; Published 21 August 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Rui Sun 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.

Natural language understanding (NLU) technologies for human-computer conversation is becoming a hot topic in the Internet of Things (IoT). Intent detection and slot filling are two fundamental NLU subtasks. Current approaches to these two subtasks include joint training methods and pipeline methods. Whether treating intent detection and slot filling as two separate tasks or training the two tasks as a joint model utilizing neural networks, most methods fail to build a complete correlation between the intent and slots. Some studies indicate that the intent and slots have a strong relationship because slots often highly depend on intent and also give clues to intent. Thus, recent joint models connect the two subtasks by sharing an intermediate network representation, but we argue that precise label information from one task is more helpful in improving the performance of another task. It is difficult to achieve complete information interaction between intent and slots because the extracted features in existing methods do not contain sufficient label information. Therefore, a novel bidirectional information transfer model is proposed in order to create a sufficient interaction between intent detection and slot filling with type-aware information enhancement. Such a framework collects more explicit label information from the network's top layer and learns discriminative features from labels. According to the experimental results, our model greatly outperforms previous models and achieves the state-of-the-art performance on the two datasets: ATIS and SNIPS.

## 1. Introduction

The definition of Internet of Things (IoT) is the network where devices or sensors deploy in physical environments using intelligent interfaces to connect and communicate within different user scenarios [1, 2]. Recent studies show that the natural language will become the primary interactive mode between people and devices in IoT. Human-computer conversation technologies can be used to connect people and a variety of objects in the network, which include natural language understanding, knowledge graph, and semantic web. To combine semantic web technologies with IoT adaptively, Xue et al. [3–5] propose some algorithms to match sensor ontologies for the purpose of implementing the semantic interoperability among intelligent sensor applications. Natural language understanding technologies are also widely utilized in IoT.

Natural language understanding (NLU) is an essential component computer conversation system of IoT, which generally includes intent detection and slot filling. Both tasks focus on determining the user's intention and collect critical constituents via annotating the utterance. There is an example from the ATIS dataset shown in Figure 1. The utterance "I need a flight from los angeles to charlotte today" is annotated by slot labels on a word-level, while intent detection gives at least one intent label to the whole sentence.

Intent detection and slot filling are naturally defined as two separate tasks [6]. Intent detection refers to the classification problem with the method of machine learning such as support vector machines (SVMs) [7] and deep learning frameworks [8–11]. In addition, intent detection utilized in IoT is normally based on keyword extraction. Slot filling can be treated as a sequence labeling task. Conditional random fields (CRF) [12, 13], maximum entropy Markov models (MEMMs) [14], and

Sentence	I	need	a	flight	from	los	angeles	to	charlotte	today
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Slot	O	O	O	O	O	B-fromloc	I-fromloc	O	B-toloc	B-depart_time
Intent	Atis_flight									

FIGURE 1: The illustration of an utterance annotated with slot labels and intent label. Three rows represent the input sequence, the corresponding slot labels, and the intent label of the input sequence, respectively.

recurrent neural networks (RNNs) [15–19] are widely adopted to forecast the labels of slot. Furthermore, ontology-based methods are also leveraged for slot filling in IoT. The pipeline methods can be used to separately perform the two tasks, but such frameworks may cause error propagation.

To solve the problems caused by pipeline manners, extracting the slot information and determining the utterance’s intent are performed with joint learning methods. Previously, neural networks are used to share the sentence-level features between the two subtasks [20–23]. In addition, an attention-based RNN method is proposed to provide additional information to slot filling and intent detection [24]. Although such approaches avoid the problem of error propagation, the interaction of the two tasks is not considered. In fact, the identified intent information may provide clues to slot filling and vice versa. For example, if the utterance is recognized as intent “atis\_flight,” it is more likely for the word “charlotte” to have a slot of “toloc” than other labels such as “personal\_name.” On the contrary, with the word “charlotte” is identified as “toloc” and “today” is identified as “depart\_date,” the utterance is more likely to be annotated by an intent label “atis\_fight” rather than “atis\_distance.” Thus, it can be seen that slots and intent are interactive.

Some existing works learn to establish interaction between slots and intent. Goo et al. [25] apply the intent context vector to the LSTM layer via a slot-gated mechanism. However, the intent detection task does not utilize the slot information, and the information interaction is unidirectional. The Capsule Neural Network [26, 27] is used to maintain hierarchical relationships in slots, intentions, and words [28]. Besides, the association between slots and intent is improved by a SF-ID network [29]. Wang et al. [30] also design a Bi-model structure to perform slot filling and intent detection jointly. These methods utilize network’s intermediate information to build correlation between slots and intent. However, we argue that the information extracted from above approaches represents the sentence features and it is insufficient to express the label information. And the joint model performance can be improved by the specific label probability distribution. Extracting sufficient features is difficult between the two tasks. Therefore, it has become a challenge in recent research to extract the explicit label information and establish a complete correlation between the intent and slots. Unlike the previous approaches, our model extracts information from network’s top layer, which preserves more explicit label information.

In this paper, we propose a bidirectional information transfer model for joint intent detection and slot filling with

type-aware information enhancement. This model seeks to respond to the problem that most approaches do not build complete correlation between the intent and slots. Firstly, to learn the discriminative features from the slots and intent labels, we provide a type-aware mechanism. Secondly, to improve the connection between intent and slots, a unique bidirectional information transfer method is devised that takes advantage of label probability distribution from the network’s top layer. As a result, more precise information of labels is collected for propagation. Furthermore, a tagging scheme is introduced to diminish the number of slot types for slot filling. The training process is faster and more efficient compared to the “BIO” format annotation.

We compare our method with some published state-of-the-art models on ATIS [31] dataset and SNIPS dataset. Experimental results show the effectiveness of our framework, which outperforms most of current state-of-the-art models. Especially in slot filling and sentence accuracy, our model achieves 1.3% improvement on SNIPS dataset and 1.2% absolute gain on ATIS dataset. In addition, we believe that we are the first to use label probability distribution collected from network’s top layer for predicting slot and intent jointly.

The remainder of this paper is arranged as follows. Firstly, this paper begins by defining the problem statement for our framework in Section 2. Afterwards, our proposed framework is presented in Section 3. Then we introduce the experimental design, present the findings, and analyze our framework in Section 4. The related work is stated in Section 5. Finally, we conclude our work in Section 6 and discuss the direction of future work in Section 7.

## 2. Problem Statement

We consider the intent detection as an utterance-level classification task, while slot filling refers to the token-level classification task. We forecast the intent label  $y^{\text{intent}}$  and a sequence of slot labels  $(y_i^{\text{start}}, y_i^{\text{end}})$  as outputs from the input sentence  $\{x_1, x_2, \dots, x_T\}$  with  $T$  tokens. Especially for slot filling, we design a simple tagging scheme to reduce the number of slot categories, which will be introduced in detail in the next section. The objective is to reduce the discrepancy between the ground-truth data and estimated outputs.

## 3. Approach

In this section, our bidirectional information transfer framework will be presented in detail. Figure 2 gives an overview

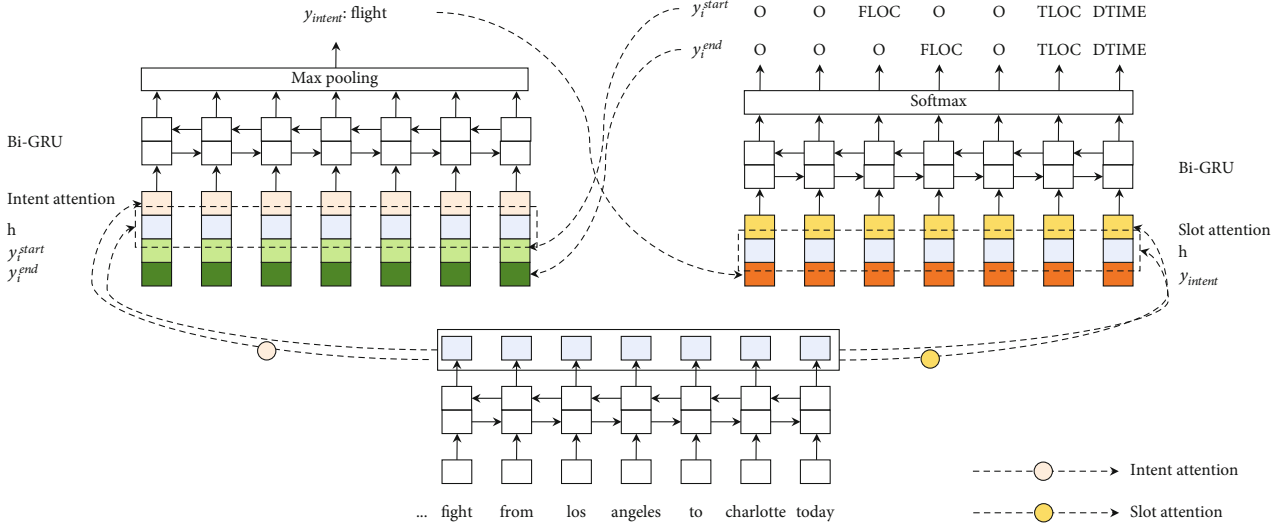


FIGURE 2: Framework of our bidirectional information transfer method. The framework is comprised of three main parts: the shared encoder, the decoder slot filling, and the intent detection. The type-aware method works on the bottom layer of the two decoders, respectively. The bidirectional information transmission module is utilized between the two decoders, which propagates tag information from top layers of the two decoders. For example, the probability distribution of intent label  $y_{i_{\text{intent}}}$  is transmitted to the bottom layer of slot filling decoder. And the probability distribution of slot labels  $[y_i^{\text{start}}, y_i^{\text{end}}]$  is transmitted to the bottom layer of intent detection decoder simultaneously.

of our approach. We can see that the intent detection is transformed into a classification problem, while the slot filling is transformed into two sequence labeling problems. Following that, the tagging scheme is introduced firstly. Then we discuss the input representation for the out-of-vocabulary (OOV) problem and how to enhance the input semantics representation. In addition, a graph recurrent unit (GRU) network is simply introduced, which is utilized as the main layer in our framework. A type-aware information enhancement is used to improve slot filling and intent detection in detail. Lastly, the bidirectional information transfer scheme is presented, and a joint training method is used to optimize both tasks simultaneously.

**3.1. Tagging Scheme.** Let us consider the slot filling task's tagging scheme first. The slot filling is transformed into two sequence labeling [32] tasks, where the inspiration comes from the Pointer Network [33]. We simplify the tasks as Figure 3 shown.

Finding the start position of the entity in an utterance is the primary objective of first task. If a token is the first word in an entity span, it will be annotated with associated slot type. If the token is not the first word, assigning the token with the label "O" (Outside) means it has no significance for the "start sequence labeling." On the contrary, the second task seeks to determine the end position of the entity in an utterance. The labeling process of "end sequence labeling" is similar to the "start sequence labeling," where the difference is to find the end position of an entity span in an utterance.

Figure 3 gives a sample of the tagging method. The words "los," "angeles," "charlotte," and "today" are tagged with the corresponding slot type label using our tagging scheme. Our tagging method is obviously distinct from the

"BIO" tagging format. Because we only estimate an entity span's beginning and end position, there are fewer number of slot types. This indicates that our framework is more effective and time-saving. The training process using our tagging method will be discussed in detail in the subsection Slot Filling with Type-aware Information Enhancement.

**3.2. Input Representation Layer.** A common way to incorporate context information of words is to use input representations learned from unannotated corpora. For most previous studies, word embedding is utilized as a direct input for most language tasks, but it is unable to address the out-of-vocabulary (OOV) problem. As characters are shared across words, the input representation layer maps input sentences into vectors via concatenating word-level embedding and character-level embedding. In this way, unknown words can be generated using their component characters. In addition, we utilize a layer of CNN and a MaxPooling layer to incorporate word character sequence embedding into a dense vector.

The input sequence  $\{x_1, x_2, \dots, x_T\}$  represents the  $i_{\text{th}}$  word of a sentence with  $T$  words. The character-level embedding is represented by  $e_i^c$ , and the word-level embedding is represented by  $e_i^w$ . Thus, the expression of input representation is  $e_i = [e_i^w, e_i^c]$ ,  $e_i \in R^{dc+dw}$ , where  $dc$  is the character-level embedding dimension and  $dw$  is the word-level embedding dimension.

**3.3. Bidirectional Graph Recurrent Unit (GRU).** The graph recurrent unit (GRU) network is leveraged in the encoder and decoder to extract contextual information and semantic features of an utterance. The GRU network is first proposed by [34] to consider sequence labeling tasks, which has a simpler framework than long-short temporary memory (LSTM) network. The formulation is written as follows [34]:



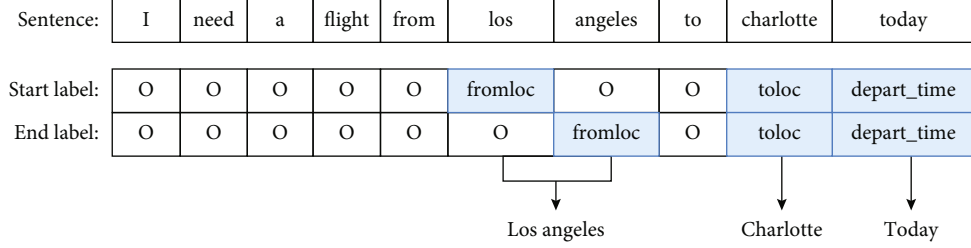


FIGURE 3: A case of tagging method utilized in our framework. Three rows are the input sequence, the star label of an entity span, and the end label of an entity span. In addition, if a word is neither the start word of an entity span nor the end word of an entity span, it will be annotated with the label “O”(Outside).

$$z_i = \sigma(W_z \cdot [h_{i-1}, x_i]), \quad (1)$$

$$r_i = \sigma(W_r \cdot [h_{i-1}, x_i]), \quad (2)$$

$$\tilde{h}_i = \tanh(W \cdot [r_i * h_{i-1}, x_i]), \quad (3)$$

$$h_i = (1 - z_i) * h_{i-1} + z_i * \tilde{h}_i, \quad (4)$$

where  $\sigma$  denotes the sigmoid function,  $r_i$  is the reset gate, and  $z_i$  represents the update gate. The reset gate decides how to integrate incoming input with the prior memory, while the update gate is used to specify how much past memory is preserved to the current time step. Such two gates allow information in long-term sequences to be preserved and not to be cleared over time.

**3.4. Intent Detection with Type-Aware Information Enhancement.** Normally the detection of intent is viewed as a classification task. Recent methods begin to use deep learning frameworks to accomplish this task [8–11]. Some of them utilize the attention mechanism [35] to focus on partial features. Type information, according to our research, is useful in modeling the learning of discriminative features. So in this paper, to effectively utilize the type information, a straightforward but efficient method named type-aware attention mechanism is suggested. Afterwards, we will detail this mechanism in the intent detection task. As shown in Equation (2),  $W_{\text{intent}}$  denotes the intent type, and  $\alpha_i$  represents the weight of intent attention. For each token, we obtain the hidden state  $h_i$  of type-aware intent, which is shown as below:

$$\alpha_i = \text{softmax}(e_i U W_{\text{intent}}^T), \quad (5)$$

$$h_i^{\text{context}} = \alpha_i W_{\text{intent}}, \quad (6)$$

$$h_i = [h_i^{\text{context}}, e_i], \quad (7)$$

where  $W^T \in R^{N \times d}$  denotes the trainable weight matrix,  $d_i$  represents the information vector dimension of intent category, and  $N$  denotes the quantity of intent categories.  $U$  represents the trainable matrix parameter.

A bidirectional GRU layer is applied to  $h_i$  to integrate the utterance representation and intent category information. The input is mapped to each intent category using a linear layer, as shown in Equations (8)–(10). By sharing the  $W_{\text{intent}}$  matrix, we create a link between the stage of intent detection and the

initial layer in Equations (5) and (10).

$$h_i^{\text{intent}} = \text{BiGRU}(h_i), \quad (8)$$

$$h^{\text{intent}} = \text{MaxPooling}\left([h_0^{\text{intent}}, \dots, h_n^{\text{intent}}]\right), \quad (9)$$

where  $h^{\text{intent}}$  represents intent hidden state.

The intent detection is considered as a single-label classification task; thus, the softmax function is used to compute the probability distribution  $y_{\text{intent}}$  of the intent label:

$$y_{\text{intent}} = \text{softmax}\left(h^{\text{intent}} W_{\text{intent}} + b_{\text{intent}}\right), \quad (10)$$

where  $W_{\text{intent}}$  and  $b_{\text{intent}}$  are the trainable matrix parameters.

**3.5. Slot Filling with Type-Aware Information Enhancement.** Slot filling, like intent detection, also utilizes our type-aware method to extract distinguishing features from slot category. In the process of slot filling, the parameter of start slot category is represented by  $W_{\text{slot}}^{\text{start}}$ , while the parameter of end slot category is denoted by  $W_{\text{slot}}^{\text{end}}$ . Because of our unique tagging scheme, the outputs of slot filling is actually divided into two classifiers. Equivalent to Equations (5)–(7), we use a slot type-aware component to calculate  $s_i$ . Formally, the tag of  $i_{\text{th}}$  word  $w_i$  when labeling the start position is formulated as Equation (13).

$$s_i^{\text{slot}} = \text{BiGRU}(S_i), \quad (11)$$

$$y_i^{\text{start}} = \text{softmax}\left(s_i^{\text{slot}} W_{\text{slot}}^{\text{start}} + b_{\text{slot}}^{\text{start}}\right), \quad (12)$$

$$\text{start\_tag}(w_i) = \text{argmax}_k(y_i^{\text{start}} = k), \quad (13)$$

where  $y_i^{\text{start}}$  denotes the slot results of  $i_{\text{th}}$  word when labeling the start position and  $W_{\text{slot}}^{\text{start}}$  and  $b_{\text{slot}}^{\text{start}}$  present the trainable matrix parameters.

Analogously, Equation (15) is formulated to compute the entity span’s end tag.

$$y_i^{\text{end}} = \text{softmax}\left(s_i^{\text{slot}} W_{\text{slot}}^{\text{end}} + b_{\text{slot}}^{\text{end}}\right), \quad (14)$$

$$\text{end\_tag}(w_i) = \text{argmax}_k(y_i^{\text{end}} = k), \quad (15)$$

where  $y_i^{\text{end}}$  denotes the slot results of  $i_{\text{th}}$  word when labeling the end position and  $W_{\text{slot}}^{\text{end}}$  and  $b_{\text{slot}}^{\text{end}}$  are the trainable matrix parameters.

**3.6. Bidirectional Information Transfer Scheme.** Studies have shown that slots and intent are related and can reinforce each other [25, 28, 29]. Recently, intermediate information of the network is used to establish the relationship between slot filling and intent detection. However, the extracted information is insufficient to express the label information. We argue that precise label information from one subtask is more useful to another one. As a result, a two-way information transfer framework is introduced for integrating the two tasks.

During the slot filling process, the probability distribution of intent label  $y_{\text{intent}}$  is combined with the slot representation  $s_i$ . The formulation below is used to replace Equation (11) and showed in Equation (16). Note that the extracted features utilized in our model contain more precise label information, which is collected from network's top layer. After replacement, the slot representation  $s_i^{\text{slot}}$  will include both intent category information and semantic information of slot.

$$s_i^{\text{slot}} = \text{BiGRU}([s_i, y_{\text{intent}}]). \quad (16)$$

Analogously, For the intent detection task, we build a slot to intent iteration component shown as Equations (17)–(19).

$$h_i^{\text{intent}} = \text{BiGRU}\left(\left[h_i, y_i^{\text{start}}, y_i^{\text{end}}\right]\right), \quad (17)$$

$$h^{\text{intent-}} = \text{MaxPooling}\left(\left[h_0^{\text{intent-}}, \dots, h_n^{\text{intent-}}\right]\right), \quad (18)$$

$$y_{\text{intent-}} = \text{softmax}\left(h^{\text{intent-}} W_{\text{intent-}} + b_{\text{intent-}}\right), \quad (19)$$

where  $h^{\text{intent-}}$  is the intent hidden state at each step that contains slot type information  $y_i^{\text{start}}$  and  $y_i^{\text{end}}$ . A MaxPooling layer is adopted to reduce the hidden state dimension. Then the probability distribution of intent  $y_{\text{intent-}}$  is calculated using the softmax function.

**3.7. Joint Training.** A joint training method is adopted to simultaneously update the parameters and learn intent detection and slot filling. The cross-entropy loss for intent detection and slot filling is computed as

$$L_{\text{intent}} = - \sum_{m=1}^M \left( \log \left( y_{\text{intent}} = y_{\text{intent}}^{\widehat{\phantom{y_{\text{intent}}}}} \right) \right), \quad (20)$$

$$L_{\text{slot}} = - \frac{1}{T} \sum_{i=1}^T \sum_{c=1}^C \left( \log \left( y_i^{\text{start}} = y_i^{\widehat{\text{start}}} \right) + \log \left( y_i^{\text{end}} = y_i^{\widehat{\text{end}}} \right) \right), \quad (21)$$

where  $C$  is the number of slot tags,  $M$  represents the number of intent tags,  $T$  is the number of words in a sentence.  $y_{\text{intent}}^{\widehat{\phantom{y_{\text{intent}}}}}$ ,  $y_i^{\widehat{\text{start}}}$ , and  $y_i^{\widehat{\text{end}}}$  are utilized to denote the ground-truth label of intent and slots.

The training objective is to calculate the minimization of the loss function. Finally, the formulation of the loss function is defined as follows:

$$L = \gamma L_{\text{intent}} + (1 - \gamma) L_{\text{slot}}, \quad (22)$$

where  $\gamma$  is applied to adjust the importance of the two tasks.

## 4. Experiments and Analysis

In this part, we will first describe the experimental datasets, which is shown in Table 1. Then we list several baselines which are compared with our model and describe the training details. Lastly, results and analysis of the proposed bidirectional information transfer model will be stated. In addition, the ablation study is also provided to support the effectiveness of our scheme.

**4.1. Dataset.** Experiments are conducted using the SNIPS dataset and the ATIS dataset [31]. Both datasets have the annotation of intent and slot labels. The statistics of ATIS and SNIPS datasets are shown in Table 1.

The ATIS dataset, which includes the recordings of people booking flights, is commonly utilized in NLU. There are 500 validation set, 893 test set, and 4478 training set in the dataset. We also make use of SNIPS dataset, which is taken from SNIPS's digital voice assistant, to assess the effectiveness of our algorithms. The SNIPS dataset contains more evenly distributed samples for intent categorization. This dataset is composed of 700 validation set, 700 test set, and 13,084 training set. We divide the dataset in the same way of [28] did in their experiments.

It should be noted that the "BIO" format annotates the original datasets with the header tags "B" and "I," which is removed in our tagging method. Thus, the slot type of SNIPS dataset is modified from 72 to 40, and the slot type of ATIS dataset is modified from 120 to 84 in the actual experiments.

**4.2. Baselines.** The validity of our framework is verified by comparing it with the latest published models. The following is the list of models:

- (i) Joint Seq. [22]: this is a method based on a RNN-LSTM framework to learn intent and slots simultaneously
- (ii) Attention-based RNN [24]: the model utilizes an attention-based RNN network to obtain utterance context features for forecasting the intent and slots jointly
- (iii) Slot-gated Full Atten [25]: the intent information is applied into the slot filling task by leveraging a slot-gated mechanism
- (iv) Capsule-NLU [28]: the Capsule Neural Network [27] is leveraged to connect intent, words, and slots
- (v) SF-ID, SF-First [29]: this framework develops a SF-ID network to build correlation between the slots and intent



TABLE 1: Dataset statistics.

	ATIS	SNIPS
Vocabulary size	722	11241
Slots	120	72
Intents	21	7
Training set size	4478	13084
Development set size	500	700
Testing set size	893	700

- (vi) Bi-Model [30]: to implement the interaction between slots and intent, Wang et al. [30] introduce a RNN-based model via semantic parsing framework
- (vii) Stack-Propagation [36]: this model develops a joint framework to integrate the intent information with slot filling

Recent works indicate that BERT-based [37] models have also been used successfully to complete the joint tasks [38]. Our method focuses on how to establish interaction between the two tasks instead of the usage of pretrained language model. Thus, our model is not compared with BERT-based models for the sake of fairness.

**4.3. Training Details.** Word embedding is used as a direct input for most language tasks in earlier studies; however, it is unable to solve the out-of-vocabulary (OOV) issue. To address above problem, the embedding layer in our experiments combines the character-level representation and word-level representation. FastText [39] is used to train the word-level embeddings, while the character-level embeddings are generated by random initialization. In the process of training, we fine-tune the above embeddings. To match the dimension of character-level vectors and word-level vectors, we set up the GRU units as 450. The Adam [40] algorithm is leveraged to optimize the loss function: cross entropy, with a batch size of 64. To decrease overfitting, a dropout of 0.15 is applied to the Bi-GRU. If the accuracy of sentence stops growing after 6 consecutive iterations, we will terminate the training process.

#### 4.4. Results and Analysis

**4.4.1. Evaluation Method.** We use the F1 score and accuracy in comparison to some published models to assess the effectiveness of our model. Following previous works, the Recall and Precision are utilized to calculate the F1 score. We score a slot as correct if both the entity type and the entity span are correct. An utterance is considered as correct if both the slots and intent are correct. Table 2 shows the main results of our experiments. The first column lists the name of some published models, while the first line shows the datasets utilized in the experiments. Slot (F1) means the slot filling F1 score, the Intent (Acc) means the accuracy of intent detection, and the Overall (Acc) means the accuracy of an utterance.

**4.4.2. Main Results.** In Table 2, our method is optimal in intent detection and slot filling, achieving an excellent performance on the two datasets: SNIPS and ATIS. On ATIS dataset, our model significantly improves in all 3 aspects comparing with the best model SF-ID, SF-First [29]. The model achieves an absolute gain of 1.2% in terms of sentence accuracy. The experimental results on the SNIPS dataset are also competitive. Our model improves 0.3% in intent detection and 1.3% in slot filling, with better performance than Stack-Propagation [36].

It should be noted that the improved performance of our framework in slot filling and intent detection is mostly due to the efficiency of the proposed bidirectional information transfer method. The findings support the premise that precise label information from one subtask is more helpful to another. As mentioned above, current joint models build the correlation between intent and slots through propagating the information from the intermediate network. However, we argue that more specific label information is contained in the network’s top layer. According to the results, we can also find that both datasets have excellent performance in terms of sentence accuracy. This might be because the relationship of slots and intent improves the sentence-level semantic comprehension and enhance the joint model integrality.

Experiments using the pretrained model BERT [37] achieves competitive results compared with current BERT-based models [38]. Our primary research objective in the future will focus on the fusion of BERT and our framework.

**4.5. Ablation Study.** We perform an ablation experiment to investigate the effects of each module in our framework. The focuses are the two modules: information transfer scheme and type-aware attention mechanism. As Tables 3 and 4 shows, the findings of our entire framework are listed in the second line, while 3 more tests are conducted from line 3 to line 5. The first experiment removes the information transfer scheme and ignores the relationship between intent and slots. The second experiment removes the type-aware method and maintains the information transfer scheme. The third experiment removes both of the two modules mentioned above. Thus, only the pointer-based annotation method is applied.

According to the ablation test of the SNIPS dataset, the slot filling achieves 0.75% improvement in F1 score with the information transfer module (refer to the third row of Table 3. As shown in the third column of Table 3, the intent detection accuracy also increases from 97.85% to 98.29%. The aforementioned results validate the effectiveness of our information transfer module. In addition, the intent detection accuracy achieves a gain of 0.29% and the slot filling F1 score achieves a gain of 0.89% with our type-aware method. Therefore, the type-aware module performs well on both intent detection and slot filling.

In the ablation test of the ATIS dataset, the slot filling F1 score achieves a 1.89% improvement, and the intent detection accuracy increases from 94.84% to 96.97% when utilizing the information transfer module (refer to the forth row of Table 4). In addition, the F1 score also achieves a 2.05% improvement in slot filling, while the intent detection accuracy achieves a 1.57% improvement when utilizing our type-

TABLE 2: Comparison with published results of joint models on the ATIS and SNIPS datasets.

Model	Overall (Acc)	Intent (Acc)	Slot (F1)	Overall (Acc)	Intent (Acc)	Slot (F1)
Joint Seq. [22]	73.2	96.9	87.3	80.7	92.6	94.3
Attention-based RNN [24]	74.1	96.7	87.8	78.9	91.1	94.2
Slot-gated [25]	75.5	97.0	88.8	82.2	93.6	94.8
Capsule-NLU [28]	80.9	97.3	91.8	83.4	95.0	95.2
SF-ID, SF-First [29]	80.6	97.4	91.4	86.8	97.8	95.8
Bi-Model [30]	83.8	97.2	93.5	85.7	96.4	95.5
Stack-Propagation [36]	86.9	98.0	94.2	86.5	96.9	95.9
Bi-transfer (our model)	85.9	98.3	95.5	88.0	98.7	96.0

TABLE 3: The ablation test on SNIPS dataset.

	Overall (ACC)	Intent (Acc)	Slot (F1)
Bi-transfer(our model)	85.86	98.29	95.50
Bi-transfer (no information transfer)	83.86	97.85	94.75
Bi-transfer (no type-aware attention)	82.85	98.00	94.61
Bi-transfer (no both component)	81.71	97.57	94.25

TABLE 4: The ablation test on ATIS dataset.

	Overall (Acc)	Intent (Acc)	Slot (F1)
Bi-transfer(our model)	88.02	98.66	96.00
Bi-transfer (no information transfer)	87.63	96.41	95.88
Bi-transfer (no type-aware attention)	87.12	96.97	95.72
Bi-transfer (no both component)	84.99	94.84	93.83

aware module (refer to the third row of Table 4). The aforementioned experimental results validate the efficiency of the information transfer module and type-aware module. Lastly, we find that the improvements within the ATIS dataset are more absolute than in the SNIPS dataset. This may be because our method is more beneficial to the dataset having a large variety of categories.

Furthermore, we find that the test only utilizing the type-aware module has better performance than the test only utilizing the information transfer module in slot filling. On the contrary, the test only using the information transfer module performs better than the test only using the type-aware module in intent detection. To sum up, our proposed type-aware module works better for slot filling task, which may credit to that this module is beneficial to learning type information.

**4.6. Error Analysis.** We further analyze some error cases of the experimental results. It is observed that most misclassification cases from the SNIPS dataset include multiple intents, while our framework is designed to connect the slots with single-label intent. In our proposed bidirectional transfer scheme, each token in slot filling is provided with the same multiple intents information, which may import irrelevant

noise for some slots. This may influence the global integrality of the joint model, which leads to worse results than the current published models in the sentence accuracy of the SNIPS dataset. In practice, the intent detection module is supposed to provide fine-grained intent information to the token-level slots so that the slot can be guided by its associated intent information. We will concentrate on how to leverage multiple intents to guide corresponding slot predictions in the future.

## 5. Related Work

The Internet of Things uses devices and sensors to connect various objects to a network, which establishes the correlation between people and the world [1, 2]. Some studies suggest that natural language will become the primary interactive mode between people and terminals in IoT. Human-computer conversation technologies such as natural language understanding, knowledge graph, and semantic web can be used to link people and the physical environments. Xue et al. [3–5] introduce some algorithms to match sensor ontologies for the purpose of applying semantic interoperability among intelligent sensor applications. In addition, natural language understanding technologies are also widely utilized in IoT.

In previous methods, slot filling and intent detection are treated as independent operations in pipeline manners. The task of intent detection is generally regarded as a classification problem, which relies on the methods of support vector machines (SVMs) [7] and deep learning frameworks [8–11]. Recently, a transformer model and universal sentence encoder-based deep averaging network are utilized in intent detection task [9]. Different from intent detection, slot filling is formulated as a task of sequence labeling. Previous work on slot filling is relied on CRF [12] and MEMMs [14]. Currently, neural network models are combined with CRF to address the slot filling issues. Gong et al. [19] perform slot filling task by a deep cascade multitask learning scheme based on BiLSTM-CRF. It is simple to conduct these two tasks separately, but it is difficult to establish the relationship between the slots and intent. Besides, pipeline approaches may result in error propagation issue.

The error propagation issue of pipeline approaches is solved by training the intent detection task and slot filling task jointly [25]. Previous methods share the input embeddings

and use a common loss function for joint models [21, 23]. Xu and Sarikaya [20] introduce a CNN-based framework to collect features from slot filling task, which is utilized to enhance the intent detection. A RNN-LSTM architecture [22] is introduced to enable the prediction of intent and slots optimized in a joint model based on bidirectional RNN with LSTM cells. In addition, Liu and Lane [24] develop a RNN-based framework using an attention mechanism to predict the intent and slots jointly. But the approaches mentioned above continue to ignore the correlation of intent and slots.

To address the aforementioned limitation, Goo et al. [25] suggest to use a slot-gated algorithm to connect intent and slots. This method creates an attention scheme to leverage intent information in slot filling. In addition, a stack-propagation model [36] is proposed to merge intent information with the prediction of slots. However, the flow of information in both methods is unidirectional. The Capsule Neural Network [27] is utilized to create correlation between intent, words, and slots [28]. The model adopts a routing-by-agreement mechanism to achieve information transmission. A SF-ID network [29] is suggested to establish bidirectional interaction between intent and slots. Hui et al. [41] also propose a continuous learning model for considering semantic information with various features. Although the methods mentioned above build bidirectional relationship between intent and slots, the features in the process of propagation is still collected from the input representation.

We believe that precise type information can promote both subtasks in joint models. Thus, we develop a two-way information transfer framework with type-aware information enhancement and pointer-based tagging method. We first propose a unique type-aware module to reinforce the discriminative information. Then, we introduce a bidirectional information transmission module to establish complete correlation between slots and intent, which collects precise type information from network's top layer. To accelerate the process of training, a point-based tagging method is leveraged in our model.

## 6. Conclusion

We introduce a joint model for the prediction of intent and slots with type-aware attention scheme. A bidirectional information transmission module and a type-aware attention module are proposed to create complete correlation between intent and slots, which utilizes the information extracted from network's top layer. Then a point-based tagging scheme is introduced to make the model be more time-saving and efficient. The results of experiments confirm the suggestion that building complete relationship between intent and slots is helpful to promote the performance of the subtasks. In summary, the proposed model outperforms other published models on the SNIPS dataset and ATIS dataset.

## 7. Future Work

Future research will focus on the fusion method of our framework with language model BERT. It is observed that our method shows poor performance on a dataset that con-

tains multiple intents; thus, we will try to establish a more fine-grained correlation between multiple intents and slot filling. Furthermore, future research should investigate how to combine natural language understanding technologies with devices and sensors.

## Data Availability

The article contains all datasets utilized to support the study's conclusions.

## Disclosure

This research was previously presented at the 17th International Conference on Computational Intelligence and Security [42]. The corresponding author is Lu Rao.

## Conflicts of Interest

The authors declare that there are no conflicts of interest.

## Acknowledgments

This work is supported by the Scientific Research Projects of Leshan Normal University (Grant No. XJR17001, ZZ201822, and LZD005), the Key Projects of Sichuan Provincial Education Department of China (Grant No. 18ZA0239), the National Key R&D Program of China (Grant No. 2017YFA0700800), and the Projects of Sichuan Tourism Development Research Center (Grant No. LY21-21).

## References

- [1] L. Tan and N. Wang, "Future internet: the Internet of Things," in *2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)*, Chengdu, China, August 2010.
- [2] K. O. M. Salih, T. A. Rashid, D. Radovanovic, and N. Bacanin, "A comprehensive survey on the Internet of Things with the industrial marketplace," 2022, <https://arxiv.org/abs/2202.03142>.
- [3] X. Xue and C. Jiang, "Matching sensor ontologies with multi-context similarity measure and parallel compact differential evolution algorithm," *IEEE Sensors Journal*, vol. 21, no. 21, pp. 24570–24578, 2021.
- [4] X. Xue, X. Wu, C. Jiang, G. Mao, and H. Zhu, "Integrating sensor ontologies with global and local alignment extractions," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6625184, 10 pages, 2021.
- [5] X. Xue and J. Chen, "Using compact evolutionary tabu search algorithm for matching sensor ontologies," *Swarm and Evolutionary Computation*, vol. 48, pp. 25–30, 2019.
- [6] T. Gokhan and D. M. Renato, *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*, Wiley, 2011.
- [7] P. Haffner, G. Tur, and J. H. Wright, "Optimizing SVMs for complex call classification," in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03)*, pp. 632–635, Hong Kong, China, 2003.



- [8] C. Xia, C. Zhang, X. Yan, Y. Chang, and P. S. Yu, "Zero-shot user intent detection via capsule neural networks," 2018, <https://arxiv.org/abs/1809.00385>.
- [9] S. Yolchuyeva, G. Németh, and B. Gyires-Tóth, "Self-attention networks for intent detection," in *Proceedings of Recent Advances in Natural Language Processing*, pp. 1373–1379, Varna, Bulgaria, October 2019.
- [10] E. Okur, S. H. Kumar, S. Sahay, A. A. Esme, and L. Nachman, "Natural language interactions in autonomous vehicles: intent detection and slot filling from passenger utterances," 2019, <https://arxiv.org/abs/1904.10500>.
- [11] Y. Tian and P. J. Gorinski, "Improving end-to-end speech-to-intent classification with reptile," 2020, <https://arxiv.org/abs/2008.01994>.
- [12] L. John, M. C. Andrew, and P. Fernando, "Conditional random fields: probabilistic models for segmenting and labeling sequence data," in *Proceedings of 18th International Conference on Machine Learning*, pp. 282–289, Williamstown, MA, USA, 2001.
- [13] C. Raymond and G. Riccardi, "Generative and discriminative algorithms for spoken language understanding," in *Interspeech 2007*, pp. 1605–1608, Antwerp, Belgium, August 2007.
- [14] A. McCallum, D. Freitag, and F. Pereira, "Maximum entropy Markov models for information extraction and segmentation," in *Proceedings of 17th International Conference on Machine Learning*, Stanford, CA, USA, 2000.
- [15] K. Yao, B. Peng, Y. Zhang, D. Yu, G. Zweig, and Y. Shi, "Spoken language understanding using long short-term memory neural networks," in *2014 IEEE Spoken Language Technology Workshop (SLT)*, pp. 189–194, South Lake Tahoe, NV, USA, December 2014.
- [16] G. Mesnil, Y. Dauphin, K. Yao et al., "Using recurrent neural networks for slot filling in spoken language understanding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 530–539, 2015.
- [17] G. Kurata, B. Xiang, B. Zhou, and M. Yu, "Leveraging sentence-level information with encoder LSTM for semantic slot filling," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2077–2083, Austin, Texas, 2016.
- [18] B. Liu and I. Lane, "Multi-domain adversarial learning for slot filling in spoken language understanding," 2017, <https://arxiv.org/abs/1711.11310>.
- [19] Y. Gong, X. Luo, Y. Zhu et al., "Deep cascade multi-task learning for slot filling in online shopping assistant," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 6465–6472, 2019.
- [20] P. Xu and R. Sarikaya, "Convolutional neural network based triangular CRF for joint intent detection and slot filling," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pp. 78–83, Olomouc, Czech Republic, December 2013.
- [21] D. Guo, G. Tur, W.-t. Yih, and G. Zweig, "Joint semantic utterance classification and slot filling with recursive neural networks," in *2014 IEEE Spoken Language Technology Workshop (SLT)*, pp. 554–559, South Lake Tahoe, NV, USA, December 2014.
- [22] D. Hakkani-Tür, G. Tur, A. Celikyilmaz et al., "Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM," in *Proceedings of the 17th Annual Meeting International Speech Communication Association (INTERSPEECH 2016)*, pp. 715–719, San Francisco, CA, USA, 2016.
- [23] Y.-N. Chen, D. Hakkani-Tür, G. Tur, J. Gao, and L. Deng, "End-to-end memory networks with knowledge carryover for multi-turn spoken language understanding," in *Proceedings of the 17th Annual Meeting of the International Speech Communication Association (INTERSPEECH 2016)*, San Francisco, CA, USA, 2016.
- [24] B. Liu and I. Lane, "Attention-based recurrent neural network models for joint intent detection and slot filling," 2016, <https://arxiv.org/abs/1609.01454>.
- [25] C.-W. Goo, G. Gao, Y.-K. Hsu et al., "Slot-gated modeling for joint slot filling and intent prediction," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 753–757, New Orleans, Louisiana, 2018.
- [26] G. E. Hinton, A. Krizhevsky, and S. D. Wang, "Transforming auto-encoders," in *Proceedings of the 21st International Conference on Artificial Neural Networks*, pp. 44–51, Espoo, Finland, 2011.
- [27] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Proceedings of Conference and Workshop on Neural Information Processing Systems*, pp. 3856–3866, Long Beach, CA, USA, 2017.
- [28] C. Zhang, Y. Li, N. Du, W. Fan, and P. Yu, "Joint slot filling and intent detection via capsule neural networks," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5259–5267, Florence, Italy, 2019.
- [29] E. Haihong, P. Niu, Z. Chen, and M. Song, "A novel bi-directional interrelated model for joint intent detection and slot filling," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5467–5471, Florence, Italy, 2019.
- [30] Y. Wang, Y. Shen, and H. Jin, "A bi-model based RNN semantic frame parsing model for intent detection and slot filling," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 309–314, New Orleans, Louisiana, 2018.
- [31] G. Tur, D. Hakkani-Tur, and L. Heck, "What is left to be understood in ATIS?," in *2010 IEEE Spoken Language Technology Workshop*, pp. 19–24, Berkeley, CA, USA, December 2010.
- [32] Z. Wei, J. Su, Y. Wang, Y. Tian, and Y. Chang, "A novel cascade binary tagging framework for relational triple extraction," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8357–8366, Seattle, WA, USA, 2020.
- [33] V. Oriol, F. Meire, and J. Navdeep, "Pointer networks," in *Proceedings of the Conference and Workshop on Neural Information Processing Systems*, pp. 2692–2700, Montreal, Quebec, Canada, 2015.
- [34] K. Cho, B. van Merriënboer, C. Gulcehre et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, <https://arxiv.org/abs/1406.1078>.
- [35] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, <https://arxiv.org/abs/1409.0473>.
- [36] L. Qin, W. Che, Y. Li, H. Wen, and T. Liu, "A stack-propagation framework with token-level intent detection for spoken language understanding," 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. 2078–2087, Hong Kong, China, 2019.
- [37] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” 2018, <https://arxiv.org/abs/1810.04805>.
- [38] Q. Chen, Z. Zhuo, and W. Wang, “BERT For Joint Intent Classification and Slot Filling,” 2019, <https://arxiv.org/abs/1902.10909>.
- [39] M. Tomas, G. Edouard, B. Piotr, P. Christian, and J. Armand, “Advances in pre-training distributed word representations,” in *Proceedings of LREC*, pp. 8357–8366, Miyazaki, Japan, 2018.
- [40] D. P. Kingma and J. Ba, “Adam: a method for stochastic optimization,” 2014, <https://arxiv.org/abs/1412.6980>.
- [41] Y. Hui, J. Wang, N. Cheng, F. Yu, T. Wu, and J. Xiao, “Joint intent detection and slot filling based on continual learning model,” in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, ON, Canada, June 2021.
- [42] R. Sun, L. Rao, and X. Zhou, “Bidirectional information transfer scheme for joint intent detection and slot filling,” in *2021 17th International Conference on Computational Intelligence and Security (CIS)*, Chengdu, China, November 2021.

## Research Article

# Multiobjective Optimization Model and Algorithm Based on Differential Brain Storm for Service Path Constructing

Xiaoqiang Jia , Li Ge, Yunfei Li, Jun Liu, and Biying Zhou

School of Computer and Technology, Weinan Normal University, Weinan, Shaanxi 714000, China

Correspondence should be addressed to Xiaoqiang Jia; xqjia@wnu.edu.cn

Received 14 March 2022; Revised 12 April 2022; Accepted 5 May 2022; Published 10 June 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Xiaoqiang Jia 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.

Network function virtualization (NFV) can provide the resource according to the request and can improve the flexibility of the network. It has become the key technology of the next-generation communication. Resource scheduling for virtual network function service chain (VNF-SC) mapping is the key issue of the NFV. Aiming at previous research primarily focused on constructing service paths with a single objective, for example, latency minimization, cost minimization, or load balance, which ignored the overall performance of constructed service paths, a multiobjective model, which minimizes benefit, expense, time delay, and load balance, to solve the service path constructing problem, is established. In this model, VNF-SCs are divided into two classes, i.e., part of the required VNFs in each VNF-SC is dependent, and others are independent. To solve this multiobjective model, an algorithm based on discrete difference brain storm (MO2DBS) in the framework of MOEA/D was proposed. In the new algorithm, a two dimensional integer coding is designed. In addition, a difference mutation was used to replace the original Gaussian mutation to adjust the mutation step adaptively. Simulation experiments show that the proposed algorithms can obtain higher benefit, load balance, lower expenses, and time delay than the compared algorithms.

## 1. Introduction

Network function virtualization (NFV) [1–3] proposes the concept of “softwarization,” which decouples network functions from customized hardware devices and uses virtualization technology to build software units. Through the deployment and combination of VNF, diversified service customization can be realized. Compared with the traditional middlebox approach, NFV enables flexible development, deployment, and migration of network functions, thus effectively reducing the cost of equipment investment. Combined with SDN (software defined network) [4, 5] technology, the management program on the upper layer of the controller is adopted to control the network, which further improves the manageability and reduces the operation and maintenance cost [6, 7].

In NFV architecture, in order to meet diversified application requirements and realize on-demand service customization, the controller needs to implement the service routing

algorithm to map each service in the demand to the node that can carry the service in order to build an end-to-end data path, so that the network flow is processed by the required service in turn [8]. This kind of directly connected sequence composed of nodes and links is called service path, which traverses the whole network to meet specific function and performance requirements. This problem is called service path construction problem, and the ordered service sequence contained in application request is called service chain. In the process of constructing the service path, the following problems need to be solved [9]: (1) the execution of the service requires computing and storage resources, and the transmission of data requires bandwidth resources. Under the condition of limited resource capacity, how to allocate resources efficiently to map more service chains. (2) Service providers need to pay for rented resources to build service paths according to the users’ demands and provide customized services to users through the service paths to obtain benefits. Therefore, costs and benefits need to be included in the evaluation system, and how to effectively rent

resources should be considered to reduce costs and improve benefits. (3) A service instance can be deployed on any network node. The location of the service bearer node affects the transmission delay and link resource usage of the service path, the performance affects the processing delay of the service instance, and the available resources affect the capacity of carrying the service. Therefore, it is necessary to solve the problem of how to choose among many candidate service instances and construct the optimal service path [10, 11].

Since service path construction is an NP-hard problem [12–14], the research work has been mainly focused on designing heuristic algorithms to obtain approximate optimal solutions. Literature [15] proposed a layered graph to solve the problem of service execution sequence. The method of constructing service step search graph, which ensured the service execution sequence and also took into account the problem of excessive use of resources, is proposed [16]. Literature [17] used the reciprocal of the available resource capacity to reset the weights of hierarchical graph edges and proposed a load balancing algorithm. Literature [18] proposed a heuristic algorithm based on dynamic programming to achieve the balance between cost and performance through the dynamic orchestration of VNF. Literature [19] proposed a mapping algorithm based on service chain decomposition, so that VNF based on the same implementation technology can be preferentially interconnected and mapped to the same server, thus effectively reducing the overall cost. Literature [20] maps topology fragments to candidate servers in data centers by requesting topology segmentation and uses virtual gateways to construct service chain diagrams to realize traffic aggregation, so as to solve cross-domain service chain mapping problems. Literature [21] studied the mechanism of network function division, description, and combination under SDN architecture and proposed a node-first algorithm. This literature builds the security service path through the flexible combination of VNF, so as to provide customized security services.

These studies have been explored from different perspectives, but they may be aimed at minimizing overhead, minimizing end-to-end delay, or load balancing. Few studies have been able to comprehensively consider the construction of a service path that not only optimizes delay and cost but also balances load. Aiming at previous research primarily focused on constructing service paths with a single objective, for example, latency minimization, cost minimization, or load balance, which ignored the overall performance of constructed service paths, a multiobjective model, which minimizes benefit, expense, time delay, and load balance, to solve the service path constructing problem, is established. In this model, VNF-SCs are divided into two classes, i.e., part of the required VNFs in each VNF-SC is dependent, and others are independent. To solve this multiobjective model, an algorithm based on discrete difference brain storm (MO2DBS) in the framework of MOEA/D was proposed. In the new algorithm, a two dimensional integer coding is designed. In addition, a difference mutation was used to replace the original Gaussian mutation to adjust the mutation step adaptively. Simulation experiments show

that the proposed algorithms can obtain higher benefit, load balance, lower expenses, and time delay than the compared algorithms.

## 2. Multiobjective Service Path Building Model

### 2.1. Network Model

**2.1.1. Physical Network.** The topology of the physical network can be described as a weighted undirected graph, denoted as  $G^S = (N^S, L^S, A_N^S, A_L^S)$ , where  $N^S$  and  $L^S$  denote the set of nodes and links in the network, respectively.  $A_N^S$  denotes the resource property of node  $n^S$  in  $N^S$ , including the available CPU capacity  $C(n^S)$  of the node and the service set  $S(n^S) = S_k$  that can be provided by the node  $n^S$ . The processing time of the service  $s_k$  in  $N^S$  denoted as  $d_{s_k}(n^S)$ , per time unit CPU capacity, is  $c^S$ .  $A_N^S$  denotes the resource property of the link, including the link available bandwidth  $B(l^S)$ , transmission delay  $D(l^S)$ , and unit bandwidth resource cost  $b^S$ . The set of all acyclic paths in the physical network is represented as  $\Pi^S$ . The set of acyclic paths between any two nodes  $n_i^S$  and  $n_j^S \in N^S$  is marked as  $\Pi^S(n_i^S, n_j^S)$ .  $p^S$  represents a acyclic path in the physical network, and  $H(p^S)$  denotes the length of path  $p^S$ , i.e., hop number. As shown in Figure 1, S1~S4 in physical network nodes represent the services that nodes can instantiate, the numbers inside nodes represent the current available computing capacity of nodes, and the numbers on links represent available bandwidth and transmission delay.

**2.1.2. Service Request.** A service request represents a user's specific functional and resource requirements in the form of a logical service path, whose topology can be described as a weighted directed graph labeled as  $G^R = (N^R, L^R, R_N^R, R_L^R)$ . The directivity of an edge represents a constraint on the execution order of a service instance.  $N^R = \{n_0^R, n_1^R, \dots, n_c^R, n_{c+1}^R\}$  represents a collection of logical path nodes.  $n_0^R$  and  $n_{c+1}^R$  denote the source node and the destination node, respectively.  $c$  represents the number of services required.  $L^R = \{<n_0^R, n_1^R>, <n_1^R, n_c^R>, \dots, <n_c^R, n_{c+1}^R>\}$  denotes a set of logical links.  $R_N^R$  denotes the collection of requirement attributes of node  $n_i^R$ , service needs  $S(n_i^R)$ , computing capacity requirements  $\mu(S(n_i^R))$ , the rental unit computing capacity required to pay fees  $c^R$ , etc. Similarly,  $R_L^R$  denotes the collection of requirement attributes of link  $l^R$ ; it consists of the bandwidth requirement ( $l^R$ ) and the charge for renting unit bandwidth  $b^R$ . Each service request can be represented by the triplet  $SR(G^R, t_a, t_d)$ , where  $t_a$  indicates the arrival time of the request and  $t_d$  indicates the duration of the request. As shown in Figure 1, for service request SR1, A and G represent the source node and destination node, respectively, S1~S3 within logical nodes, A, B, and C represent the required service, the numbers inside the nodes represent the computing power demand, A~G represent the underlying network nodes, and the numbers on the logical link represent the bandwidth required for data transmission. In addition, the dotted arrows in Figure 1 represent the mapping of services. Dashed lines between A and G indicate that there is no mapping between nodes.

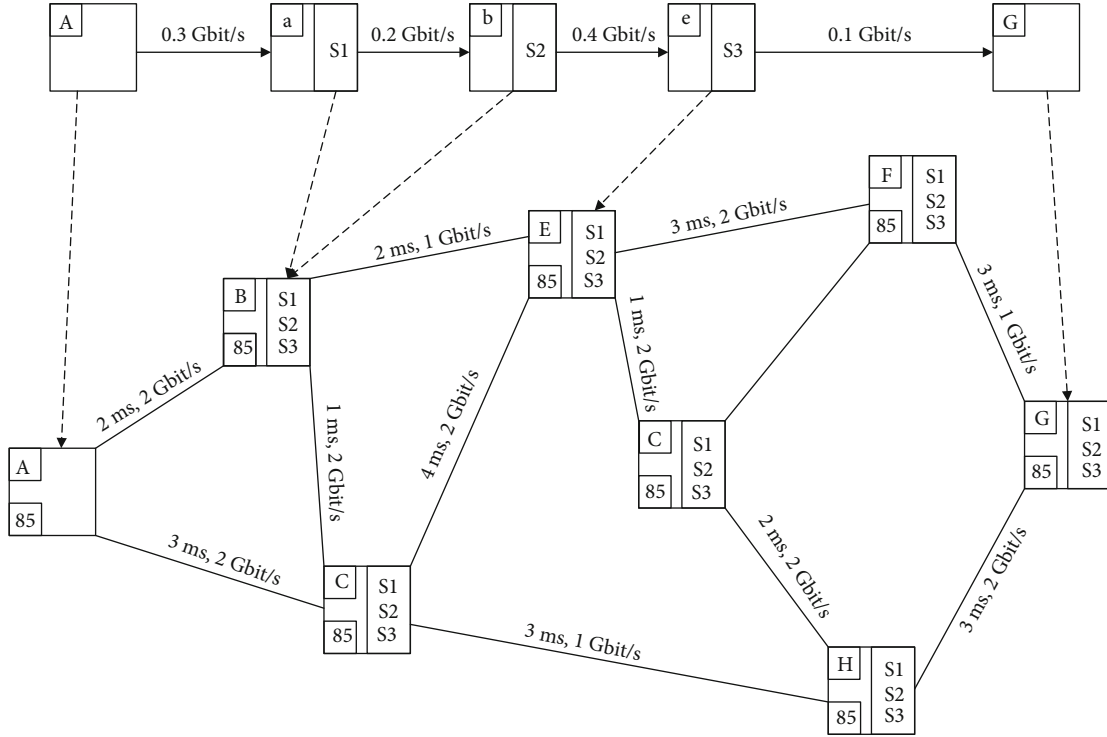


FIGURE 1: Examples of physical network and service paths.

2.2. *Formula Description.*  $SC = S(n_1^R), S(n_2^R), \dots, S(n_c^R)$  denotes the service chain required in the service request; each remaining logical node represents a required service except the source node and the destination node. Service path to complete the build process can be described as after receiving the user's service request, the service is mapped to the controller; in turn, SC can provide the service and computing ability to meet the demand of the underlying nodes. At the same time, in the service load between the nodes to establish and meet the corresponding optimal underlying path bandwidth demand, it ensures that the data flows on the path pass through the service instances agreed on the SC and finally build an end-to-end service path that meets the requirements of specific functions, performance, and latency. If the path does not exist, the request is rejected.

2.2.1. *Objective Function.* A single service path built for a service request needs to achieve the following goals:

(1) *Profit.* Similar to literature [5, 13], the revenue obtained by the service provider for successfully constructing a service path for a service request is equal to the sum of the fees paid by the users for renting computing resources and bandwidth resources, as defined below:

$$R(SP) = \sum_{n^r \in N^R} \mu(S(n^R))c^R + \sum_{l^r \in L^R} \mu(l^R)b^R. \quad (1)$$

The cost paid by a service provider to successfully construct a service path for a user is equal to the sum of the

computing resources of the underlying node and the bandwidth resources of the underlying link, as defined below:

$$C(SP) = \sum_{n^r \in N^R} \mu(S(n^R))c^R + \sum_{l^r \in L^R} \sum_{p^s \in M_L(l^R)} \sum_{f \in P^S} \mu(l^R)b^R. \quad (2)$$

The profit of the service request is the revenue minus to the cost; thus, the profit can be defined as

$$P(SP) = \sum_{l^r \in L^R} \mu(l^R)b^R - \sum_{l^r \in L^R} \sum_{p^s \in M_L(l^R)} \sum_{f \in P^S} \mu(l^R)b^R. \quad (3)$$

Since  $P(SP) < R(SP)$ , thus, we can normalise the objective as follows:

$$\min f_1 = \min \left\{ \frac{P(SP)}{R(SP)} \right\}. \quad (4)$$

According to the definition, we can see that  $0 \leq f_1 \leq 1$ .

(2) *Transmission Delay.* The end-to-end delay of the service path represents the time taken by the data flow from the source node to the destination node and is composed of the execution delay of the service instance on the service path and the transmission delay of the communication link, as defined below:



$$D(SP) = \sum_{n^R \in N^R} \sum_{n^S \in M_S(S(n^R))} d_{S(n^R)} - \sum_{l^R \in L^R} \sum_{P^S \in M_L(l^R), F \in P^S} d(l^S). \quad (5)$$

Another variable is defined:

$$D'(SP) = \sum_{n^R \in N^R} \sum_{n^S} d_{S(n^R)} - \sum_{l^R \in L^R} \sum_{F \in P^S} d(l^S). \quad (6)$$

Obviously, we have  $D(SP) < D'(SP)$ ; thus,  $0 \leq f_2 = D(SP)/D'(SP) \leq 1$ . The second objective is defined as

$$\min f_2 = \min \left\{ \frac{D(SP)}{D'(SP)} \right\}. \quad (7)$$

(3) *Load Degree*. When constructing the service path, besides the functional requirements of the service, the load of the service bearing node and its adjacent links should be considered to map the service and logical links to the resource-rich underlying nodes and links as far as possible. Load degree (LD) measures the resource usage and load of a service path. The definition of the load intensity of the bottom node  $n_{ik}^R$  in the service path is as follows:

$$LD_{n_{ik}^R} = \sum_{S(n^R) \text{ and } n_{ik}^R \in M_{ik}} \frac{\mu(S(n^R))}{C(n_{ik}^R)}, \forall n_{ik}^R \in N^S P. \quad (8)$$

The load intensity of a node takes into account the available computing capacity of the node and the computing capacity demand of the service. According to Formula (3), its value ranges from 0 to 1. The smaller the value is, the more likely it is to map services to this node, and the more beneficial it is to load balancing of the underlying node. Based on the idea of node load intensity, the load intensity of link  $l_{ij}$  in the service path is defined as follows:

$$LD_{l_{ij}^R} = \sum_{P^S \in M_L(l_{ij}^R), l_{ij}^R \in P^S} \frac{\mu(l_{ij}^R)}{B(l_{ij}^R)}, \forall l_{ij}^R \in L^S P. \quad (9)$$

The load intensity of the service path SP is defined as

$$LD(SP) = w_N \sum_{n_{ik}^S \in N^{SP}} LD_{n_{ik}^S} + w_L \sum_{l_{ik}^S \in L^{SP}} LD_{l_{ik}^S}, \quad (10)$$

where  $w_N$  and  $w_L$  are the two weight parameters used to adjust the load intensity of nodes and links, and  $0 \leq w_N, w_L \leq 1$  and  $w_N + w_L = 1$ . Similarly, another variable is defined:

$$LD'(SP) = \sum_{n_{ik}^S \in N^{SP}} LD_{n_{ik}^S} + \sum_{l_{ik}^S \in L^{SP}} LD_{l_{ik}^S}. \quad (11)$$

Obviously, we have  $LD(SP) < LD'(SP)$ ; thus,  $0 \leq f_3 = LD(SP)/LD'(SP) \leq 1$ . The third objective is defined as

$$\min f_3 = \min \left\{ \frac{LD(SP)}{LD'(SP)} \right\}. \quad (12)$$

### 2.2.2. Constraints

- (1) For computing resource constraints, ensure that the resources available on the underlying nodes can meet the needs of executing the service instance:

$$\sum_{u \in N^R} x_i^{S(u)} \mu(S(u)) \leq C(i), \forall i \in N^{SP}. \quad (13)$$

The  $x_i^{S(u)}$  value is 1 if the service  $S(u)$  required by the logical node  $u$  is mapped to the underlying node  $i$ , and otherwise is 0

- (2) Bandwidth resource constraints ensure that the bandwidth available on the underlying link is sufficient to bear the logical link mapped to

$$\sum_{l_{u,v} \in L^R} y_{ij}^{uv} \mu(l_{u,v}) \leq B(l_{ij}), \forall l_{ij} \in L^{SP}. \quad (14)$$

If the underlying link  $l_{ij}$  carries the logical link  $l_{u,v}$ , the  $y_{ij}^{uv}$  value is 1. Otherwise, the value is 0

- (3) The service-to-underlying node mapping constraint ensures that services in the same service request can be mapped to only one underlying node.

$$\sum_{i \in N^S} x_i^{S(u)} = 1, \forall u \in N^R \quad (15)$$

- (4) The underlying node hosts constraints on services, and one underlying node can host multiple services for the same service request.

$$\sum_{u \in N^R} x_i^{S(u)} \leq \lambda, \forall u \in N^{SP} \quad (16)$$

- (5) Connectivity constraints on logical links to mapped underlying network paths:

$$\sum_{L_{i,j} \in L^{SP}} y_{ij}^{uv} - \sum_{L_{j,i} \in L^{SP}} y_{ji}^{uv} = x_i^{S(u)} - x_i^{S(v)}, \forall i \in N^{SP} \quad (17)$$

- (6) End-to-end latency constraint ensures that the end-to-end latency of service paths meets the service level agreement (SLA) requirements.

$$\sum_{u \in N^R} \sum_{i \in N^{SP}} x_i^{S(u)} d_{S(u)}(i) + \sum_{l_{u,v} \in L^R} \sum_{L_{i,j} \in L^{SP}} y_{ij} d_{l_{i,j}} \leq D_{\max} \quad (18)$$

- (7) Variable's constraints:

$$\begin{aligned} \forall i \in N^S, \\ u \in N^R, \\ x_i^S(u) \in \{0, 1\} \\ \forall L_{i,j} \in L^S, \\ L_{u,v} \in L^R, \\ y_{ij}^{uv} \in \{0, 1\} \end{aligned} \quad (19)$$

### 3. Multiobjective Discrete Difference Brain Storm (MO2DBS)

In recent years, BSO, as a new swarm intelligence optimization algorithm, has attracted extensive attention of many scholars. BSO and its improved version have been successfully applied in some fields, for example, satellite formation optimization [22], BRUSHless DC motor optimization [23], image processing [24], and route planning [25]. On the basis of BSO, scholars at home and abroad derived different brainstorming optimization algorithms based on its clustering, selection, mutation, and other methods, which improved the performance of the algorithm [26]. BSO has the following four methods to select individuals to be mutated: (1) select a class according to roulette probability, and select the class center of the class as individuals to be mutated. (2) A class was selected according to the roulette probability, and a random individual in the class was selected as the individual to be mutated. (3) Two classes were randomly selected, and the class centers of the two classes were fused to become individuals to be mutated. (4) Two classes were randomly selected, and one individual was randomly selected from each of the two classes and fused into the individual to be mutated. Select the fusion process in the operation by pressing the following formula:

$$y = r \cdot x_1 + (1 - r) \cdot x_2, \quad (20)$$

where  $y$  is the individual to be mutated after the fusion of two individuals,  $x_1$  and  $x_2$  are the two individuals receiving

fusion, and  $r$  is a random number from 0 to 1 that adjusts the weight of two individuals.

The mutation operation will add disturbance quantity to the individual to be mutated, which is called the mutation step size in this paper. Gaussian mutation is carried out as follows:

$$y^d = x^d + \xi \cdot N(\mu, \sigma), \quad (21)$$

where  $y^d$  is the  $d$ -dimension of the new individual,  $x^d$  is the  $d$ -dimension of the individual to be mutated, and  $N(\mu, \sigma)$  is a Gaussian random number whose mean value is  $\mu$  and variance is  $\sigma^2$ . The calculation formula of coefficient of variation is as follows:

$$\xi = \log \text{sig} \left( \frac{0.5e_{\max} - e}{k} \right) \times \text{rand}(), \quad (22)$$

where  $e_{\max}$  is the maximum number of iterations,  $e$  is the current iteration number,  $k$  is the coefficient regulating the slope of the S-type transfer function  $\log \text{sig}()$ , and  $\text{rand}()$  is a random number between 0 and 1.

At the beginning of human brainstorming, everyone's ideas are very different. When creating new ideas, take into account the differences between the existing ideas. Therefore, difference variation is used to determine the variation step. The mutation operation based on differential variation is as follows:

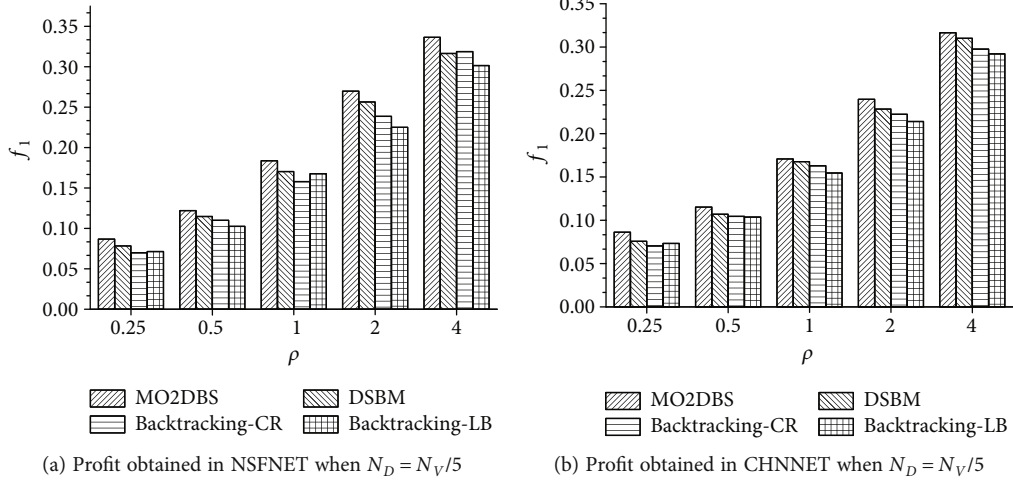
$$y = \begin{cases} R \times (H_d - L_d) + L_d, & \text{rand}() < p_r, \\ x + R \times (x_a - x_b), & \text{else,} \end{cases} \quad (23)$$

where  $L_d$  and  $H_d$  are the boundary values of the search space,  $p_r$  means that there is a certain probability to get a random new individual, and  $x_a$  and  $x_b$  are any two different individuals in the population.

Differential variation has two advantages over Gaussian variation. On the one hand, the operation of Gaussian variation includes  $\text{logsig}()$  function, Gaussian distribution function, random function, and four mixed operations, while difference variation only has random function and four mixed operations, which greatly reduces the amount of computation. On the other hand, the step size of differential variation is based on the contemporary population and adaptively adjusts according to the dispersion degree of individuals in the population: there is a larger step size when the population is dispersed, and a smaller step size when the population is concentrated. The mutation step size is confirmed in real time according to the population feedback, and the algorithm can capture the search feature better.

## 4. Experiment and Analysis

**4.1. Experimental Setting.** The underlying network topology is randomly generated by GT-ITM tool, which contains 50 nodes and about 130 links. The computing capacity of the bottom node and the bandwidth of the bottom link are evenly distributed [50,100], and the cost of unit computing

FIGURE 2: Total profit obtained when  $N_D = N_V/5$ .

resource cS and bandwidth resource bS are 1. The number of service types supported by the underlying network is set to 10. Each underlying node provides one to five service types randomly. According to literature [5, 24], the processing time of a service instance depends on the type of the service and the processing capacity of the network node. The transmission delay of the underlying link is proportional to the Euclidean distance between the two endpoints of the link, which is set according to the above principle and ranges from 1 to 10 time units.

The arrival process of service requests obeys the Poisson distribution, with an average of four requests arriving within 100 time units, and the duration of each service request obeys an exponential distribution with an average of 1000 time units. The service chain is composed of four services, the service type is random and nonrepetitive, the computing power required by each service is evenly distributed in [1, 50], and the bandwidth required by logical link is evenly distributed in [1, 50], and the charges cR and bR for unit computing power and unit bandwidth are both 1. The maximum allowable end-to-end delay Dmax is set to 100 time units. The time of each simulation experiment was about 50000 time units, and the data were recorded every 4000 time units from the 2000 time unit. Each group was set up for 10 simulation experiments, and the experimental results were averaged.

The population size was set as 100, the upper limit of iterations was set as 10000, and the weight parameters  $w_N$  and  $w_L$  of load intensity in Equation (10) are set as 0.5 and 0.5.

**4.2. Experimental Results.** To demonstrate the performance of the proposed algorithm, three compared algorithms are introduced. The first literature [27] proposes a novel algorithm to map NSCs to the network infrastructure while allowing possible decompositions of network functions. The algorithm is based on integer linear programming (ILP) which minimizes the cost of the mapping. To solve the scalability issue of the ILP formulation, it targets to minimize the mapping cost by making a reasonable selection of

the network function decompositions, represented as DSBM. Similarly, literature [28] proposes a novel backtracking heuristic algorithm for virtual network composition. Based on this algorithm, two approaches with two different objectives are presented. The first approach (Backtracking-CR) was aimed at composing a virtual network using the least amount of network resources, while the second (Backtracking-LB) applies load balancing for virtual network composition. Furthermore, a linear programming approach that optimizes the virtual network composition with an objective of using the least amount of network resources is presented and used to bench mark the heuristic algorithm.

The number of data center nodes is fixed as  $N_D = N_V/5$ ,  $N_D = 2N_V/5$ ,  $N_D = 3N_V/5$ , and  $N_D = 4N_V/5$ . In each experiment, number of VNF-SCs is set as  $N_R = \rho N_V (N_V - 1)$ , and  $\rho = 0.25, 0.5, 1, 2$ , and  $4$ , respectively. Figures 2–5 show the profit obtained in NSFNET and CHNNET when  $N_D = N_V/5$ ,  $N_D = 2N_V/5$ ,  $N_D = 3N_V/5$ , and  $N_D = 4N_V/5$ .

The transmission delay obtained in NSFNET and CHNNET when  $N_D = N_V/5$ ,  $N_D = 2N_V/5$ ,  $N_D = 3N_V/5$ , and  $N_D = 4N_V/5$  is shown in Figures 6–9, respectively.

Similarly, Figures 10–13 show the load degree in NSFNET and CHNNET when  $N_D = N_V/5$ ,  $N_D = 2N_V/5$ ,  $N_D = 3N_V/5$ , and  $N_D = 4N_V/5$ .

**4.3. Experimental Analysis.** In simulation experiments, the underlying network builds a service path of long-term average income, and the success rate of average build, average cost, revenue cost ratio, average load intensity, and the end-to-end delay of the main evaluation index were measured and recorded, showing that they change over time, so as to compare different construction strategy of long-term running effect.

Main reason is that when only target path delay, ignoring the underlying network resources and effective use of load balancing, hard to avoid service path through long and resources fragments, leads to excessive consumption of resources, uneven load, and easy generation resource bottleneck, the path of service building success rate and long-term benefits is greatly reduced. And only to load strength

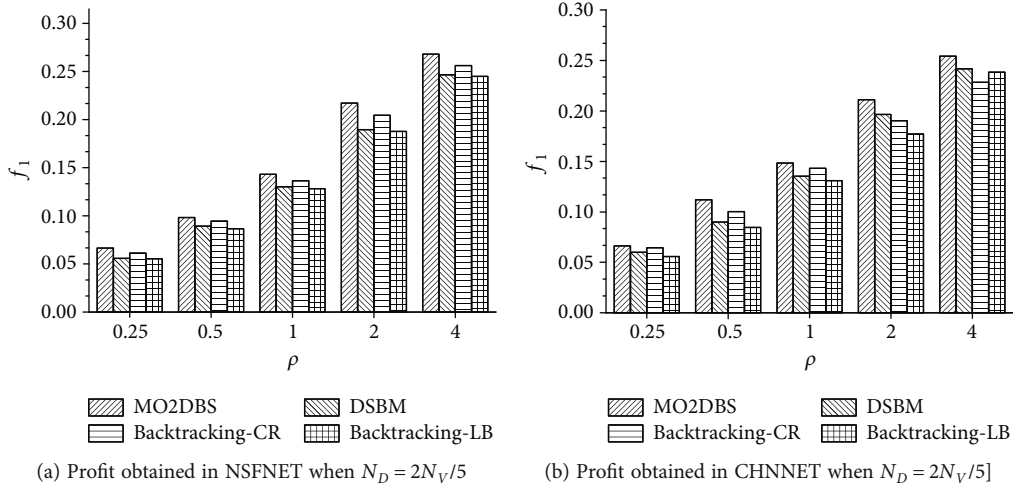


FIGURE 3: Total profit obtained when  $N_D = 2N_V/5$ .

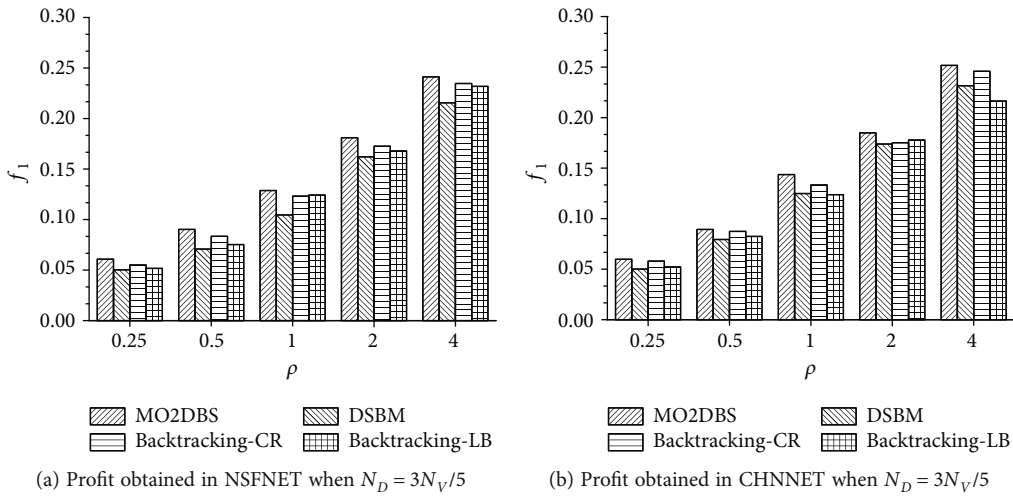


FIGURE 4: Total profit obtained when  $N_D = 3N_V/5$ .

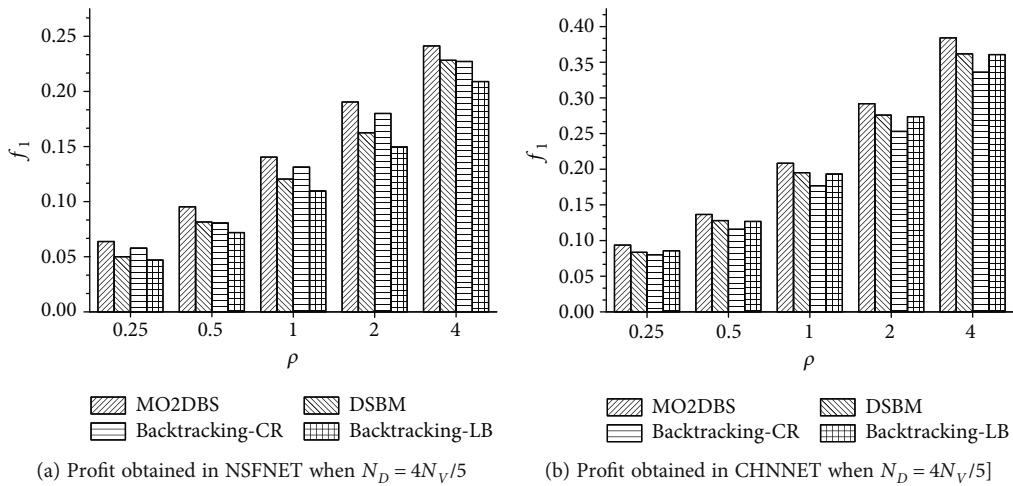
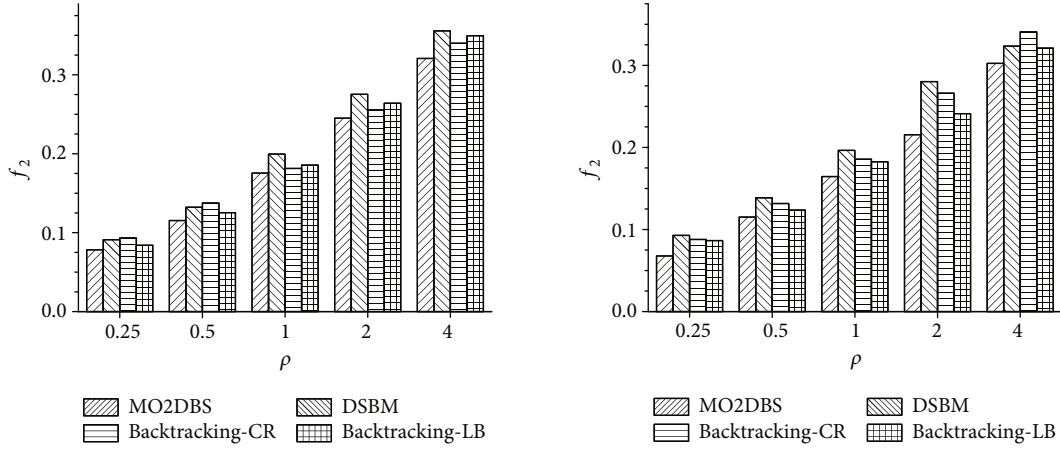
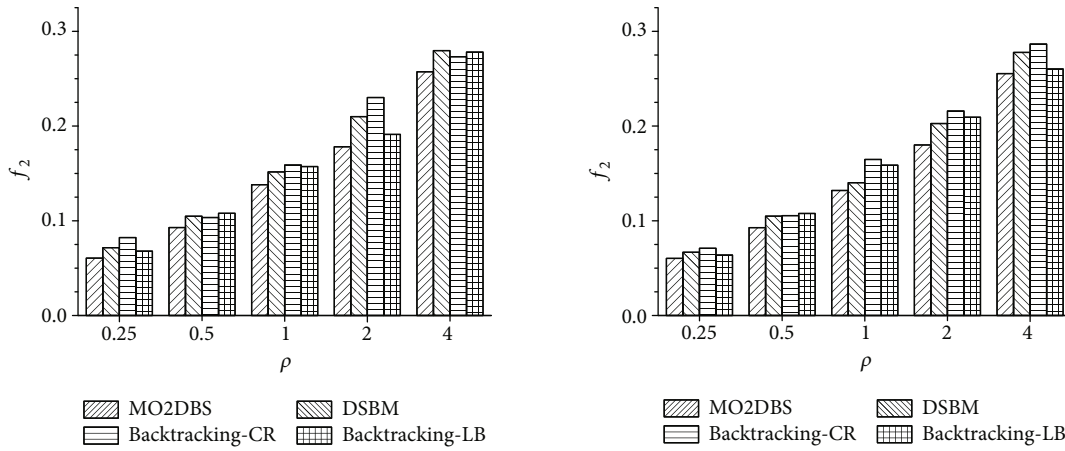


FIGURE 5: Total profit obtained when  $N_D = 4N_V/5$ .



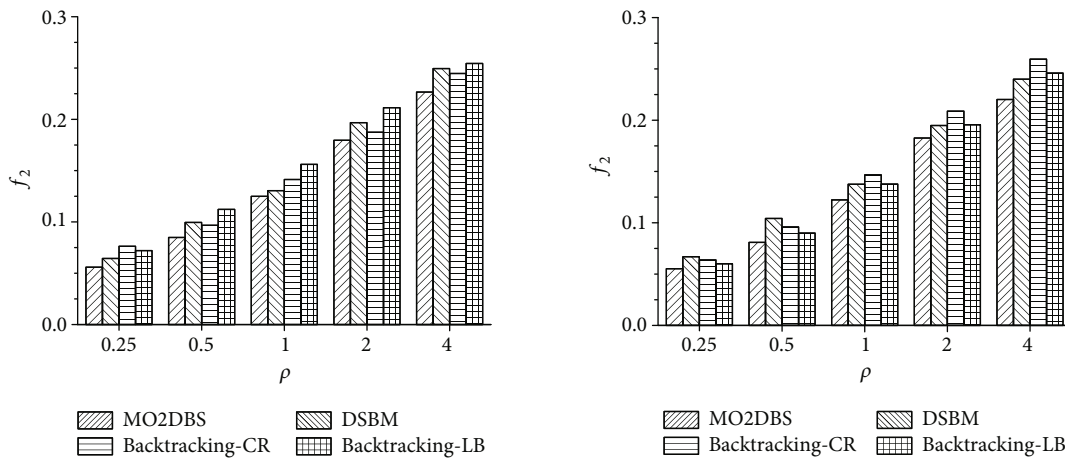
(a) Transmission delay obtained in NSFNET when  $N_D = N_V/5$  (b) Transmission delay obtained in CHNNET when  $N_D = N_V/5$

FIGURE 6: Transmission delay obtained when  $N_D = N_V/5$ .



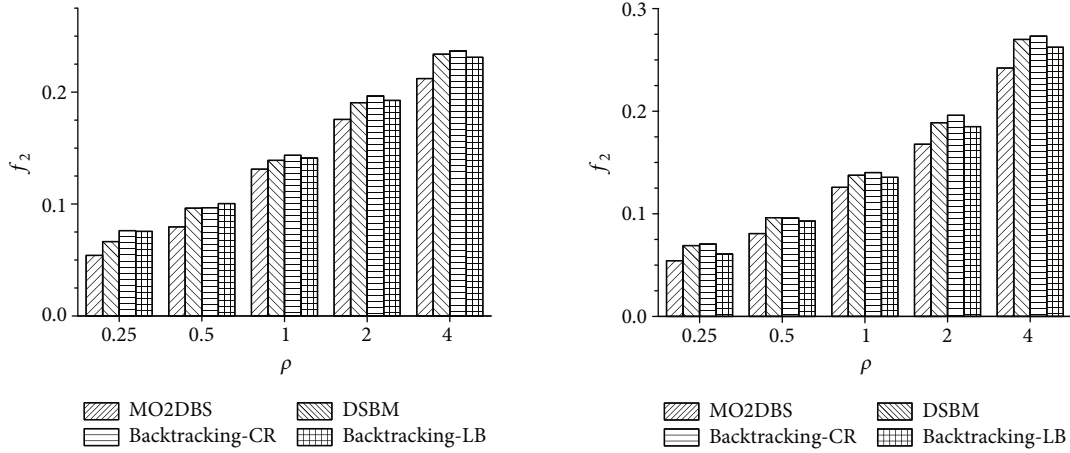
(a) Transmission delay obtained in NSFNET when  $N_D = 2N_V/5$  (b) Transmission delay obtained in CHNNET when  $N_D = 2N_V/5$

FIGURE 7: Transmission delay obtained when  $N_D = 2N_V/5$ .



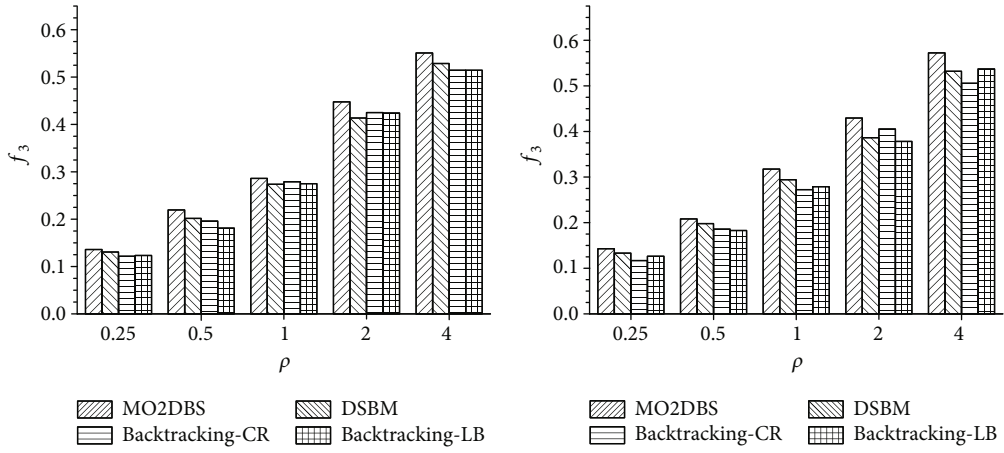
(a) Transmission delay obtained in NSFNET when  $N_D = 3N_V/5$  (b) Transmission delay obtained in CHNNET when  $N_D = 3N_V/5$

FIGURE 8: Transmission delay obtained when  $N_D = 3N_V/5$ .



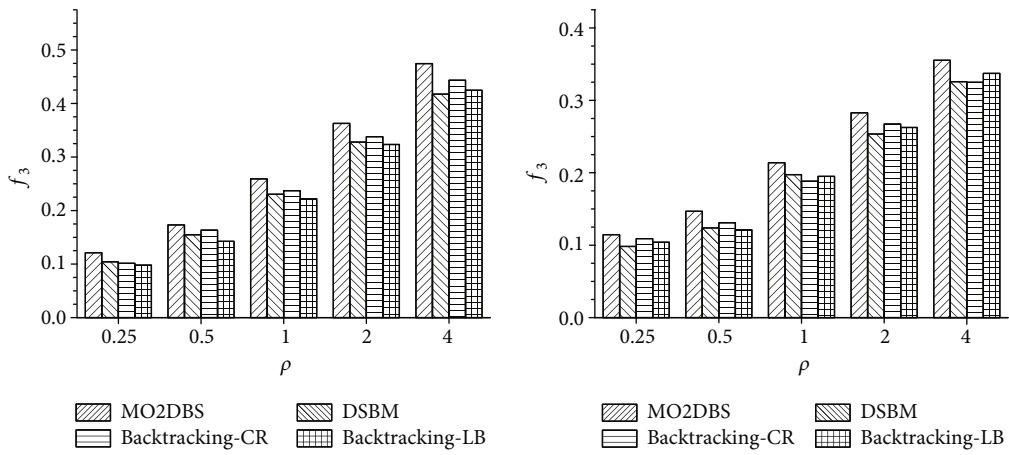
(a) Transmission delay obtained in NSFNET when  $N_D = 4N_V/5$  (b) Transmission delay obtained in CHNNET when  $N_D = 4N_V/5$

FIGURE 9: Transmission delay obtained when  $N_D = 4N_V/5$ .



(a) Load degree obtained in NSFNET when  $N_D = N_V/5$  (b) Load degree obtained in CHNNET when  $N_D = N_V/5$

FIGURE 10: Load degree obtained when  $N_D = N_V/5$ .



(a) Load degree obtained in NSFNET when  $N_D = 2N_V/5$  (b) Load degree obtained in CHNNET when  $N_D = 2N_V/5$

FIGURE 11: Load degree obtained when  $N_D = 2N_V/5$ .



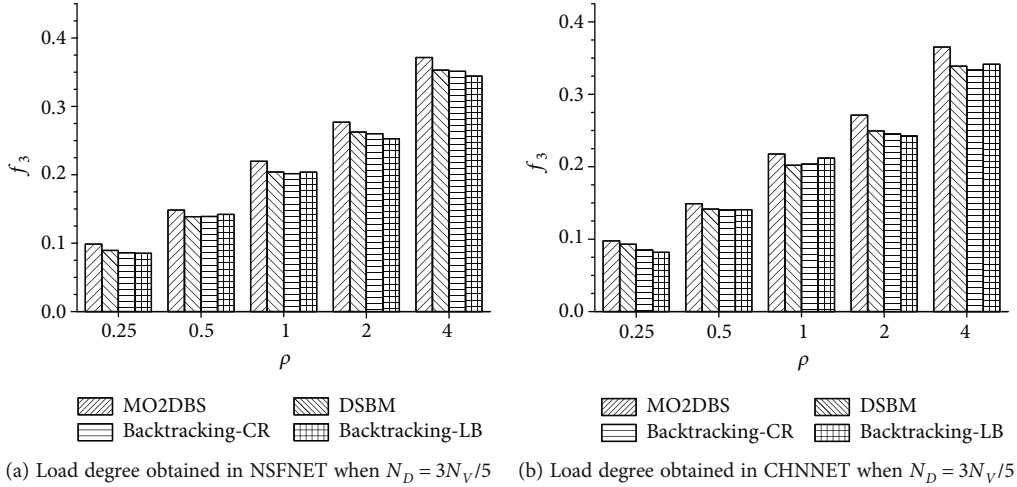


FIGURE 12: Load degree obtained when  $N_D = 3N_V/5$ .

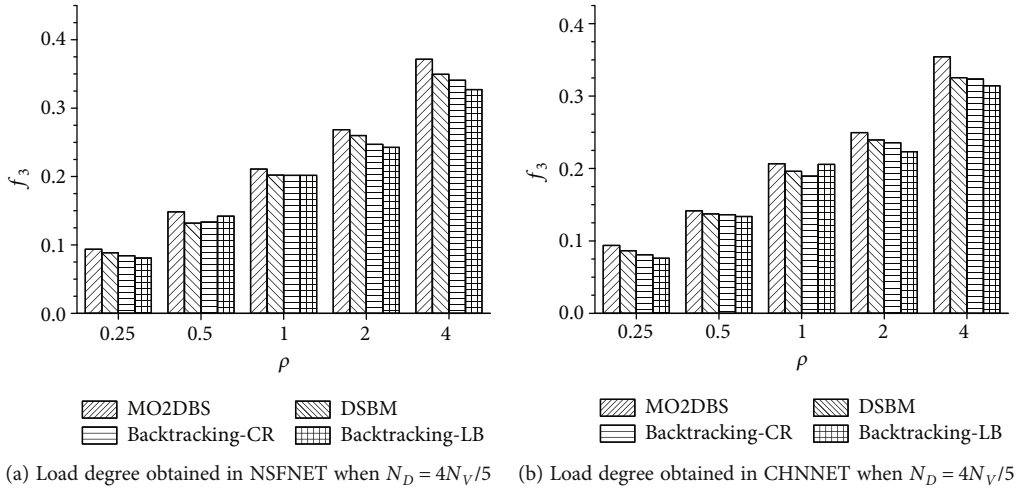


FIGURE 13: Load degree obtained when  $N_D = 4N_V/5$ .

compared to optimize the target scene, with the aim to optimize the path overhead scene only consider only logical link by mapping the underlying network path length, because the algorithm allows multiple services mapping to the same underlying node, therefore, the probability of a single underlying node load multiple services to improve, make the path length shorter, link resource consumption is reduced. The resources saved make it possible to build more service paths, so the success rate of service path building is slightly higher. However, since the former does not consider the load of nodes and links, the possibility of generating resource fragments increases and the probability of receiving service requests with large resource demands decreases. Therefore, the long-term average income of the two is relatively close.

When only a single optimization objective is considered, the algorithm achieves the best performance in the corresponding evaluation indexes, and when only the path cost and load intensity are considered, the two algorithms achieve close performance. Figure 5 shows the average income and the ratio of average cost. The index can reflect the resources used for the efficiency of the algorithm, only

target path overhead. The algorithm achieved the lowest cost and highest earnings; therefore, revenue/cost ratio is highest, which can be the most efficient use of the resources, but only to delay as the goal, the highest cost, income minimum, resource utilization are the worst. According to the above comparison and analysis, if only a single optimization objective is considered when constructing the service path, it is difficult to achieve the mapping effect of low overhead, low latency, high income, and high reception rate. Therefore, the three factors need to be considered comprehensively. (1) Both of them are optimized to minimize end-to-end delay and only consider to avoid overuse of resources, without considering efficient resource allocation and load balancing. (2) Although the candidate graphs constructed by both of them are different (hierarchical graph and step search graph), Dijkstra shortest path algorithm is applied to construct the service path on the candidate graph, and the resources are reserved during the construction of the shortest path spanning tree. Due to the NP complexity of service path construction and the insufficiency of Dijkstra algorithm in dealing with resource reservation, effective service paths

have not been discovered and constructed in both of them, resulting in low success rate and low profit of construction. As in hierarchical graph, in order to minimize the time delay, Dijkstra algorithm in the map service does not take into account the service load node at this time, resource utilization, minimum value is always the option of vertical side, in the corresponding node at the same time to make the same request multiple services, resource consumption too fast, the formation of resource bottleneck, affect the subsequent service path to build. From the perspective of the algorithm, the reasons are as follows: (1) the algorithm utilizes the population evolution effect of the particles to expand the search scope and can effectively find the existing service path. (2) The algorithm comprehensively considers the two factors of service path cost and load intensity, reduces link resource consumption, balances network load, and can effectively improve revenue and build more service paths.

As the number of service requests increases and the amount of available resources decreases, the end-to-end delay increases gradually. As the resource bottleneck appears, the number of failed service path construction increases. When the curve flattens until convergence, it indicates that no new service path is constructed. Its long-term average revenue and cost-of-income ratio were significantly lower than those of algorithm, and its average cost and load intensity were significantly higher than those of the algorithm. The main reason is that the algorithm does not consider the path length and load of the underlying network when initializing and updating particle positions, resulting in high link resource consumption and many resource fragments.

**4.4. Complexity of Proposed Algorithm.** In the proposed algorithm,  $K$  shortest paths should be calculated for each connection request in advance, and its complexity is  $o(K(N^s)^2)$ . There are  $N_R$  connection requests, so the complexity for all connection requests for calculating  $K$  shortest paths is  $o(K(N^s)^2 N_R)$ . The fitness function calculation in the proposed algorithm remains the most complicated, and its complexity is  $o(2G_m P_s (N^s)^2 N_F)$ , where  $G_m$ ,  $P_s$ , and  $N_F$  denote iteration times, population size, and maximum of frequency slots. Therefore, the complexity of proposed algorithm is  $O(K(N^s)^2 N_R + 2G_m P_s (N^s)^2 N_F)$ .

## 5. Conclusion

Network virtualization functions to the network layer and transport layer network function in the form of a software unit in the core network routers or server; using the controller to the function of network to carry on the arrangement and combination, to build an end-to-end service path, and to support a variety of custom service solves the present middle box deployment flexibility and scalability of the defect. Aiming at the key problem of service path construction in the above service delivery mechanism, this paper, from the perspective of reducing delay, reducing overhead, and balancing load, transforms multiple indexes into a single service path quality evaluation index by weighted sum

method and establishes an integer linear programming model of service path construction problem. Then, a discrete particle swarm optimization model for service path construction was established according to the characteristics of particle swarm parallel search, and the proposed algorithm was designed for service path construction. In order to further reduce resource consumption, reduce the probability of resource fragmentation, and improve the convergence speed of particle swarm optimization, an optimization strategy was proposed to guide the initialization and updating of particle positions. Combined with the proposed algorithm, the proposed algorithm was obtained. Simulation results show that compared with the existing service path construction algorithms, the proposed algorithm can effectively optimize the comprehensive quality of service path and improve the success rate of service path construction and long-term average return. Compared with the rand-proposed algorithm, which uses random method to initialize and update particle positions, the proposed algorithm formulates evaluation criteria for candidate nodes and paths, which provides effective guidance for particle flight and further optimizes the performance of the algorithm.

## Data Availability

All the data can be found in the paper.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by the Doctoral Research Initiation Program of Weinan Normal University (Nos. 20RC15 and 20RC14) and the Third Group of Outstanding Talents Supporting Projects in Shaanxi Colleges and Universities (20RC03).

## References

- [1] R. Mijumbi, J. Serrat, J. L. Gorricho, N. Bouten, F. D. Turck, and R. Boutaba, "Network function virtualization: state-of-the-art and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 236–262, 2016.
- [2] Y. Li and M. Chen, "Software-defined network function virtualization: a survey," *IEEE Access*, vol. 3, pp. 2542–2553, 2017.
- [3] H. Xuan, X. Zhao, Z. Liu, J. Fan, and Y. Li, "Energy efficiency opposition-based learning and brain storm optimization for vnf-sc deployment in iot," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6651112, 9 pages, 2021.
- [4] I. F. Akyildiz, S. C. Lin, and P. Wang, "Wireless software-defined networks (W-SDNs) and network function virtualization (NFV) for 5G cellular systems: an overview and qualitative evaluation," *Computer Networks*, vol. 93, pp. 66–79, 2015.
- [5] M. Xia, M. Shirazipour, Y. Zhang, H. Green, and A. Takacs, "Optical service chaining for network function virtualization," *IEEE Communications Magazine*, vol. 53, no. 4, pp. 152–158, 2015.

- [6] Y. Hui, Z. Jie, Y. Ji, T. Rui, and Y. Lee, "Performance evaluation of multi-stratum resources integration based on network function virtualization in software defined elastic data center optical interconnect," *Optics Express*, vol. 23, no. 24, p. 31192, 2015.
- [7] H. Cao, Y. Zhu, Z. Gan, and L. Yang, "A novel optimal mapping algorithm with less computational complexity for virtual network embedding," *IEEE Transactions on Network & Service Management*, vol. 15, no. 1, pp. 356–371, 2018.
- [8] C. Liang, F. R. Yu, and X. Zhang, "Information-centric network function virtualization over 5G mobile wireless networks," *IEEE Network*, vol. 29, no. 3, pp. 68–74, 2015.
- [9] S. Sun, M. Kadoch, L. Gong, and B. Rong, "Integrating network function virtualization with SDR and SDN for 4G/5G networks," *IEEE Network*, vol. 29, no. 3, pp. 54–59, 2015.
- [10] X. Fu, R. Yu, J. Wang, Q. Qi, and J. Liao, "Performance optimization for blockchain-enabled distributed network function virtualization management and orchestration," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6670–6679, 2020.
- [11] B. Chen, J. Zhang, W. Xie, J. P. Jue, Y. Zhao, and G. Shen, "Cost-effective survivable virtual optical network mapping in flexible bandwidth optical networks," *Journal of Lightwave Technology*, vol. 34, no. 10, pp. 2398–2412, 2016.
- [12] Y. Yu, X. Bu, K. Yang, H. K. Nguyen, and Z. Han, "Network function virtualization resource allocation based on joint benders decomposition and ADMM," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 2, pp. 1706–1718, 2020.
- [13] H. Hawilo, M. Jammal, and A. Shami, "Exploring microservices as the architecture of choice for network function virtualization platforms," *IEEE Network*, vol. 33, no. 2, pp. 202–210, 2019.
- [14] J. Wu, M. Dong, K. Ota, J. Li, W. Yang, and M. Wang, "Fog-computing-enabled cognitive network function virtualization for an information-centric future internet," *IEEE Communications Magazine*, vol. 57, no. 7, pp. 48–54, 2019.
- [15] M. Karimzadeh-Farshbafan, V. Shah-Mansouri, and D. Niyato, "A dynamic reliability-aware service placement for network function virtualization (NFV)," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 2, pp. 318–333, 2020.
- [16] T. Lin and Z. Zhou, "Robust network function virtualization," *Networks*, vol. 75, no. 4, pp. 438–462, 2020.
- [17] Y. Li, Y. Zhao, B. Li et al., "Joint balancing of it and spectrum resources for selecting virtualized network function in inter-datacenter elastic optical networks," *Optics Express*, vol. 27, no. 11, pp. 15116–15128, 2019.
- [18] C. Yao, X. Wang, Z. Zheng, G. Sun, and L. Song, "EdgeFlow: open-source multi-layer data flow processing in edge computing for 5G and beyond," *IEEE Network*, vol. 33, no. 2, pp. 166–173, 2019.
- [19] Z. Luo, C. Wu, Z. Li, and W. Zhou, "Scaling geo-distributed network function chains: a prediction and learning framework," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 8, pp. 1838–1850, 2019.
- [20] D. M. Manias and A. Shami, "Making a case for federated learning in the internet of vehicles and intelligent transportation systems," *IEEE Network*, vol. 35, no. 3, pp. 88–94, 2021.
- [21] L. Gu, D. Zeng, S. Tao et al., "Fairness-aware dynamic rate control and flow scheduling for network utility maximization in network service chain," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 5, pp. 1059–1071, 2019.
- [22] S. Cheng, Q. Qin, J. Chen, and Y. Shi, "Brain storm optimization algorithm: a review," *Artificial Intelligence Review*, vol. 46, pp. 445–458, 2015.
- [23] H. Duan, S. Li, and Y. Shi, "Predator-prey brain storm optimization for dc brushless motor," *IEEE Transactions on Magnetics*, vol. 49, no. 10, pp. 5336–5340, 2013.
- [24] X. Xue and Y. Wang, "Optimizing ontology alignments through a memetic algorithm using both matchfmeasure and unanimous improvement ratio," *Artificial Intelligence*, vol. 223, pp. 65–81, 2015.
- [25] J. Xue, Y. Wu, Y. Shi, and C. Shi, "Brain storm optimization algorithm for multi-objective optimization problems," *Lecture Notes in Computer Science*, vol. 7331, no. 4, pp. 513–519, 2012.
- [26] M. El-Abd, "Global-best brain storm optimization algorithm," *Swarm and Evolutionary Computation*, vol. 37, pp. 5336–5340, 2017.
- [27] A. Ss, A. Wt, B. Mr et al., "Network service chaining with optimized network function embedding supporting service decompositions," *Computer Networks*, vol. 93, pp. 492–505, 2015.
- [28] A. Hammad, R. Nejabati, and D. Simeonidou, "Novel methods for virtual network composition," *Computer Networks*, vol. 67, no. jul. 4, pp. 14–25, 2014.

## Research Article

# Study of Water Resources Optimal Operation Model of Multireservoir: A Case Study of Kuitun River Basin in Northwestern China

Changlu Qiao <sup>1,2</sup>, Yan Wang <sup>1,3</sup>, Yanxue Liu <sup>1,2</sup>, Junfeng Li <sup>1,2</sup>, Heping Zhang <sup>3</sup>  
and Jiangang Lu <sup>3</sup>

<sup>1</sup>School of Water Conservancy & Architectural Engineering, Shihezi University, Shihezi, Xinjiang 832003, China

<sup>2</sup>Xinjiang Production and Construction Group Key Laboratory of Modern Water-Saving Irrigation, Shihezi 832003, China

<sup>3</sup>Xinjiang Fukang Pumped Storage Power Company Limited, Fukang 831500, China

Correspondence should be addressed to Yan Wang; wangyan930510@qq.com and Yanxue Liu; 1064896050@qq.com

Received 2 March 2022; Accepted 10 May 2022; Published 6 June 2022

Academic Editor: Kingsi Xue

Copyright © 2022 Changlu Qiao 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.

Aiming at the problems that need to be solved urgently in the current operation of a multireservoir in Kuitun River Basin, such as the uneven distribution of water resources in time and space, the large workload of manual operation calculation, and low coordination level, the paper takes the optimal operation of water resources in the basin as the main goal and carries out the research on the optimal operation model of the multireservoir in combination with the complex characteristics of local water resources system. Firstly, based on the generalization of hydraulic engineering in Kuitun River Basin, a water resources optimal operation model of the multireservoir is established and is solved by the graph theory. Then, the actual data of typical years were selected to test the model. The test results show that, compared with the actual water distribution, the water shortage rate of 2015 and 2016 in high flow years decreased by 98.57% and 100%, respectively; the water shortage rate of 2013 and 2014 in normal flow years decreased by 92.65% and 96.38%, respectively; and the water shortage rate of 2009 in a low flow year decreased by 87.78%. The model can provide the optimal operation scheme for the optimal operation of the multireservoir in the basin. And it can solve the problems such as the uneven distribution of water resources and the large workload of manual operation calculation and can provide technical support for the optimal operation of water resources of the multireservoir in Kuitun River Basin in the future.

## 1. Introduction

The function of reservoir operation is more and more prominent in the management of the multireservoir. How to maximize the function of reservoir has become one of the hot topics [1]. Formulating the optimal operation scheme of the multireservoir will become an effective method in operation and management of the multireservoir [2, 3]. At present, twelve reservoirs have been built in the Kuitun River Basin to solve various problems caused by the uneven distribution of water resources in time and space. However, the operation scheme of each reservoir was formulated only from the perspective of its own benefit, instead of the benefit of the whole multireservoir. The water resources of the mul-

tireservoir is always distributed according to the real-time situation and the experience, which is difficult to achieve optimal operation and resulted in unbalanced water supply and waste of water resources in the later stage. Therefore, in view of the present situation of the Kuitun River Basin, it is of great practical significance to study the optimal operation model of the multireservoir in this basin.

In recent years, scholars make a series of research and practices to study the optimal operation of the multireservoir. In general, connecting the scattered reservoirs into a whole: multireservoir, and comprehensively optimizing the multireservoir with different methods can improve the utilization rate of water resources and improve the overall regional benefit. Kumar et al. [4] used the simulation



optimization method to optimize the operation of reservoirs in several basins of the Indian Peninsula. The study showed that the utilization efficiency of water resources was significantly improved when the reservoirs were united as one. Ye and He [5] put forward an optimal operation model of the multireservoir water supply based on particle swarm optimization. The results showed that the PSO algorithm and the new model could obtain reliable and efficient optimization results. Goor et al. [6] used the stochastic programming method to optimize the operation of the reservoir system in the east Nile River Basin. The optimized scheme increased the area of irrigation district by 5.5%. Yin et al. [7] put forward a general plan for the operation of large reservoirs in the Yangtze River Basin, which included the objectives, principles, and operation scheme of reservoir operation, and provided a comprehensive reference for the operation of large reservoirs in the future. Bai et al. [8] used the successive approximation method of dynamic programming to propose a synergistic benefit scheme for two key reservoirs in the Yellow River Basin during their operation in different situations. Li and Ouyang [9] proposed a generalized multiobjective flood control model (MOFCM) for joint optimal operation of cascade reservoirs in the lower reaches of the Jinsha River and the Three Gorges of Yangtze River, which realized the optimal operation of main and tributary reservoirs. Thechamani et al. [10] used nontime modeling methods to model and optimize the operation of the multireservoir in Chaoshan River Basin. Yekit [11] had formulated reservoir optimal operation strategies related to the Subak irrigation scheme water supply to support agricultural productivity at upstream, midstream, and downstream. Wang et al. [12] had carried out two sets of joint operation rules (JOR-I and JOR-II) for the multireservoir in Liaoning Province. The results showed that JOR-I was suitable for the operation of large reservoirs with large runoff and JOR-II was suitable for the operation of small reservoirs with small runoff, which provided guidance for the management of reservoir systems. At present, genetic algorithm, cuckoo algorithm, frog jump algorithm, and improved heuristic algorithm are the most popular optimization algorithms for the multireservoir, but the heuristic algorithm has many shortcomings such as low accuracy and instability. Zhou et al. [13] proposed a graph theory to solve the integration problem of the multireservoir and relationship; they applied it to the integration of the multireservoir flood forecasting and operation system, and it obtained good results. Based on the graph theory, the node graph is established, and the topological relationship among adjacency table, adjacency matrix, and correlation matrix can be used to effectively solve the water distribution problem between the multireservoir and irrigation districts.

## 2. Materials and Methods

*2.1. Overview of the Study Area.* The Kuitun River Basin is located in the southwestern margin of Junggar Basin on the northern slope of Tianshan Mountains, Xinjiang. It is bordered by Turgou and Bayingou River Basin in the east, Toto River Basin in the west, Kashi River Basin of Yili in

the south, and the watershed of Mayierli Mountain and Zaire Mountain in the north [14]. Geographical coordinates are  $83^{\circ}22'00''$ - $85^{\circ}47'00''$  in the east longitude and  $43^{\circ}30'00''$ - $45^{\circ}04'00''$  in the north latitude. The main stream of Kuitun river is 360 km long, and the total area of the basin is  $2.83 \times 10^4$  km<sup>2</sup>. The average temperature in this district is 7°C, the average temperature in January is -16°C, the average temperature in July is 26°C, the annual precipitation is 150~170 mm, and the annual evaporation is 1710~1930 mm.

The main stream of Kuitun River are composed of the Guertu river, Sikeshu river, and Kuitun river. The runoff of each river is greatly affected by seasons, and the interannual variation is small. The total annual runoff of the three rivers is about  $1.256 \times 10^9$  m<sup>3</sup>, accounting for 80.9% of the total water in the Kuitun River Basin. At present, there are twelve both large and small reservoirs in the Kuitun River Basin, one large (2) type reservoir, six medium-sized reservoirs, and five small (2) type reservoirs (Figure 1).

### 2.2. Data Sources

*2.2.1. Sources of Hydrological Data.* All hydrological data used in this paper are from the Irrigation Management Department of Water Conservancy Project in Kuitun River Basin, Xinjiang.

*2.2.2. Engineering Data Sources.* All hydraulic engineering data used in this paper are from the Irrigation Management Department of Water Conservancy Project in Kuitun River Basin, Xinjiang, as shown in Table 1.

*2.3. Model Construction.* According to the actual situation of Kuitun River Basin, the multireservoir of Kuitun River Basin is generalized. On the premise of ensuring the ecological water use in the downstream of Kuitun River Basin, the optimal operation model of the multireservoir in Kuitun River Basin is established with the goal of minimizing the water shortage rate in the irrigation district by using the coordination decomposition theory of a large-scale system, and the physical model is transformed into mathematical model by graph theory and solved by computer.

*2.3.1. General Thought of Model Construction.* The hydraulic engineering in the study area is complex, and there are both series and parallel relationships between reservoirs. According to the relationship between supply and demand, it is divided into three subsystems, namely, subsystem 1, subsystem 2, and subsystem 3.

- (i) Subsystem 1: the main water user is Liugou irrigation district. Its water supply sources are the Guertu river and Sikeshu river, and the upstream reservoir is Liugou reservoir.
- (ii) Subsystem 2: the main water user is Chepaizi irrigation district. Its water supply sources are the Guertu river, Sikeshu river, and Kuitun river, and the upstream reservoirs are Liugou reservoir and Huanggou reservoir.

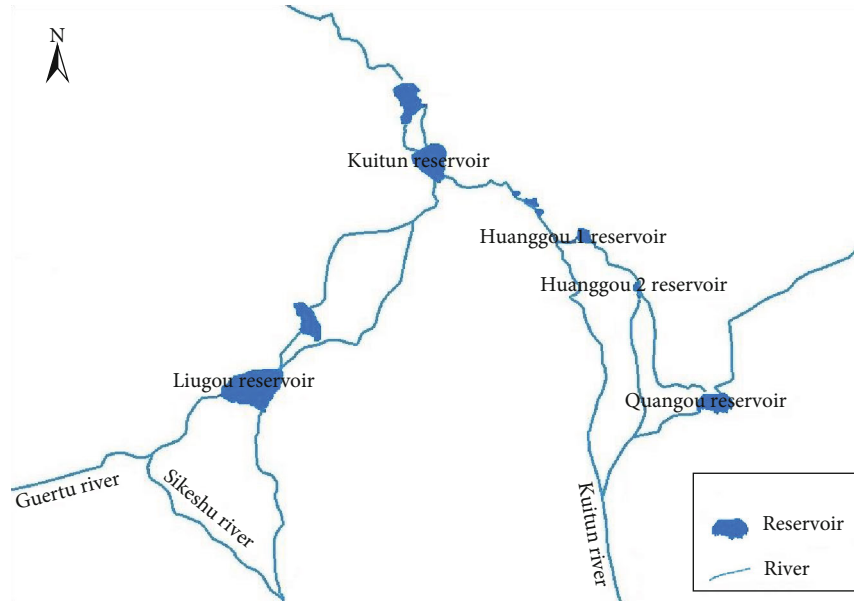


FIGURE 1: Schematic diagram of the multireservoir in Kuitun River Basin.

TABLE 1: The characteristic parameter of reservoirs.

Name of reservoir	L reservoir	K reservoir	C reservoir	H 1 reservoir	H 2 reservoir	Q reservoir
River	S river G river	K river	K river	K river	K river	K river
Normal water level (m)	373.10	318.00	308.68	344.50	362.00	417.50
Dead water level (m)	359.50	311.10	303.00	—	350.00	406.00
Total storage capacity ( $10^4 \text{ m}^3$ )	10200	5000	4000	3220	2481	4000
Flood regulation storage capacity ( $10^4 \text{ m}^3$ )	4900	1119	1000	200	500	721
Benefit storage capacity ( $10^4 \text{ m}^3$ )	10200	5000	4000	1668	2481	4000
Dead storage capacity ( $10^4 \text{ m}^3$ )	0	28	35	100	100	26

- (iii) Subsystem 3: the main water user is Huanggou irrigation district. its water supply source is the Kuitun river, and the upstream reservoirs are Huanggou 1 reservoir and Huanggou 2 reservoir.

The subsystems are related to each other, and there is a feedback regulation relationship between the large-scale system and subsystems. So, the operation model of the whole system is established (Figure 2).

**2.3.2. Multireservoir Combined Water Supply System.** The series and parallel relationships between rivers and reservoirs in Kuitun River Basin are complex, by analyzing the hydraulic relationship between rivers and reservoirs, the characteristics of the water resources system in the basin are studied. The project is generalized by means of nodes and connections, and the relationship between various variables and parameters in the system is expressed by mathematical language and computer language to reflect the actual characteristics of the basin and the hydraulic relationship in the system. Finally, the reservoir is abstracted as a “point” element, and the water diversion and supply route are abstracted as a “line” element to form the joint commis-

sioning node map of the multireservoir in Kuitun River Basin. The network simulation model of water distribution system is built, as shown in Figure 3.

According to water supply and demand, it is divided into three subsystems as shown in Figure 4.

In subsystem 1, the upstream Liugou reservoir and the downstream Dazimiao reservoir are connected in series to provide water distribution for Liugou irrigation district, and the Dazimiao reservoir only regulates the water distribution in Liugou irrigation district, so Liugou reservoir is chosen as the key reservoir. In subsystem 2, the water of Quanguo reservoir can only meet 40%~60% of the water demand of other water users in Huanggou irrigation district, and Quanguo reservoir basically does not supply water to Huanggou irrigation district. Huanggou 1 reservoir and Huanggou 2 reservoir supply water to Huanggou irrigation district in series. Huanggou 1 reservoir and Huanggou 2 reservoir are generalized as Huanggou reservoir, which is regarded as a key reservoir; Quanguo reservoir and other water users are not considered. In subsystem 3, the mid-stream Kuitun reservoir and the downstream Chepaizi reservoir are connected in series to supply water to Chepaizi irrigation district, and Chepaizi reservoir only regulates the



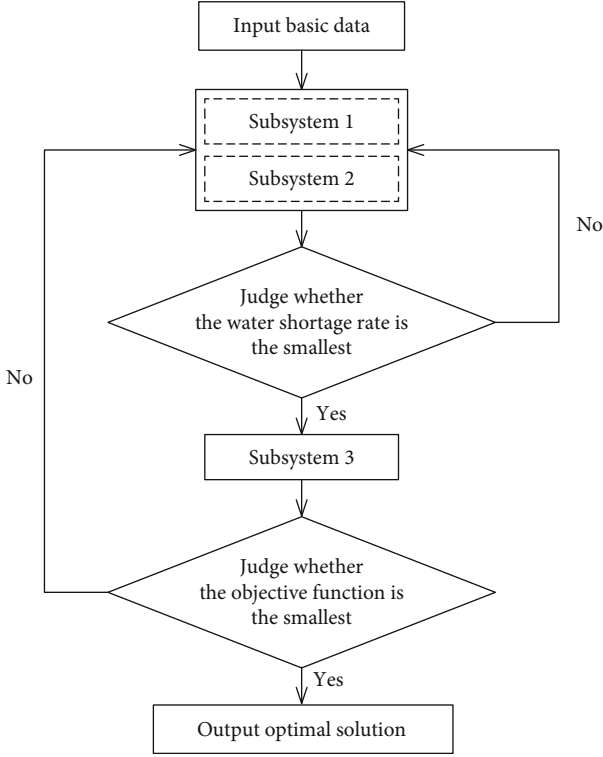


FIGURE 2: Flow chart for solving the model of optimal operation of reservoirs in Kuitun River Basin.

water distribution in Chepaizi irrigation district, so Kuitun reservoir is chosen as the key reservoir. The generalized node diagram of the multireservoir is shown in Figure 5.

**2.3.3. Construction of Optimal Operation Model.** Liugou reservoir and Kuitun reservoir in the multireservoir of Kuitun River Basin are river-blocking reservoirs, while the other reservoirs are plain reservoirs. There are special drainage channels for ecological water use near Liugou reservoir, so only Kuitun reservoir needs to consider the ecological base flow.

Dingwen Tian used the Tennant method to optimize the operation of a power station in Kuerle District of Xinjiang. The ecological base flow downstream of the dam was 10% of the average annual flow at the dam site [15]; Shuzhen Li put forward the rationality of discharging ecological base flow in different time periods under the condition that the total amount of ecological base flow remains unchanged and took Baiyanggou reservoir in Toudao as an example to illustrate its rationality [16]. According to the value of ecological base flow of many local reservoirs in Xinjiang and the situation of ecological water use in the downstream of Kuitun river, the ecological base flow is 10% of the average annual flow of the Kuitun reservoir.

**(1) Water Supply Target of Irrigation District.** Optimal operation of the multireservoir in Kuitun River Basin needs to meet the minimum requirements of agricultural water shortage in irrigation district, namely,

$$\min K_{ij} = \min \left\{ \frac{WD_{ij} - WS_{ij}}{WD_{ij}} \right\}, \quad (1)$$

where  $K_{ij}$  represents the water distribution rate in the  $j$  month of the  $i$ th system,  $WS_{ij}$  represents the water supply in the  $j$  month of the  $i$ th system, and  $WD_{ij}$  represents the water demand in the  $j$  month of the  $i$ th system.

**(2) Objective Function.** Taking the minimum water shortage rate in the irrigation district of Kuitun River Basin as the objective function, it is divided into total objective function and subsystem objective function.

Total objective function:

$$\min f_m = \min \{ \max (f_i) \}. \quad (2)$$

Among them,  $f_m$  represents the objective function and  $f_i$  represents the water shortage rate of the  $i$ th system.

In order to ensure that the water shortage rate of each irrigation district in the basin is the lowest and the whole water distribution is uniform, the total objective function is set as follows: In the same period, the largest water shortage rate of each irrigation district is the smallest, that is, the water shortage rate of each irrigation district is the smallest.

Subsystem objective function:

$$\min f_i(WS_{ij}, WD_{ij}) = \min \left\{ \frac{WD_{ij} - WS_{ij}}{WD_{ij}} \right\}. \quad (3)$$

Among them,  $i = 1, 2, 3$  are Liugou irrigation district, Huanggou irrigation district, and Chepaizi irrigation district, respectively.

**(3) Constraint Conditions.**

**(1) Water balance constraint**

$$V(t+1) = V(t) + (Q(m, t) - q(m, t)) \times \Delta t - S(m, t). \quad (4)$$

Among them,  $S(m, t)$  is the loss of  $m$  reservoir in  $t$  period,  $Q(m, t)$  is the reservoir inflow,  $q(m, t)$  is the reservoir discharge flow,  $Q(m, t) = I(m, t) + Y(m, t)$ ,  $I(m, t)$  is the amount of water transferred from the upstream reservoir to  $m$  reservoir in  $t$  period,  $Y(m, t)$  is the water intake of the river in  $t$  period,  $q(m, t) = Q(m+1, t) + X(m, t)$ , where  $X(m, t)$  is the amount of water supply to  $m$  reservoir in  $t$  period.

**(2) Storage constraint**

$$V_{\min}(m, t) < V(m, t) < V_{\max}(m, t). \quad (5)$$

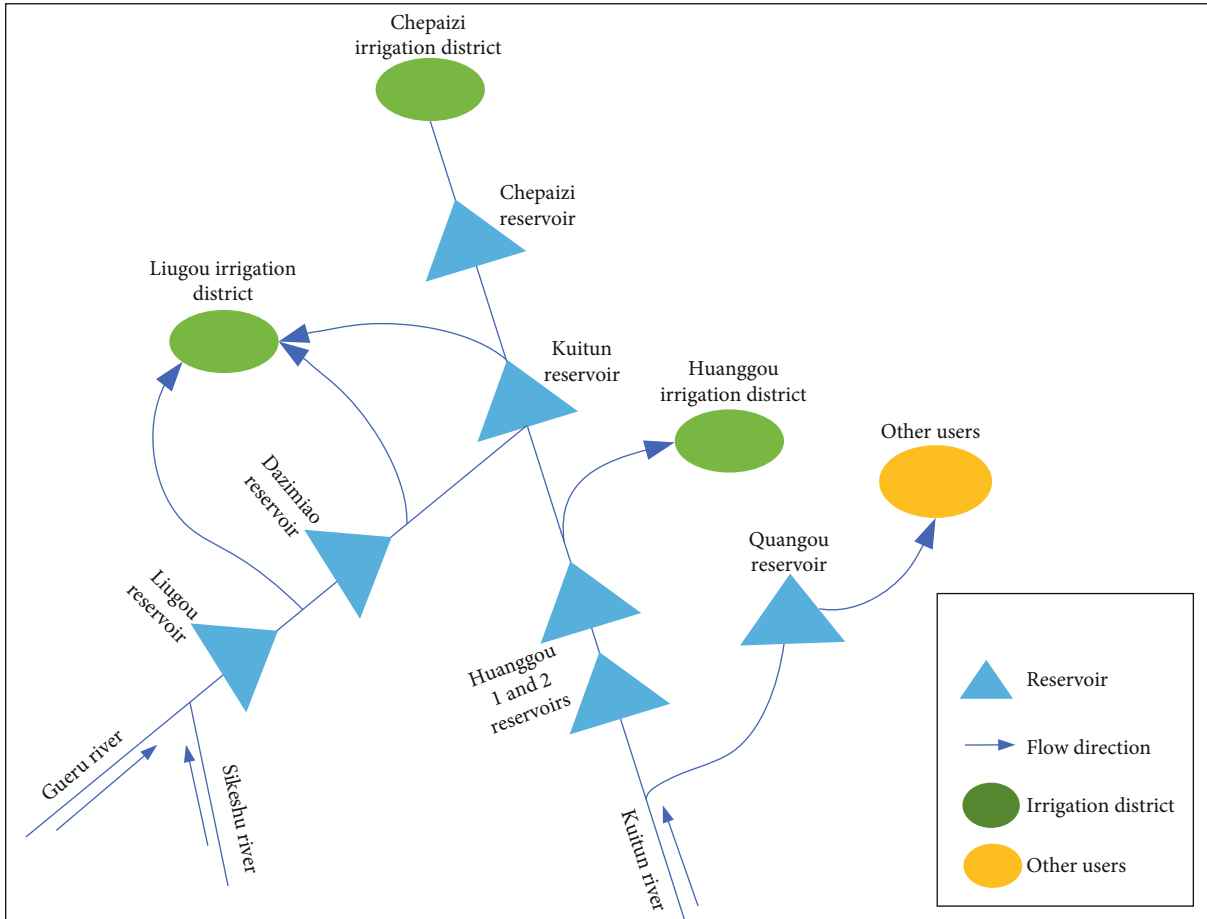


FIGURE 3: The reservoir regulation node graph.

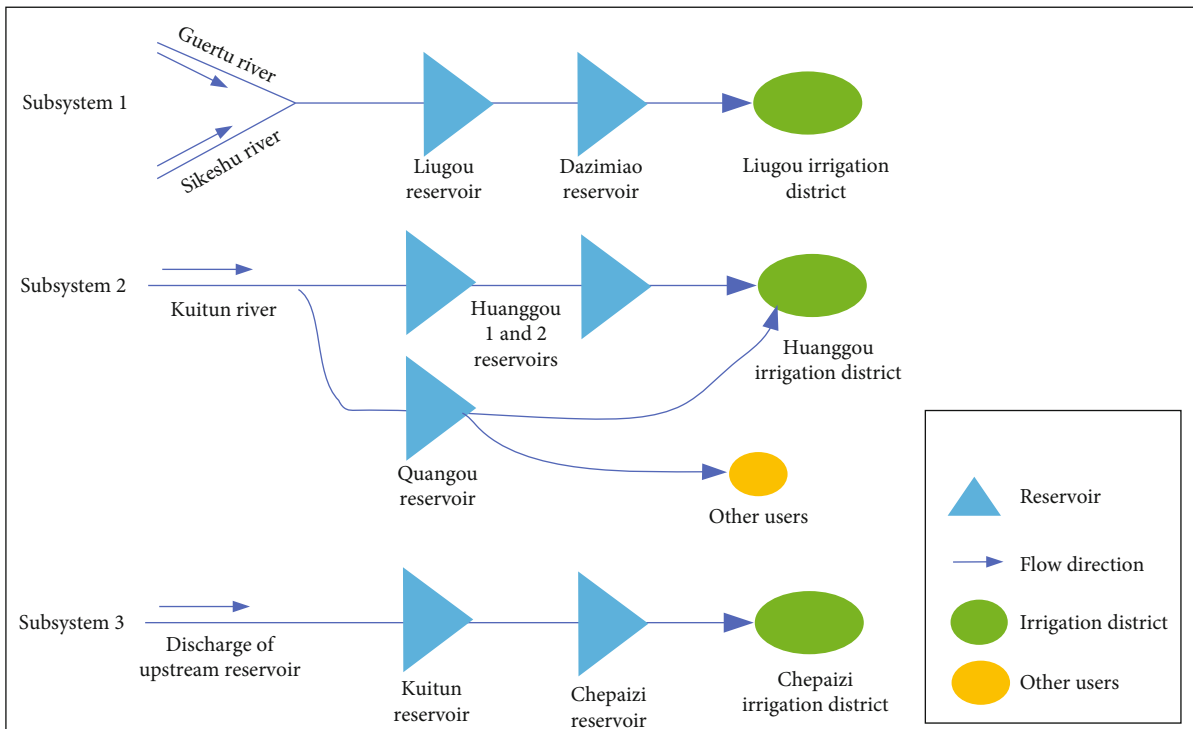


FIGURE 4: Summary schematic of subsystems.

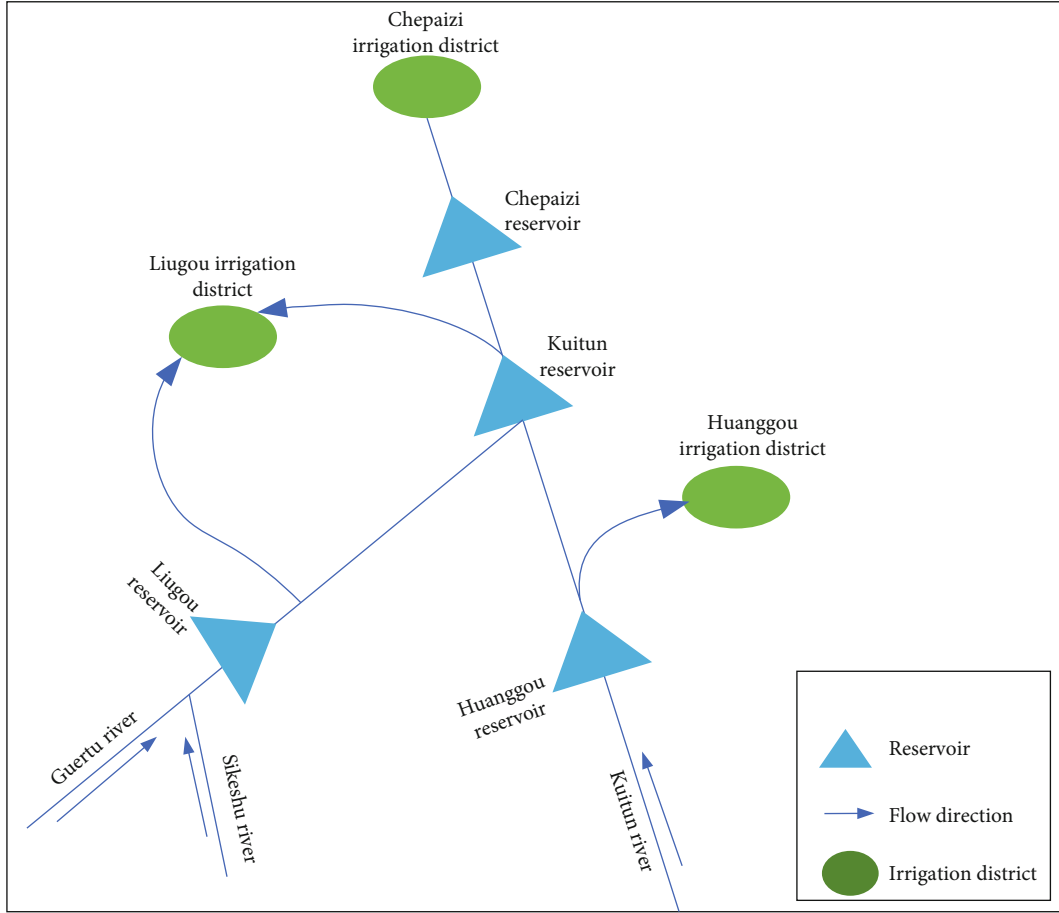


FIGURE 5: Summary diagram of joint operation nodes of the multireservoir.

Among them,  $V(m, t)$ ,  $V_{\max}(m, t)$ , and  $V_{\min}(m, t)$  are the storage capacity, the maximum allowable storage capacity, and the minimum allowable storage capacity of  $m$  reservoir in  $t$  period.

(3) Water level constraint

$$Z_{\min}(m, t) < Z(m, t) < Z_{\max}(m, t). \quad (6)$$

Among them,  $Z(m, t)$ ,  $Z_{\max}(m, t)$ , and  $Z_{\min}(m, t)$  are the water level of  $m$  reservoir in  $t$  period, normal water level, and dead water level.

(4) Channel diversion flow constraint

$$Q(m, t) < q. \quad (7)$$

Among them,  $Q(m, t)$  represents inflow;  $q$  is the ability of the channel to divert water into the warehouse.

(5) Ecological base flow constraint

$$q(t) > Q_{\min}(t). \quad (8)$$

Among them,  $q(t)$  is the discharge flow at the end of  $t$  period of Kuitun reservoir;  $Q_{\min}(t)$  is the minimum discharge flow of Kuitun reservoir, which is the ecological base flow.

(6) Nonnegative constraint: all variables are not negative

#### 2.4. Solution for Model

**2.4.1. Solution Method.** The basic idea of graph theory is that the whole multireservoir operation system is regarded as an organic whole connected by nodes with different attributes. The attributes of nodes are determined by the node type. There are two types of nodes: inflow and outflow.

**(1) Node Graph Model.** The inflow types of nodes including (1) runoff prediction, (2) diversion from upstream rivers, (3) depending on outflow of upstream nodes, (4) the superposition of multi-inflow (set this type to reduce data redundancy), and (5) other types (used to expand the attributes of nodes).

The outflow types of nodes including (1) reckoning from reservoir, (2) reckoning from reservoir operation, (3) reckoning from downstream water demand, and (4) other types (used to expand the attributes of nodes).

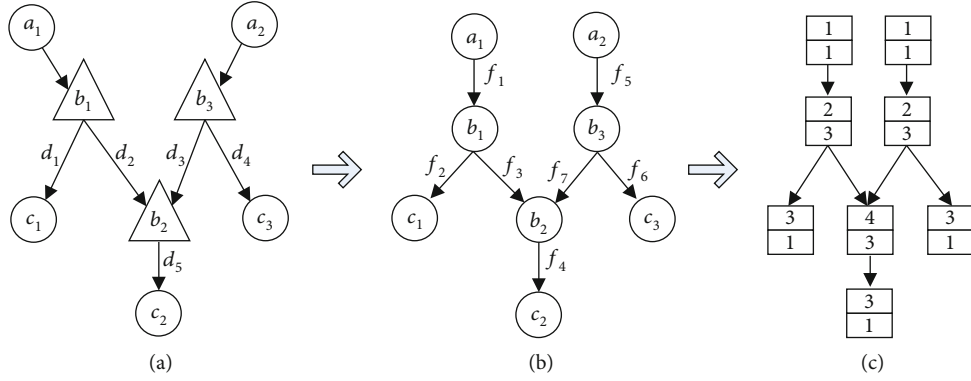


FIGURE 6: Optimal operation model of reservoirs in Kuitun River Basin based on graph theory.

The inflow types and outflow types of a node not only determines the attributes of the node itself but also the relationships between nodes, which forms the basis of system integration.

In the optimal operation model of the multireservoir of Kuitun River Basin based on graph theory (Figure 6):  $a_1$  represents the total amount of water from the Guertu river and the Sikeshu river, and  $a_2$  represents the total amount of water from the Kuitun river.  $b_1$ ,  $b_2$ , and  $b_3$  represent Liugou reservoir, Kuitun reservoir, and Huanggou reservoir, respectively.  $c_1$ ,  $c_2$ , and  $c_3$  represent Liugou irrigation district, Chepaizi irrigation district, and Huanggou irrigation district, respectively. In Figure 6(c), the number above the square is the inflow type, and the number below the square is the outflow type. The model of the subsystem 1~3 based on the graph theory are shown in Figures 7-9.

- Subsystem 1: Liugou irrigation district.
- Subsystem 2: Huanggou irrigation district.
- Subsystem 3: Chepaizi irrigation district.

(2) *Digital Connotation of Node Graph.* The 2-tuple consisted of the point set  $P = \{p_1, p_2, \dots, p_n\}$  and the unordered edge set  $F = \{f_1, f_2, \dots, f_m\}$  are denoted as  $G = (P, F)$ . The  $p_i$  element in  $P$  is called the vertex and the  $f_j$  element in  $F$  is called the edge. If  $p_i$  to  $p_j$  is directed, it is called the directed graph, and the directed edge is denoted as  $\langle p_i, p_j \rangle$ ; otherwise, it is an undirected graph and an undirected edge is denoted as  $(p_i, p_j)$ .

It is particularly important for computer to describe geometric figures with graph  $G = (P, F)$ . There are three main ways of representing the graph: adjacency table, adjacency matrix, and correlation matrix. The adjacency table is more convenient and faster for the database to save operation data.

(1) Adjacency table

The results of Figures 6(b)-9(b) denoted by the adjacency table are shown in Table 2.

(2) Adjacency matrix

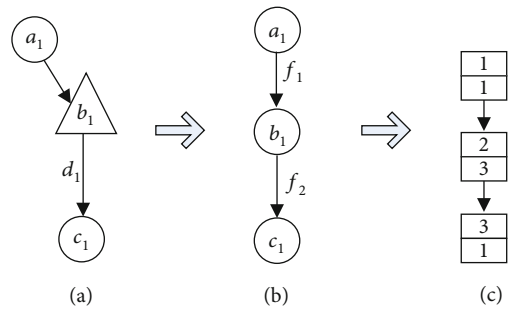


FIGURE 7: Model of subsystem 1 based on graph theory.

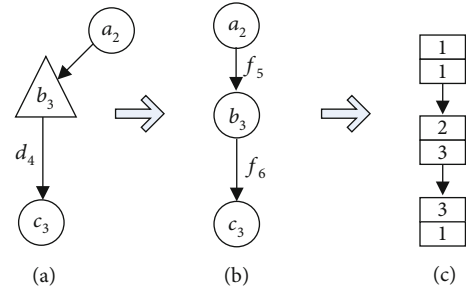


FIGURE 8: Model of subsystem 2 based on graph theory.

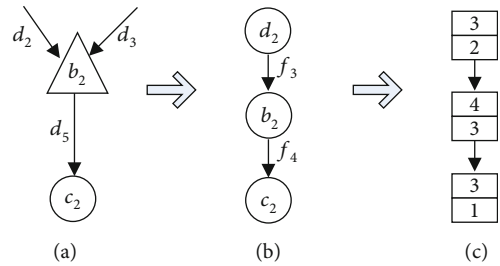


FIGURE 9: Model of subsystem 3 based on the graph theory.

Figure 7(b) is denoted by adjacency matrix A. Figure 8(b) is denoted by adjacency matrix B. Figure 9(b) is denoted by the adjacency matrix C. The adjacency matrices A, B, and C are as follows:

TABLE 2: Adjacency table of the system node graph.

Object	Subsystem 1		Subsystem 2		Subsystem 3		Whole system						
Line	$f_1$	$f_2$	$f_5$	$f_6$	$f_3$	$f_4$	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$
Upstream node	$a_1$	$b_1$	$a_2$	$b_3$	$d_2$	$b_2$	$a_1$	$b_1$	$b_1$	$b_2$	$a_2$	$b_3$	$b_3$
Downstream node	$b_1$	$c_1$	$b_3$	$c_3$	$b_2$	$c_2$	$b_1$	$c_1$	$b_2$	$c_2$	$b_3$	$c_3$	$b_2$

$$A = \begin{matrix} & a_1 & b_1 & c_1 \\ a_1 & \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \\ b_1 & \\ c_1 & \end{matrix}, B = \begin{matrix} & a_2 & b_3 & c_3 \\ a_2 & \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \\ b_3 & \\ c_3 & \end{matrix}, C = \begin{matrix} & d_2 & b_2 & c_2 \\ d_2 & \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \\ b_2 & \\ c_2 & \end{matrix}. \quad (9)$$

### (3) Correlation matrix

Figure 7(b) is denoted by the correlation matrix  $D$ . Figure 8(b) is denoted by correlation matrix  $E$ . Figure 9(b) is denoted by the correlation matrix  $F$ . The correlation matrices  $D$ ,  $E$ , and  $F$  are as follows:

$$D = \begin{matrix} & f_1 & f_2 \\ a_1 & \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 0 & -1 \end{bmatrix} \\ b_1 & \\ c_1 & \end{matrix}, E = \begin{matrix} & f_5 & f_6 \\ a_2 & \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 0 & -1 \end{bmatrix} \\ b_3 & \\ c_3 & \end{matrix}, F = \begin{matrix} & f_3 & f_4 \\ d_2 & \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 0 & -1 \end{bmatrix} \\ b_2 & \\ c_2 & \end{matrix}. \quad (10)$$

### 2.4.2. Typical Year Selection and Optimal Operation Criteria

(1) *Determination of the Typical Year.* It has to consider that the optimal operation model of the multireservoir in Kuitun River Basin should be in line with the current situation of water conservancy projects in the basin as far as possible. Therefore, the typical year is selected only in recent years. The selected results are as follows: three high flow years, 2015, 2016, and 2017; two normal flow years, 2013 and 2014; and one low flow year, 2009.

(2) *Optimal Operation Criteria.* The reservoir water supply in Kuitun River Basin is mainly used for irrigation in every irrigation district, the irrigation period is from April to November every year; April, May, and June are spring irrigation; July and August are summer irrigation; September, October, and November are autumn and winter irrigation.

- (1) Crop water requirement: based on the annual temperature, precipitation, and crop water requirement characteristics, in order to improve crop yield, the water supply of the basin in June, July, and August should be guaranteed to the greatest extent.

- (2) Reservoirs regulation and storage period: the water supply period of reservoirs is from April to November, and the water storage period of reservoirs is from December to March of the next year.

- (3) Starting and regulating capacity of reservoir operation: according to the starting and regulating capacity of reservoirs over the years, in combination with the water inflow from December to March of the next year to comprehensively consider, the Liugou reservoir starting and regulating capacity is  $9.324 \times 10^4 \text{ m}^3$ , the Kuitun river reservoir starting and regulating capacity is  $5.417 \times 10^4 \text{ m}^3$ , and the Huanggou reservoir starting and regulating capacity is  $3.549 \times 10^4 \text{ m}^3$ .

2.4.3. *Model Test.* The optimal operation calculation of the multireservoir in Kuitun River Basin can be carried out by using the adjacency table representation with the nonforward vertex-first topological ordering method. The main idea of this method is as follows: define a stack  $T$  to hold the sequence of nodes and select the most upstream node from the node adjacency table, as shown in Figure 7(a) with nodes  $b_1$  and  $a_1$ . The node  $p_i$  is put into the stack  $T$ , and all edges of  $p_i$  and  $p_i$  are deleted from  $G_{z_0}$ . The above selection and deletion are repeated until there are no nodes. Finally, stack  $T$  saves the topological sequence of subgraph  $G_{z_0}$ . When  $c_1$  is selected as the working node in Figure 7(b), the calculation sequence of nodes is  $a_1$ ,  $b_1$ , and  $c_1$ . When  $c_3$  is selected as the working node in Figure 8(b), the calculation sequence of nodes is  $a_2$ ,  $b_3$ , and  $c_3$ . When  $c_2$  is selected as the working node in Figure 9(b), the calculation sequence of nodes is  $d_2$ ,  $d_3$ ,  $b_2$ , and  $c_2$ .

According to the inflow type and outflow type of each node and the parameter settings of each node set in advance, the system chooses the corresponding calculation model for calculation. The subsystems feedback to each other, and the node sequence that meets the objective function is determined; finally, it is recorded in the water resources optimal operation database of the multireservoir in Kuitun River Basin in turn and output the optimal operation scheme. We take 2017 as an example, see Table 3.

## 3. Results

According to the principles of water diversion and irrigation district water supply, the annual water demand of each irrigation district can be maximized, and the water supply in June, July, and August can be guaranteed.

TABLE 3: Results of optimal operation of the multireservoir in Kuitun River Basin in 2017.

Node	4	5	6	7	8	9	10	11
$a_1$	1472.00	2513.00	5121.00	4684.00	6526.00	6024.00	1477.00	877.00
$b_1$	684.00	682.00	3590.65	12488.96	8903.94	999.00	48.00	35.00
$c_1$	684.00	842.00	2507.00	4336.00	3266.00	999.00	48.00	35.00
$a_2$	379.00	2126.00	3125.00	4928.00	4398.00	2600.00	509.00	420.00
$b_3$	1246.00	659.00	3125.00	4928.00	4398.00	2600.00	509.00	420.00
$c_3$	1246.00	499.00	2589.00	4110.00	3120.00	93.00	290.00	187.00
$b_2$	2073.32	1192.88	4798.33	9417.66	6915.94	746.05	580.72	61.94
$c_2$	2022.00	1140.00	4768.00	9385.00	6886.00	725.00	550.00	9.00

TABLE 4: Results of annual water shortage in each irrigation district in high flow years.

Month	L irrigation district			C irrigation district			H irrigation district		
	2015	2016	2017	2015	2016	2017	2015	2016	2017
Sp irrigation	4	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	5	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	6	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Su irrigation	7	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	8	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
A and W irrigation	9	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	10	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	11	0.01%	0.00%	0.00%	0.01%	0.00%	0.00%	0.01%	0.00%

TABLE 5: Minimum discharge of reservoirs in high flow years (unit:  $10^4 \text{ m}^3$ ).

Month		4	5	6	7	8	9	10	11
2015	L reservoir	866.42	827.88	9045.62	13393.90	9683.25	2382.52	5199.94	3285.30
	H reservoir	1448.32	1028.62	3706.40	5438.57	7324.12	331.61	2413.88	2108.88
	K reservoir	2244.90	1586.35	7066.42	9809.10	9077.79	1463.95	3926.88	3030.39
2016	L reservoir	1011.75	322.64	4926.43	6718.60	4646.96	1657.31	1360.37	1896.18
	H reservoir	1306.25	138.24	3721.20	3293.95	2590.14	537.23	1143.15	1731.93
	K reservoir	2290.47	313.28	3751.53	7132.96	5840.14	1807.62	1667.94	1512.79
2017	L reservoir	684.00	682.00	3590.65	12488.96	8903.94	999.00	48.00	35.00
	H reservoir	1246.00	659.00	3125.00	4928.00	4398.00	2600.00	509.00	420.00
	K reservoir	2073.32	1192.88	4798.33	9417.66	6915.94	746.05	580.72	61.94

TABLE 6: Minimum inflow of Kuitun reservoir in high flow years (unit:  $10^4 \text{ m}^3$ ).

Month	4	5	6	7	8	9	10	11
2015	0.00	0.00	6304.47	9809.10	9077.79	1463.95	3926.88	2994.95
2016	0.00	61.74	2367.93	3499.98	2534.30	558.28	1173.87	1784.87
2017	0.00	1903.00	2066.35	8970.96	6915.94	2507.00	219.00	233.00

### 3.1. Optimal Results of High Flow Year

3.1.1. Water Shortage Rate of Each Irrigation District after Optimization. After the optimal operation calculation of

the optimal model, the water shortage rate of each irrigation district decreases significantly in high flow years of 2015, 2016, and 2017. The specific results are shown in Table 4.



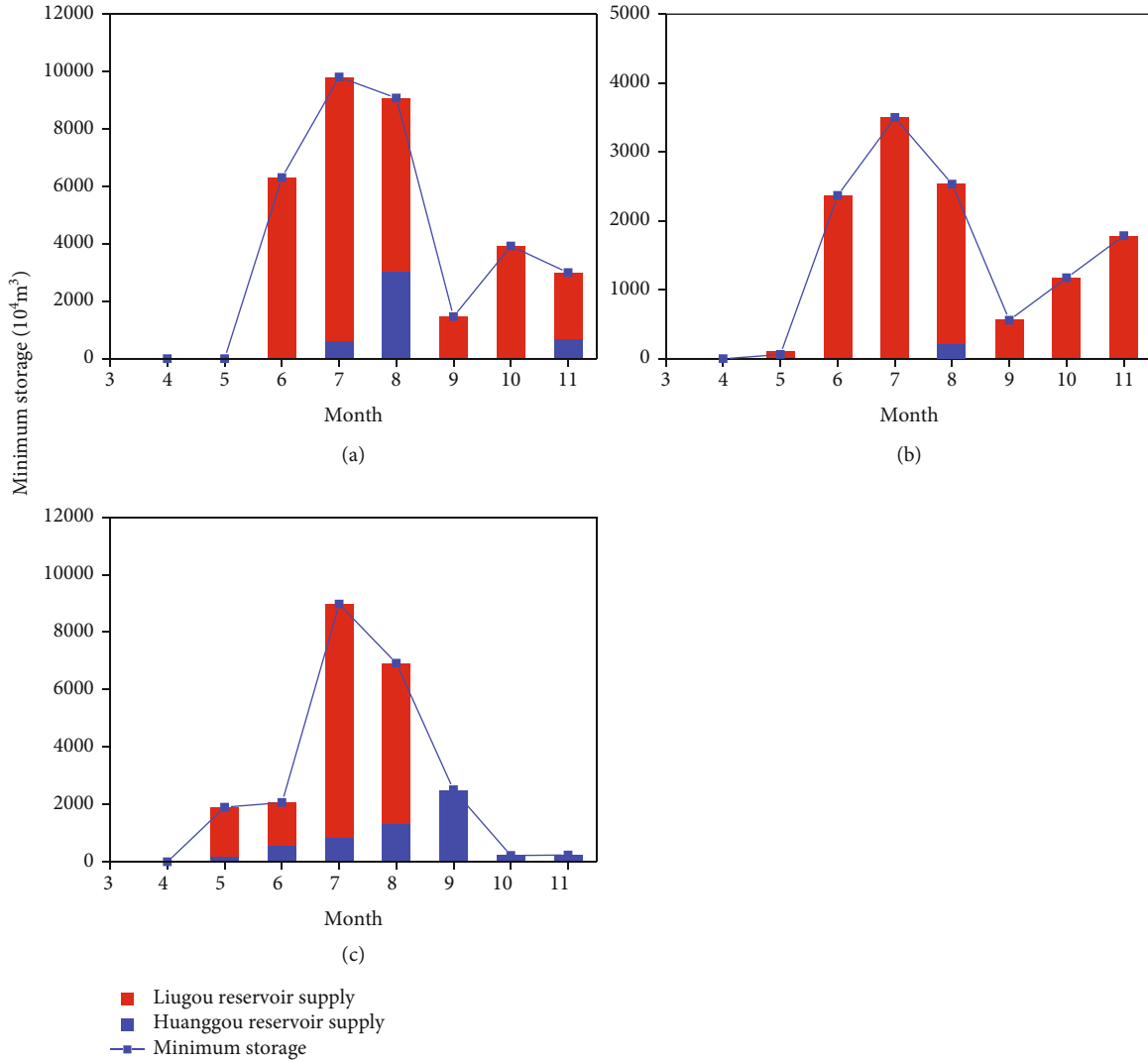


FIGURE 10: Minimum inflow and supply distribution diagram of Kuitun reservoir: (a) 2015, (b) 2016, and (c) 2017.

3.1.2. Regulation and Storage Process under the Upstream Reservoir Discharges according to the Minimum Discharge Flow (Taking Huanggou Reservoir as an Example)

- (1) Minimum discharge is shown in Table 5
- (2) The minimum inflow of Kuitun reservoir is shown in Table 6

It can be seen from the change of minimum inflow and supply distribution diagram of Kuitun reservoir in Figure 10, when Huanggou reservoir discharges according to the minimum discharge flow: The minimum inflow of Kuitun reservoir in high flow years of 2015, 2016, and 2017 is the largest in July, followed by August and the smallest in April. In July, August, and November 2015, Huanggou reservoir and Liugou reservoir jointly provide water supply for Kuitun reservoir. In August 2016, Huanggou reservoir independently provide water supply for Kuitun reservoir. From May to November 2017, Kuitun reservoir needed Huanggou reservoir and Liugou reservoir to jointly provide water supply.

- (3) Storage capacity change curve of the multireservoir operation

It can be seen from the storage capacity change curve of the multireservoir operation in Figure 11, when Huanggou reservoir supply water with minimum discharge: From July to August in high flow years of 2015, 2016, and 2017, the irrigation district has the largest water demand, and the regulation capacity of Liugou reservoir and Kuitun reservoir is basically close to the dead storage capacity; after September, the water supply decreases and the upstream reservoir begins to store water; Huanggou irrigation district only needs Huanggou reservoir for water supply, and the reservoir capacity is the lowest in August and November.

3.2. Optimal Results of Normal Flow Year

3.2.1. Water Shortage Rate of Each Irrigation District after Optimization. After the optimal operation calculation of the optimal model, the water shortage rate of each irrigation

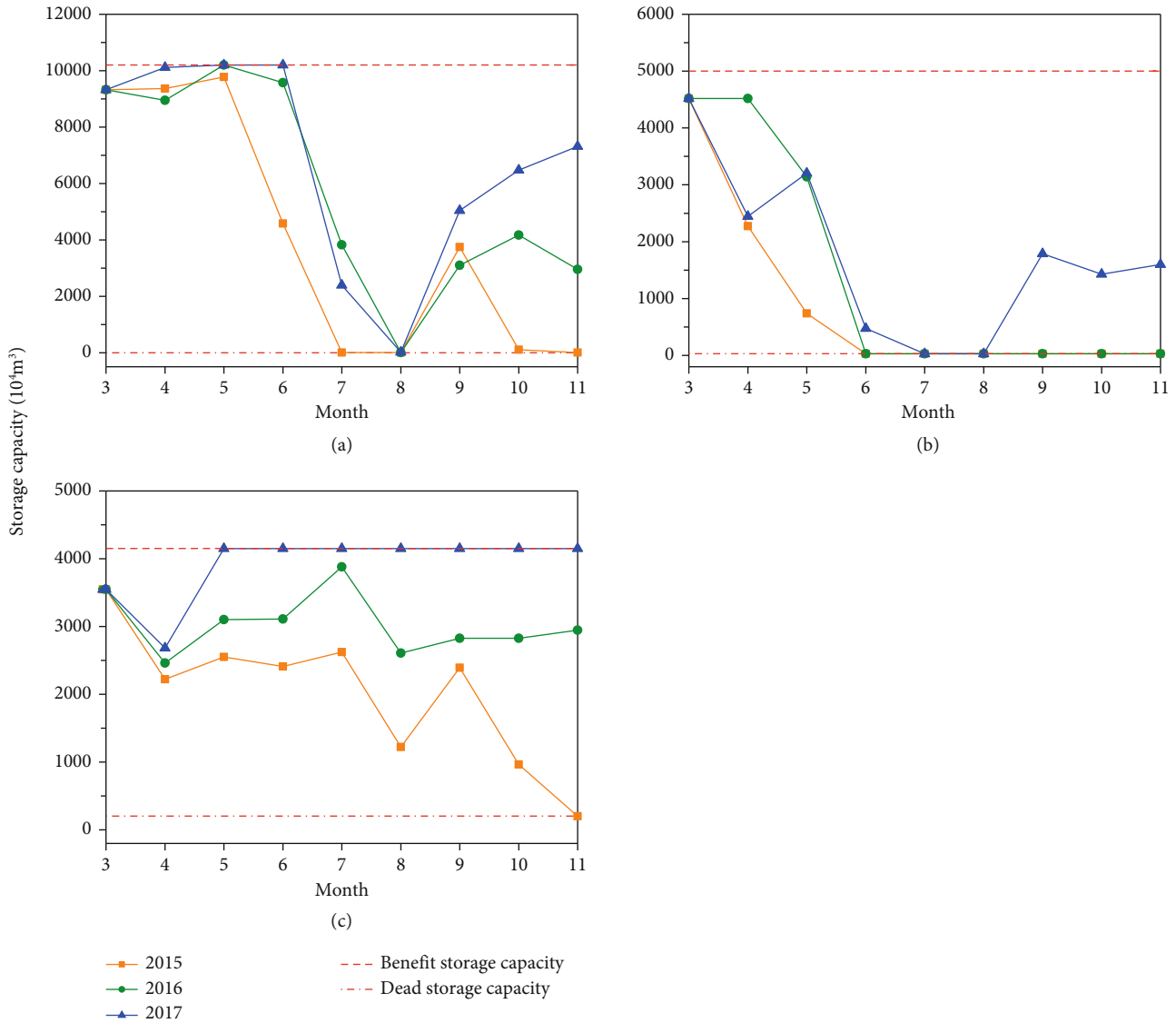


FIGURE 11: Storage capacity change curve of the multireservoir: (a) L reservoir, (b) K reservoir, and (c) H reservoir.

district decreases significantly in the normal flow years in 2013 and 2014. The specific results are shown in Table 7.

3.2.2. Regulation and Storage Process under the Upstream Reservoir Discharges according to the Minimum Discharge Flow (Taking Huanggou Reservoir as an Example)

- (1) Minimum discharge is shown in Table 8
- (2) The minimum inflow of Kuitun reservoir is shown in Table 9

It can be seen from the change of minimum inflow and supply distribution diagram of Kuitun reservoir in Figure 12, when Huanggou reservoir discharges according to the minimum discharge flow: In the normal flow year 2013, the minimum inflow of Kuitun reservoir is the largest in July, followed by September, slightly lower in August than in September, and the minimum is 0 in April and May. In

the normal flow year 2014, the minimum inflow of Kuitun reservoir is the largest in July, followed by August, and the minimum in April and May was 0. In the normal flow year 2013, Liugou reservoir independently provide water supply for Kuitun reservoir. In the normal flow year 2014, Huanggou reservoir and Liugou reservoir jointly provide water supply for Kuitun reservoir in August, September, and October.

- (3) Storage capacity change curve of the multireservoir operation

It can be seen from the storage capacity change curve of the multireservoir operation in Figure 13, when Huanggou reservoir supply water with minimum discharge: In July in the normal flow year 2013, Liugou reservoir, Kuitun reservoir, and Huanggou reservoir have the smallest reservoir capacity. In August in the normal flow year 2014, Liugou reservoir, Kuitun reservoir, and Huanggou reservoir have

TABLE 7: Results of annual water shortage in each irrigation district in normal flow years.

Month		L irrigation district		C irrigation district		H irrigation district	
		2013	2014	2013	2014	2013	2014
Sp irrigation	4	0.00%	2.50%	0.00%	2.50%	0.00%	2.50%
	5	0.00%	4.00%	0.00%	4.00%	0.00%	4.00%
	6	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Su irrigation	7	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	8	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
A and W irrigation	9	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	10	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	11	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

TABLE 8: Minimum discharge of reservoirs in normal flow years (unit:  $10^4 \text{ m}^3$ ).

Month		4	5	6	7	8	9	10	11
2013	L reservoir	848.17	572.47	2138.13	7941.54	6770.27	3871.19	3011.00	1985.00
	H reservoir	1415.18	375.77	2950.91	4296.61	3447.70	11.55	1042.19	675.22
	K reservoir	1393.24	437.06	3267.19	7861.53	6857.91	3653.17	3204.22	2911.99
2014	L reservoir	606.36	1018.41	2297.28	8569.55	4300.87	1515.62	759.08	1828.69
	H reservoir	1199.93	839.42	2660.37	4095.62	5637.73	1789.19	1711.28	796.28
	K reservoir	1402.95	957.37	2913.69	4607.40	3452.97	1204.62	1740.77	1237.68

TABLE 9: Minimum inflow of Kuitun reservoir in normal flow years (unit:  $10^4 \text{ m}^3$ ).

Month	4	5	6	7	8	9	10	11
2013	0.00	0.00	434.06	3748.85	3428.95	3591.70	2169.40	1249.91
2014	0.00	0.00	680.81	4607.40	3452.97	1204.62	1740.77	1237.68

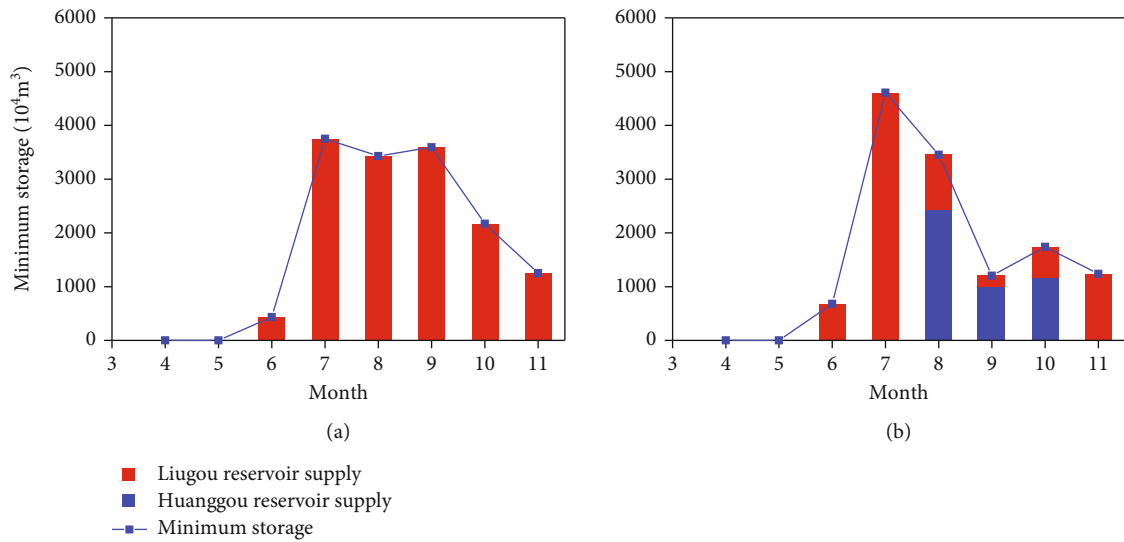


FIGURE 12: Minimum inflow and supply distribution diagram of the Kuitun reservoir: (a) 2013 and (b) 2014.

the smallest reservoir capacity; Liugou reservoir is close to dead storage capacity from July to November; and Kuitun reservoir is dead water level from June to November; The

storage capacity of Liugou reservoir, Kuitun reservoir, and Huanggou reservoir in November in normal flow years 2013 and 2014 are the maximum.

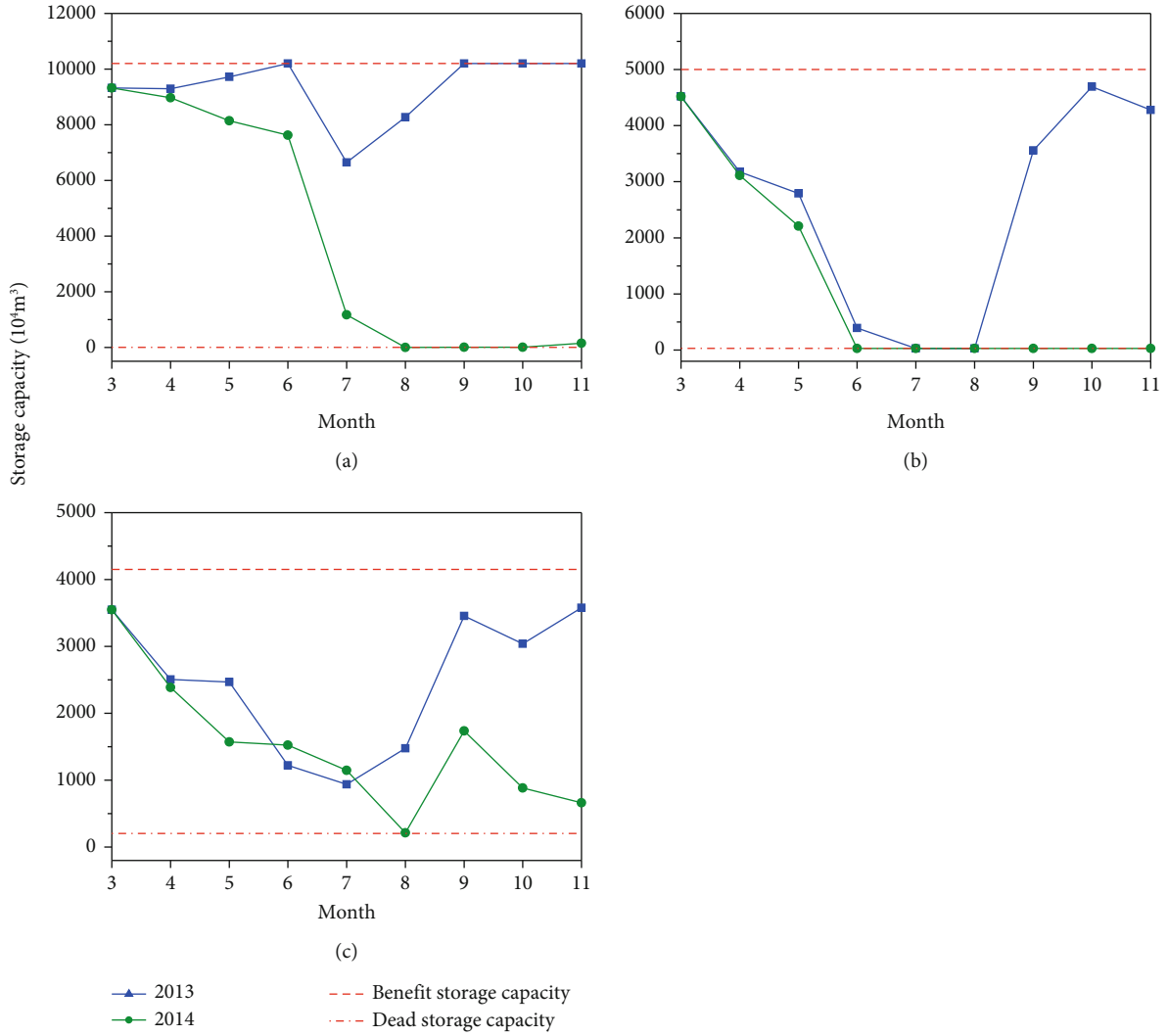


FIGURE 13: Storage capacity change curve of the multireservoir: (a) L reservoir, (b) K reservoir, and (c) H reservoir.

3.3. Optimal Results of Low Flow Year

3.3.1. Water Shortage Rate of Each Irrigation District after Optimization. After the optimal operation calculation of the optimal model, the water shortage rate of each irrigation district decreases significantly in the low flow year in 2009. The specific results are shown in Table 10.

3.3.2. Regulation and Storage Process under the Upstream Reservoir Discharges according to the Minimum Discharge Flow (Taking Huanggou Reservoir as an Example)

- (1) Minimum discharge is shown in Table 11
- (2) The minimum inflow of Kuitun reservoir is shown in Table 12
- (3) Storage capacity change curve of the multireservoir operation

It can be seen from the storage capacity change curve of the multireservoir operation in Figure 14, when Huanggou

TABLE 10: Results of annual water shortage in each irrigation district in the low flow year.

Month	L irrigation district	C irrigation district	H irrigation district
Sp irrigation	4	0.00%	58.50%
	5	0.00%	58.50%
	6	0.00%	40.00%
Su irrigation	7	0.00%	15.42%
	8	0.00%	0.00%
A and W irrigation	9	0.00%	1.00%
	10	0.00%	12.00%
	11	0.00%	1.50%

reservoir supply water with minimum discharge: Liugou reservoir has the lowest storage capacity in June and July, and the storage capacity rises at the end of August; Kuitun reservoir is close to the dead storage capacity at the end of May;

TABLE 11: Minimum discharge of reservoirs in the low flow year (unit:  $10^4 \text{ m}^3$ ).

Month	4	5	6	7	8	9	10	11
L reservoir	2376.50	3211.38	3582.78	3637.99	3635.27	1927.38	1937.05	1959.27
H reservoir	2027.69	2027.69	2113.40	2979.19	3522.33	1667.49	1482.21	1659.07
K reservoir	2686.82	3523.25	3992.44	4049.99	4044.54	2154.09	2173.43	2217.88

TABLE 12: Minimum inflow of Kuitun reservoir in the low flow year (unit:  $10^4 \text{ m}^3$ ).

Month	4	5	6	7	8	9	10	11
Minimum storage capacity	0.00	834.88	1969.78	2024.99	2022.27	1077.05	1086.72	1108.94

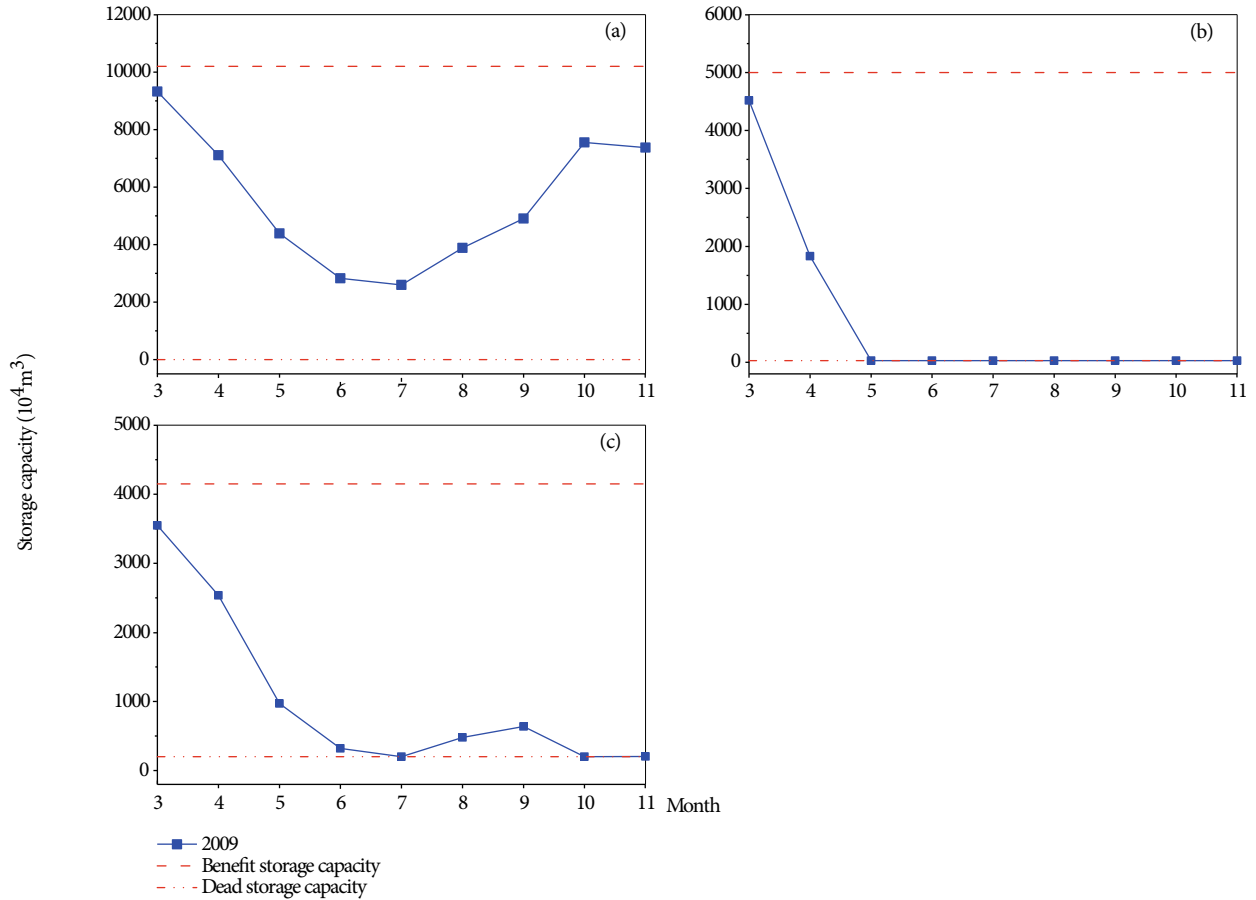


FIGURE 14: Storage capacity change curve of the multireservoir: (a) L reservoir, (b) K reservoir, and (c) H reservoir.

Huanggou reservoir is close to dead storage capacity at the end of June.

#### 4. Countermeasures and Suggestions

4.1. *Optimal Operation Scheme.* When Liugou reservoir and Huanggou reservoir supply water within the allowable water supply range of storage capacity in each period, the goal of minimum water shortage rate and minimum discharge of upstream reservoirs in Kuitun River Basin can be achieved (Figures 14–16). Therefore, the above optimization results can be obtained when the reservoir supply water within the allowable water supply range of storage capacity; otherwise,

the water shortage rate will be higher than the current results.

- (1) Reservoirs operation diagram in upstream in high flow year.
- (2) Reservoir operation diagram in upstream in normal flow year.
- (3) Reservoir operation diagram in upstream in low flow year.

Reservoirs operation diagram in upstream in the low flow year is seen in Figure 14 in Section 3.3.2.

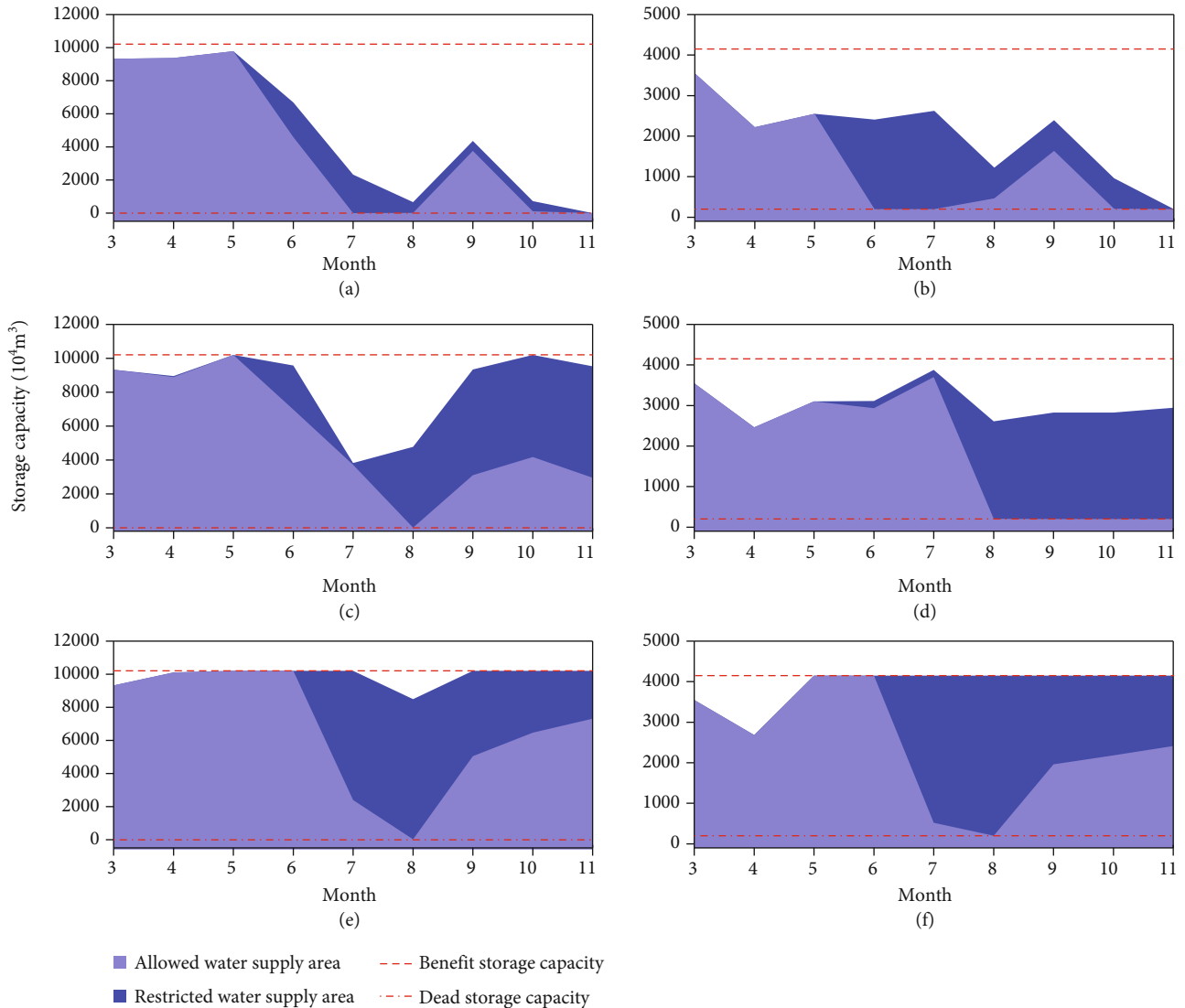


FIGURE 15: Reservoirs operation diagram in upstream in high flow years: (a) L reservoir in 2015, (b) H reservoir in 2015, (c) L reservoir in 2016, (d) H reservoir in 2016, (e) L reservoir in 2017, and (f) H reservoir in 2017.

4.2. Suggestions on Optimal Operation of the Multireservoir

4.2.1. Suggestions on Optimal Operation of Kuitun River Multireservoir.

At present, the multireservoir operation in Kuitun River Basin is mainly based on artificial experience. The time and space of water demand in each crop are not uniform. In most cases, for the crop that firstly applies for the water distribution, the actual water supply can not only meet the water demand plan but also even exceed the quota of water supply, which leads to the high water shortage rate of the crop that later applies for water distribution.

In view of the phenomenon of uneven distribution of water resources in time and space in operation of the multireservoir, the water distribution plan for the multireservoir operation in that year should be formulated in advance. The optimal operation of the multireservoir is simulated according to the predicted inflow of the year and the water use plan of each irrigation district, to alleviate the contradiction between water supply and demand in the basin caused

by the uneven distribution of water resources in time and space in operation of the multireservoir to the greatest extent.

4.2.2. Suggestions for the Independent Water Distribution District of Quangou Reservoir.

For many years, the water distribution of Quangou reservoir in Kuitun River Basin can only meet 40%~60% of its independent water allocation area. According to the reservoir optimal operation results in high flow years and normal flow years mentioned above, there are more water in the reservoir at the end of November. The following suggestions are made for less available water:

- (1) Improving the prediction accuracy of upstream inflow and increasing the water diversion from Quangou reservoir to the river channel according to the water use plan of each unit.



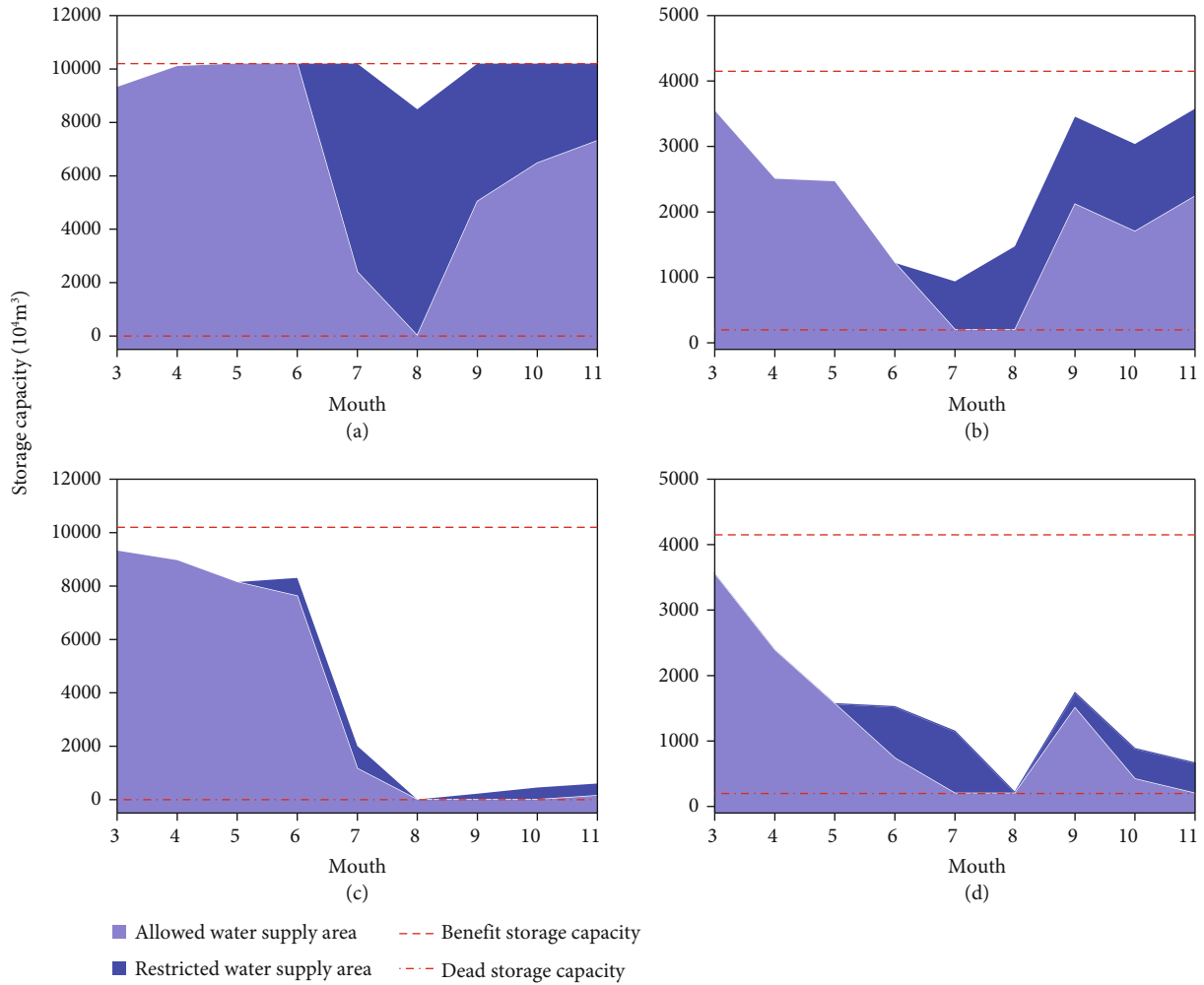


FIGURE 16: Reservoir operation diagram in upstream in normal flow years: (a) L reservoir in 2013, (b) H reservoir in 2013, (c) L reservoir in 2014, and (d) H reservoir in 2014.

- (2) Users in other upstream districts should consider establishing links with Huanggou reservoir, such as diverting water to Quangou reservoir and constructing channels and other measures.
- (3) Users in other downstream districts should consider establishing links with the Kuitun reservoir and Chepaizi reservoir, such as constructing channels to increase the available water supply from the Guertu river and the Sikeshu river to users in other downstream districts through the Liugou reservoir and the Kuitun reservoir.

**5. Conclusion**

Aiming at the problems that need to be solved urgently in the current operation of the multireservoir in Kuitun River Basin, such as the uneven distribution of water resources in time and space, the time cost of massive manual calculation, and low coordination level, a water resources optimal operation model of the multireservoir is established, and the actual data of typical years are selected to test the model.

The test results show that the water shortage rate of 2015 and 2016 in high flow years decreased by 98.57% and 100%, respectively, compared with the actual water distribution; the water shortage rate of 2013 and 2014 in normal flow years decreased by 92.65% and 96.38%, respectively, compared with the actual water distribution; and the water shortage rate of 2009 in the low flow year decreased by 87.78% compared with the actual water distribution.

**Abbreviations**

L reservoir:	Liugou reservoir
K reservoir:	Kuitun reservoir
C reservoir:	Chepaizi reservoir
H reservoir:	Huanggou reservoir
Q reservoir:	Quangou reservoir
S river:	Sikeshu river
G river:	Guertu river
K river:	Kuitun river
Sp irrigation:	Spring irrigation
Su irrigation:	Summer irrigation
A and W irrigation:	Autumn and winter irrigation

L irrigation district: Liugou irrigation district  
 C irrigation district: Chepaizi irrigation district  
 H irrigation district: Huanggou irrigation district.

### Data Availability

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

### Conflicts of Interest

The authors declare no competing interests.

### Authors' Contributions

Changlu Qiao made the formulation of overarching research goals and aims and establishment of the model. Yan Wang made solution method for the model and did data analysis and preparation of the original draft. Yanxue Liu did the data curation and writing of the initial draft. Junfeng Li performed the analysis with constructive discussions and did the review and editing of the manuscript. Heping Zhang and Jiangang Lu made the model test and diagramming.

### Acknowledgments

This work was supported by the National Natural Science Foundation of China (grant number 51769030) and Xinjiang Production and Construction Corps (grant number 2021AA003).

### References

- [1] O. Bozorg-Haddad, M. Mani, M. Aboutalebi, and H. A. Loáiciga, "Choosing an optimization method for water resources problems based on the features of their solution spaces," *Journal of Irrigation & Drainage Engineering*, vol. 144, no. 2, p. 04017061, 2018.
- [2] L. Ubertini, P. Manciola, and S. Casadei, "Evaluation of the minimum instream flow of the tiber river basin," *Environmental Monitoring & Assessment*, vol. 41, no. 2, pp. 125–136, 1996.
- [3] C. B. Stalnaker, "The instream flow incremental methodology: a primer for IFIM," *Instead Flow Incremental Methodology A Primer for Infirm*, vol. 29, 1995.
- [4] V. V. Kumar, B. V. Rao, and P. P. Mujumdar, "Optimal operation of a multibasin reservoir system," *Sadhana*, vol. 21, no. 4, pp. 487–502, 1996.
- [5] S. Ye and B. He, "Application research on optimal operation of multi-reservoir water supply based on particle swarm optimization algorithm," *2010 Second WRI Global Congress on Intelligent Systems*, 2010, Wuhan, China, Dec. 2010, 2010.
- [6] Q. Goor, C. Halleux, Y. Mohamed, and A. Tilmant, "Optimal operation of a multipurpose multireservoir system in the eastern Nile river basin," *Hydrology & Earth System Sciences Discussions*, vol. 14, no. 10, pp. 1895–1908, 2010.
- [7] Z. J. Yin, W. Huang, and J. Chen, "Issues on water quantity operation of large-sized reservoirs in the Yangtze River basin," *Journal of Yangtze River Scientific Research Institute*, vol. 28, no. 7, p. 7, 2011.
- [8] T. Bai, J. X. Chang, F. J. Chang, Q. Huang, Y. M. Wang, and G. S. Chen, "Synergistic gains from the multi-objective optimal operation of cascade reservoirs in the upper Yellow River basin," *Journal of Hydrology*, vol. 523, pp. 758–767, 2015.
- [9] Q. Li and S. Ouyang, "Research on multi-objective joint optimal flood control model for cascade reservoirs in river basin system," *Natural Hazards*, vol. 77, no. 3, pp. 2097–2115, 2015.
- [10] I. Thechamani, S. Visessri, and P. Jarumaneeroj, "Modeling of multi-reservoir systems operation in the Chao Phraya River basin," in *2017 International Conference on Industrial Engineering, Management Science and Application*, Seoul, Korea (South), June 2017.
- [11] M. I. Yekit, *Role of reservoir operation in sustainable water supply to Subak irrigation schemes in Yeh Ho River*, CRC Press, 2017.
- [12] Q. Wang, W. Ding, and Y. Wang, "Optimization of multi-reservoir operating rules for a water supply system," *Water Resources Management*, vol. 32, no. 14, pp. 4543–4559, 2018.
- [13] H. C. Zhou, Y. Peng, and G. L. Wang, "Research on integration in system of multi-reservoir flood forecast and operation based on graph theory," *Journal of Dalian University of Technology*, vol. 45, no. 6, p. 871, 2005.
- [14] Q. C. Wang Yan and L. Tingbo, "Coordination mechanism of water resources in Xinjiang region and Xinjiang production & construction corps," *Population, Resources and Environment in China*, vol. 2, pp. 170–173, 2017.
- [15] T. Dingwen, "Effects of different ecological base flows on power generation loss and loss of water conservancy hubs," *Water Conservancy Planning and Design*, vol. 7, pp. 25–27, 2016.
- [16] L. Shuzhen, "Rationality analysis and practice of ecological base flow discharge of small reservoirs in mountainous districts of Xinjiang in different periods," *Water Conservancy and Hydropower Technology*, vol. 47, no. 3, pp. 27–28, 2016.

## Research Article

# Energy-Efficient Data Transmission in Mobility-Aware Wireless Networks

Mengmeng Xu <sup>1</sup>, Guixiang Zhang <sup>2</sup>, Xinpei Liang <sup>1</sup>, Hai Zhu <sup>1</sup>  
and Juanjuan Wang <sup>1</sup>

<sup>1</sup>School of Network Engineering, Zhoukou Normal University, Zhoukou, Henan, China

<sup>2</sup>Zhengzhou Railway Vocational & Technical College, Zhengzhou, Henan, China

Correspondence should be addressed to Mengmeng Xu; [mmxu@zknw.edu.cn](mailto:mmxu@zknw.edu.cn)

Received 25 January 2022; Revised 5 March 2022; Accepted 16 May 2022; Published 24 May 2022

Academic Editor: Yuemin Ding

Copyright © 2022 Mengmeng Xu 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.

Data transmission scheme is an effective mode to improve the energy-efficiency in packet delivery. In this paper, we investigate the energy-efficient data transmission over reliable links and unreliable links in mobility-aware wireless networks. The network topologies in mobility-aware wireless networks are changed from one time slot to another, and they could be described by a sequence of static graphs. We first model these network topologies in a time period as a virtual space-time graph. On the virtual space-time graph, energy-efficient data transmission problems over reliable links and unreliable links are defined. The aim of the two data transmission problems is to find a spatial-temporal path with the minimum energy cost. Next, we propose an Energy-Efficient Data Transmission algorithm over Reliable Links (EEDT-RL) to find the optimal space-time path. Based on EEDT-RL, we also develop a heuristic data transmission protocol over unreliable links (named EEDT-UL), in which the path reliability is taken into consideration. Simulation results show that our proposed algorithms perform well in terms of energy cost and transmission count compared with some existing algorithms.

## 1. Introduction

Data transmission scheme is an effective manner to reduce the energy consumption of packet delivery in wireless networks. However, many intractable problems have been imposed in data transmission because of nodes' mobility, i.e., the network topologies are changed from one time slot to another, which is an inherent characteristic of wireless networks. Specially, the topology connectivity of the mobile network in each time-slot and even the existence of a routing path between two remote nodes could not be guaranteed. Opportunistic or epidemic transmission schemes are effective forwarding manners for packet transmissions between two remote nodes [1–4], in which data packets are forwarded by utilizing the sporadic contacts between two mobile nodes. However, packet transmissions using opportunistic or epidemic transmission schemes may result in plenty of copy transmissions. Actually, opportunistic or epidemic transmission schemes do not exploit the full potential

of the mobility. Although the problem of the absence of routing path could be alleviated through opportunistic or epidemic transmission schemes, how to select the relay nodes, i.e., making forwarding decision over time-varying network topologies is still an intractable problem in mobility-aware wireless networks. Therefore, it is imperative to investigate the data transmission problem in mobility-aware networks.

In the past decades, energy-efficient data transmission protocols over reliable and/or unreliable links in static networks have been widely investigated, as shown in [5–8]. However, these network protocols are not suitable for the applications in mobile wireless networks due to the frequent link breakage and rebuilding. In mobile wireless networks, most of the researches for routing design have concentrated on the stable and reliable path selection which may have a long duration [9–12]. In these research works, node mobility is assumed in terms of some classical random mobility models, such as random direction model and random way-

point model. Due to the high randomness, these researches have overmuch emphasis on the path duration in the packet delivery but neglect the importance of another factor—energy consumption.

In the human-centric mobile networks, the temporal characteristics of network topology in many network scenarios could be known a priori. Actually, in [13], the authors show that the potential predictability for human mobility can reach up to 93%. The literature [14] found that about 78%-99% of the vehicle location is predicable. Thanks to the development of cloud computing technique, the authors in [15, 16] investigated the mobility prediction in bike-sharing systems and in public bus systems, respectively. For perpetual trajectory tracking, the authors in [17] propose an energy and mobility-aware scheduling framework to improve the long-term tracking performance. All of these researches validate that the strong regularities are existed in the daily human and vehicle mobility. In these mobility-aware wireless networks, the literatures [18–20] proposed some mobility-aware energy-efficient data transmission schemes. However, these proposed algorithms are based on the connect network topology, and the temporal information of topology change from one time slot to another is not involved explicitly. By exploiting vehicle mobility trajectories, the authors in [21] developed an efficient multicast algorithm in wireless vehicular networks. Using machine learning for mobility prediction, the literature [22] proposed a centralized routing scheme to minimize the overall vehicular service delay. Some other mobility-aware-based network protocols could be found in [23, 24] for low-cost topology control, in [25] for location management, in [26] for clustering, and in [27] for resource allocation.

In this paper, we investigate the energy-efficient data transmission problems in mobility-aware wireless networks over reliable links and unreliable links. A series of network topologies, each of which is disconnected with high probability, is modeled as a virtual space-time graph. In this space-time topology graph, a spatial link refers as to a wireless link between two nodes at one time slot; while a temporal link means that a node carries its data packet from one time slot to the next time slot. Data transmission problems over reliable links and unreliable links on the virtual space-time graph are defined, in which the space-time path with minimum energy cost needs to be found. An Energy-Efficient Data Transmission algorithm over Reliable Links (EEDT-RL) is proposed to solve the data transmission problem over reliable links. By extending the EEDT-RL, we also develop a heuristic transmission algorithm over unreliable links, i.e., EEDT-UL. Some simulations are performed to study the performances of our proposed algorithms. Compared with our conference paper in [28], many new research results are added, such as the optimal data transmission algorithm over reliable links, more elegant optimal objective over unreliable links, and more simulation results.

The remainder of this paper is organized as follows. We review some of the related works in Section 2. Section 3 presents the network model, link model, and energy model. Energy-efficient data transmission over reliable links and unreliable links are investigated in Sections 4 and 5, respec-

tively. We provide some simulations to illustrate our algorithm in Section 6 and conclude the paper in Section 7.

## 2. Related Works

In this section, we summarize some up to date research works related with routing protocol or data transmission in mobility wireless networks.

In delay-tolerant mobile networks, the authors in [1] provided a detailed survey of opportunistic transmission algorithms to study the nature of mobility. Specially, it revealed that human mobility is not random at all but have a definite and repetitive pattern. Actually, using opportunistic transmission, a single packet transmission may result in plenty of copies of the packet message. The authors in [29, 30] focused on the controlling two-hop forwarding policies, where the problem concerned the decision on whether or not forwarding a given packet to a specific mobile node. By restricting the number of transmission hops, the proposed algorithms in [29, 30] could markedly reduce the copies in packet dissemination. In UAV-assisted vehicular delay-tolerant networks, the literature [31] developed a routing protocol to improve the reliability of packet transmission by considering both the encounter probability and the persistent connection time. In [32], theoretical upper and lower bounds for the information propagation speed were derived, in which store-carry-forward routing model was used. Considering wireless transmissions and sojourns on node buffers, the authors in [33] computed the packet speed and cost according to utility-based routing rules. However, most of the research results obtained in delay-tolerant mobile networks are based on the assumption of random mobility and do not exploit the full potential of the mobility traces.

In mobility-aware networks, a novel graph metric named mobile conductance was conceived to evaluate the information spreading time in [34]. The mobility-connectivity trade-off was also quantitatively analyzed in [34] to determine how much mobility may be exploited to compensate for network connectivity deficiency. Bedogni et al. in [35] developed a methodology to infer complete trajectories of individual vehicles and then proposed some temporal connectivity algorithms for packet transmissions. In [36], the authors introduced a concept of energy-aware temporal reachability graph (ETRG) and proposed an algorithm to calculate ETRG. According to ETRG, these results revealed the fundamental relations among the system metrics of energy budget, tolerable delay, and data size on the network performance. In [23], a reliable topology design was investigated in delay-tolerant networks with unreliable links. According to the known or predictable network topology, several heuristic algorithms were proposed to build reliable and low-cost network topologies. Nevertheless, the energy-efficient data transmission protocol does not involve explicitly in these works.

For data transmission or routing design in mobility-aware networks, the authors in [20] developed a relay selection strategy by utilizing partially predictable mobility, in which the directional correlation of destination movement was considered. Simulations showed that the proposed

forwarding strategy could achieve a higher delivery utility compared with a forwarding scheme without mobility prediction. In [21], a trajectory-based multicast (TMC) routing was proposed for efficient multicast in vehicular networks. By exploiting vehicle trajectories, TMC routing could achieve a delivery ratio close to that of the flooding-based approach while the cost is reduced by over 80%. Li et al. in [37] concentrated on the communication services of passengers in the train from the base station. According to the regular mobility of the train, the authors in [37] proposed a quality-of-service-distinguished power allocation algorithm to meet each user's data rate requirement. In [38], a novel social-based routing approach was proposed, in which a new metric of social energy was introduced by exploiting social behaviors of nodes. Liu et al. in [39] proposed a mobility-aware transmission scheduling scheme, which consists of a relay path planning algorithm and a global time scheduling algorithm. Extensive simulations under realistic human mobility trajectories showed that the transmission scheduling scheme in [39] could achieve high throughput transmission. An aeronautical ad hoc network with rapidly changing topology was modeled as a dynamic graph in [40], and then data transmission problem was formulated as an integer nonlinear programming. A detailed review work is available in the literature [41]. Different from these works, this paper investigates the energy-efficient data transmission problem in mobility-aware wireless networks. Specially, a series of dynamic network topologies is modeled as a virtual space-time graph, which is similar to the literature [23]. The space-time graph model is also utilized in wireless duty-cycle sensor networks for data transmission, as shown in [42, 43]. In our new data transmission problems, both reliable and unreliable links are involved.

### 3. System Models

In this section, we present the network model, link model, and energy consumption model. A description of the key notations is listed in Table 1.

**3.1. Network Model.** In mobile wireless networks, the locations of network nodes are changed over time, resulting in topology evolutions. To describe such evolution, a sequence of static graphs is introduced. Let a period of time  $T$  be divided into discrete and equal time slots, i.e.,  $\{1, 2, \dots, T\}$ . Define the network topology at time-slot  $t$  as an undirected graph  $G^t(V, L^t)$ , where  $V = \{1, 2, \dots, n\}$  is the set of nodes, and  $(i, j) \in L^t$  represents the wireless link between nodes  $i$  and  $j$  at time-slot  $t$ . The dynamic network over a period of time  $T$  could be modeled as a sequence of static graphs  $\{G^t | t = 1, 2, \dots, T\}$ , which describes the topology evolutions due to node mobility. Figure 1(a) provides an example to show the sequence of snapshots of the network topology at each time slot. Note that the topology connectivity at each time slot in mobile wireless networks could not be guaranteed, which incurs the data transmission over them intractable. Here, we assume that the moving track of each node (such as the smart device in public bus) is known in advance, i.e., the topology graph  $G^t$  at time-slot  $t$  is predictable.

TABLE 1: List of key notations.

Notation	Description
$T$	Time period
$G^t$	Network topology at time-slot $t$
$(i, j)$	Wireless link between nodes $i$ and $j$
$\mathcal{G}$	Space-time graph
$\overleftarrow{(i(t-1), j(t))}$	Spatial link
$\overleftarrow{(i(t-1), i(t))}$	Temporal link
$p(i, j)$	Reliability probability for link $(i, j)$
$p(\pi)$	Reliability of path $\pi$
$N$	Number of bits per message
$E_{rx}, e_{tx}, \beta$	Energy consumption parameters
$\alpha$	Path attenuation factor

We introduce a new graph structure associated with the sequence of network topologies  $\{G^t | t = 1, 2, \dots, T\}$ , named virtual space-time graph  $\mathcal{G}(\mathcal{V}, \mathcal{L})$ . In this virtual graph  $\mathcal{G}$ , the virtual node  $i(t)$  is the virtualization of node  $i \in V$  at the end of time slot  $t \in \{0, 1, \dots, T\}$ . As a consequence, each node  $i \in V$  is replaced with  $T + 1$  associated virtual nodes. The number of nodes in virtual graph  $\mathcal{G}$  is equal to  $n(T + 1)$ , i.e.,  $|\mathcal{V}| = n(T + 1)$ . If  $(i, j) \in L^t$ , ( $t = 1, \dots, T$ ), two directed edges (or links) are generated in  $\mathcal{G}$ , that is,  $\overleftarrow{(i(t-1), j(t))} \in \mathcal{L}$  and  $\overleftarrow{(j(t-1), i(t))} \in \mathcal{L}$ . This kind of link  $\overleftarrow{(i(t-1), j(t))}$  refers as to a spatial link, meaning that one node  $i$  can forward a message to node  $j$  at time-slot  $t$ . And besides, another kind of link  $\overleftarrow{(i(t-1), i(t))}$ , referred as to a temporal link, is defined in the link set  $\mathcal{L}$ , representing that the node  $i \in V$  can carry the message in the  $t$ -th time slot. The virtual space-time graph  $\mathcal{G}$  clearly includes both the spatial information in each time slot and the temporal information due to topology change. Figure 1(b) illustrates the virtual space-time graph of the sequence of topologies in Figure 1(a). In this example, the red path in Figure 1(b) is a space-time path from 4 (0) to 1 (3). That is, the message transmission from node 4 to node 1 needs 3 time slots: node 4 holds its packet at  $t = 1$  and sends it to node 3 at  $t = 2$ , and then node 3 sends it to node 1 at  $t = 3$ . Note that at most one message could be transmitted within one time slot. That is, we assume that the time slot is only long enough for one message transmission.

**3.2. Link Model.** A wireless link  $(i, j)$  exists at time-slot  $t$ , i.e.,  $(i, j) \in L^t$ , if and only if the two mobile nodes  $i$  and  $j$  are located within the transmission range of each other at time-slot  $t$ . We assume that a mobile node  $i$  always fails to transmit its packet to another node  $j$  which is beyond node  $i$ 's transmission range, while packet delivery within each other transmission range succeeds with a probability. We define a reliability probability  $p(i, j)$  for each link  $(i, j) \in L^t$ , which means node  $j$  can successfully receive a packet sent from node  $i$ . The values of reliability probability are influenced by the stochastic nature of wireless channel and/or



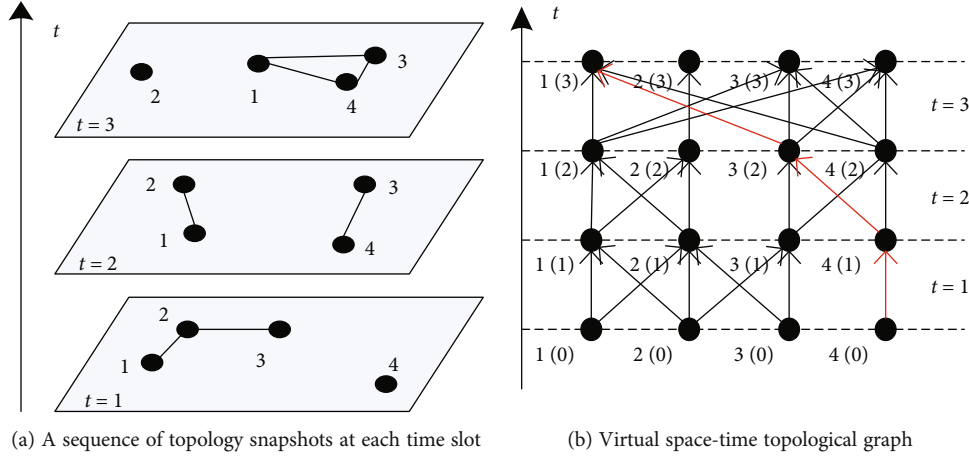


FIGURE 1: An example of network model.

imperfect mobility prediction. According to the value of  $p(i, j)$ , we define two link models as follows, i.e., reliable link model and unreliable link model.

In the reliable link model, if a wireless link is  $(i, j) \in L^t$ , we have  $p(i, j) = 1$ ; otherwise,  $p(i, j) = 0$ . Under this model, data packets transmitted within the transmission range are always succeeded.

In the unreliable link model, a wireless link  $(i, j) \in L^t$  connecting two nodes  $i$  and  $j$  at time-slot  $t$  is not necessarily reliable. That is, if a wireless link is  $(i, j) \in L^t$ , we have  $0 < p(i, j) \leq 1$ ; otherwise,  $p(i, j) = 0$ . Under this model, data packets were transmitted from node  $i$  to node  $j$  over a wireless link  $(i, j)$  with a successful reception probability  $p(i, j)$ .

A wireless link  $(i, j) \in L^t$  associates with two spatial links, i.e.,  $\overleftarrow{(i(t-1), j(t))} \in \mathcal{L}$  and  $\overrightarrow{(j(t-1), i(t))} \in \mathcal{L}$ , in space-time graph  $\mathcal{G}$ . Therefore, the reliability probability  $p(i, j)$  of wireless link  $(i, j) \in L^t$  could be converted into  $p(\overleftarrow{(i(t-1), j(t))})$  and  $p(\overrightarrow{(j(t-1), i(t))})$ , and we have

$$p(\overleftarrow{(i(t-1), j(t))}) = p(\overrightarrow{(j(t-1), i(t))}) = p(i, j). \quad (1)$$

For each temporal link  $\overleftarrow{(i(t-1), i(t))}$  in space-time graph  $\mathcal{G}$ , it may also have a reliability probability for successfully holding its data packet over one time slot. Actually, if buffer overflow or energy depletion incurs at a mobile node, the node may fail to hold its data packet. However, the failures in holding a packet over one time slot are seldom compared with those in data transmission over spatial link. Therefore, it is reasonable to assume that all temporal links are reliable, i.e.,  $p(\overleftarrow{(i(t-1), i(t))}) = 1$  for each  $i \in V$ ,  $t \in \{0, 1, \dots, T\}$ . We define the reliability of a space-time path  $\pi$  as the production of all links' reliability in  $\pi$ , i.e.,  $p(\pi) = \prod_{l \in \pi} p(l)$ .

**3.3. Energy Consumption Model.** Both sending a message over a spatial link and holding a message over a temporal link incur energy consumption. The energy cost for holding

one bit of data message in one time slot for any node is assumed to be a fixed value  $E_0$ , i.e., for any temporal link  $l \in \mathcal{L}$ , the corresponding energy cost for holding a message is  $E(l) = NE_0$ , where  $N$  is the number of bits per message. The energy consumption for sending a message over a spatial link comes from two parts: transmission and reception. Let  $E_{rx}$  be the amount of energy consumption required to receive one bit of data message. The amount of energy consumption for transmitting one bit of data message to  $r$  meters away is  $E_{tx}(r)$ . From [44], we have

$$E_{tx}(r) = e_{tx} + \beta r^\alpha, \quad (2)$$

where  $e_{tx}$  is the energy consumed by the sender circuit,  $\beta$  is the antenna output energy to reach the receiver unit distance away, and  $\alpha \in [2, 4]$  is the path attenuation factor. We define the energy cost for sending a message over a spatial link  $l = \overleftarrow{(i(t-1), j(t))} \in \mathcal{L}$  as

$$E(l) = N(E_{rx} + E_{tx}(d_{ij}))e^{-\rho t}, \quad (3)$$

where  $d_{ij}$  is the distance between nodes  $i$  and  $j$  at time-slot  $t$ , and  $\rho$  is the discount rate in time. The energy cost of a space-time path  $\pi$  is the summation of all links' energy cost in  $\pi$ , i.e.,  $E(\pi) = \sum_{l \in \pi} E(l)$ .

In next two sections, we firstly define the data transmission problems over reliable links and unreliable links in space-time graph and then develop two algorithms to solve the proposed transmission problems.

#### 4. Energy-Efficient Data Transmission over Reliable Links

We now define the energy-efficient data transmission problem over reliable links on the virtual space-time graph  $\mathcal{G}(\mathcal{V}, \mathcal{L})$ .



*Definition 1.* Given a source node  $s \in V$ , a destination node  $d \in V$ , the aim of energy-efficient data transmission over reliable links is to find a space-time path  $\pi$  with the minimum energy cost from node  $s(0) \in \mathcal{V}$  to node  $d(t) \in \mathcal{V}$ , ( $0 < t \leq T$ ).

Let the space-time graph  $\mathcal{G}$  over a period of time  $T$  be connected, such that the space-time path  $\pi$  from source node  $s$  to destination node  $d$  could be found over the time period  $T$ . Here, a space-time graph  $\mathcal{G}$  is connected over time period  $T$  if and only if there exists at least one space-time path for each pair of nodes  $(v_i^0, v_j^T)(i, j \in V)$ .

Note that, if the time period  $T$  is small, the number of selectable space-time paths is less. For example, in Figure 1(b), if  $T$  is equal to 2 time slots, the space-time path between source node  $s = 4$  and destination node  $d = 3$  is just one, i.e.,  $4(0) \rightarrow 4(1) \rightarrow 3(2)$ . If  $T$  is equal to 3 time slots, another selectable path between  $s = 4$  and  $d = 3$  is added, i.e.,  $4(0) \rightarrow 4(1) \rightarrow 4(2) \rightarrow 3(3)$ . With the increasing of time period  $T$ , the number of selectable paths is nondecreasing, and then the optimal space-time path which has the minimum energy cost should be found. Therefore, the energy-efficient data transmission problem in a time period  $T$  can also be viewed as a tradeoff between energy consumption and transmission delay.

In the data transmission problem over reliable links on the virtual space-time graph  $\mathcal{G}(\mathcal{V}, \mathcal{L})$ , all virtual nodes  $i(0), i(1), \dots, i(T)$  in the space-time graph  $\mathcal{G}$  associate with one node  $i \in V$  but at different moment. Therefore, each space-time path  $\pi$  in  $\mathcal{G}$  that connects node  $s(0)$  and anyone node  $d(t)$ , ( $0 < t \leq T$ ) constitutes the candidate path set  $\Pi$ . That is, the source node  $s$  could transmit its data packet to destination node  $d$  at time-slot  $t$  across the space-time path  $\pi$ . The solution of data transmission problem over reliable links is to find the space-time path  $\pi^*$  in  $\Pi$  which has the minimum energy cost. Next, Energy-Efficient Data Transmission algorithm over Reliable Links (EEDT-RL) is developed to solve the new data transmission problem. Algorithm 1 shows the detailed procedure of EEDT-RL.

In EEDT-RL, we firstly convert the sequence of static network graphs  $\{G^t | t = 1, 2, \dots, T\}$  into a space-time graph  $\mathcal{G} = (\mathcal{V}, \mathcal{L})$ . Then, on the space-time graph  $\mathcal{G}$ , an extension of Dijkstra's algorithm is implemented to find the space-time path with minimum energy cost from node  $s(0)$  to  $d(T)$ . In EEDT-RL,  $\mathcal{R}(i(t))$  is comprised of the links of the optimal space-time path from source node  $s(0)$  to node  $i(t)$  with the minimum energy cost  $\mathcal{E}(i(t))$ . In our assumption, the space-time graph  $\mathcal{G}$  over a period of time  $T$  is connected. Thus, the space-time path  $\pi$  from node  $s(0)$  to node  $d(T)$  could be found over the time period  $T$ , which means  $\mathcal{E}(d(T)) < \infty$ . Finally, we find the minimum value of  $\mathcal{E}(d(1)), \dots, \mathcal{E}(d(T))$  and denote it by  $\mathcal{E}(d(t))$ . Then, the optimal space-time path from source node  $s$  to destination node  $d$  over the time period  $T$  is  $\mathcal{R}(d(t))$  with the minimum energy consumption  $\mathcal{E}(d(t))$ . Since the space-time graph  $\mathcal{G}$  has  $n(T+1)$  virtual nodes, the computational complexity of EEDT-RL is  $O(n^2((T+1)^2)) = O(n^2T^2)$  in the worst case. Note that the link  $l \in \mathcal{L}$  is a directed link, and the arrow is omitted in the EEDT-RL algorithm.

## 5. Energy-Efficient Data Transmission over Unreliable Links

In this section, energy-efficient data transmission problem over unreliable links is defined on the virtual space-time graph  $\mathcal{G}(\mathcal{V}, \mathcal{L})$ .

*Definition 2.* Given a source node  $s \in V$ , a destination node  $d \in V$ , the aim of energy-efficient data transmission over unreliable links is to find a space-time path  $\pi$  with the minimum energy cost and the maximum path reliability from node  $s(0) \in \mathcal{V}$  to node  $d(t) \in \mathcal{V}$ , ( $0 < t \leq T$ ).

The energy-efficient data transmission problem over unreliable links has two optimal objectives: the minimum energy cost

$$E(\pi^*) = \sum_{l \in \pi^*} E(l) = \min \{E(\pi) | \pi \in \Pi\} \quad (4)$$

and the maximum path reliability

$$p(\pi^*) = \prod_{l \in \pi^*} p(l) = \max \{p(\pi) | \pi \in \Pi\}. \quad (5)$$

Here,  $\Pi$  is the candidate path set in which each space-time path connects node  $s(0)$  and anyone node  $d(t)$ , ( $0 < t \leq T$ ). To solve the biobjective optimization problem, the second objective is rewrote as

$$-\ln \left( \prod_{l \in \pi^*} p(l) \right) = \sum_{l \in \pi^*} (-\ln(p(l))) = \min \{-\ln(p(\pi)) | \pi \in \Pi\}. \quad (6)$$

By following a popular approach used to deal with biobjective optimization problems, the two objectives (4) and (6) have been transformed into a single objective, using an importance weight factor  $\lambda$  [45]. Then, define a composite link weight for each link  $l \in \mathcal{L}$  as

$$w(l) = \frac{\lambda}{E_{\max}} E(l) - \frac{1 - \lambda}{\ln(p_{\min})} \ln(p(l)), \quad (7)$$

where  $E_{\max}$  and  $p_{\min}$  are the maximum of  $E(l)$  and the minimum of  $p(l)$  for  $l \in \mathcal{L}$ , respectively. Note that  $E_{\max}$  and  $p_{\min}$  are normalization factors used to have the same range for the two objectives.

Based on minimizing  $w(\pi) = \sum_{l \in \pi} w(l)$  on the space-time graph  $\mathcal{G}$ , an extension version of Dijkstra's algorithm could be employed, which is similar to Algorithm 1. If the optimal space-time path  $\mathcal{R}(d(t))$  with minimum composite weight is found, the corresponding energy cost and path reliability could be further calculated. It should be noted that, because of the link unreliability, if the data packet from source node  $s$  is failed to be forwarded to the next node by one intermediate node  $j$  at time-slot  $t$ , this intermediate node  $j$  will become the source node at time-slot  $t+1$  and find the optimal space-time path again in the remaining time-slots. We call this

```

Input:  $G^t, s, d, T$ ;
Output:  $\pi^*, \mathcal{E}_{\min}$ ;
1: Compute space-time graph  $\mathcal{G} = (\mathcal{V}, \mathcal{L})$  According to a series of static graph  $G^t$  and time period  $T$ ;
2: for each node  $m(t) \in \mathcal{V}$  do
3:    $\mathcal{R}(m(t)) = \emptyset$  and  $\mathcal{E}(m(t)) = \infty$ 
4: end for
5:  $k = s(0)$ ,  $\mathcal{E}(k) = 0$ , and  $\mathcal{N} = \{k\}$ 
6: while  $k \neq d(T)$  do
7:    $temp = \infty$ ;
8:   for each node  $i(t) \in \mathcal{V} - \mathcal{N}$  do
9:     if  $l \triangleq (k, i(t)) \in \mathcal{L}$  then
10:      Calculate  $E(l)$  according to the Subsection 3-C;
11:     else
12:        $E(l) = \infty$ ;
13:     end if
14:     if  $\mathcal{E}(k) + E(l) \leq \mathcal{E}(i(t))$  then
15:        $\mathcal{R}(i(t)) = \mathcal{R}(k) \cup l$ ;
16:        $\mathcal{E}(i(t)) = \mathcal{E}(k) + E(l)$ ;
17:     end if
18:     if  $\mathcal{E}(i(t)) < temp$  then
19:        $temp = \mathcal{E}(i(t))$  and  $k' = i(t)$ ;
20:     end if
21:   end for
22:    $k = k'$  and  $\mathcal{N} = \mathcal{N} \cup \{k\}$ 
23: end while
24: Let  $\mathcal{E}(d(t))$  be the minimum of  $\mathcal{E}(d(1)), \dots, \mathcal{E}(d(T))$ 
25: return  $\pi^* = \mathcal{R}(d(t))$ ,  $\mathcal{E}_{\min} = \mathcal{E}(d(t))$ 

```

ALGORITHM 1: EEDT-RL.

TABLE 2: Values of various parameters in simulations.

Parameter	Value
Number of nodes ( $n$ )	10 ~ 30
Time period ( $T$ )	10 time slots
Energy for holding a message ( $E_0$ )	5 nJ/bit
Energy for receiver circuit ( $E_{rx}$ )	180 nJ/bit
Energy for transmitter circuit ( $e_{tx}$ )	80 nJ/bit
Path attenuation factor ( $\alpha$ )	2
Antenna output energy ( $\beta$ )	100 pJ/bit
Number of bits per message $N$	256 bytes
Maximum transmission range $r_{\max}$	150 m
Time discount rate $\rho$	0

heuristic data transmission method over unreliable links as EEDT-UL. In the worst case, each transmission has failed, and then the path finding algorithm needs to be executed  $T$  times. Therefore, the computational complexity of EEDT-UL is  $O(n^2 T^3)$ .

## 6. Simulations

In this section, some simulations are provided to illustrate the proposed EEDT-RL and EEDT-UL algorithms. In a  $500 \times 500\text{m}^2$  network region, some mobile nodes are distributed randomly at the beginning. A sequence of network

topologies is generated in the following manner. All nodes move according to the random direction mobility model. That is, each node moves from the current position to the next position with an arbitrary direction and a random velocity selected from the range  $[0, 50\text{m}]$  per time slot. When the mobile node hits the network region boundary, its mobile direction is reversed. By recording each node's position at each time slot, a sequence of static network graphs could be derived according to the maximum transmission range  $r_{\max}$  of each node. The source and destination pairs are selected randomly for each topology instance. Note that the trajectory prediction is not within the scope of this paper. Thus, in our simulation, the random direction mobility model is employed just for generating each node's mobility trajectory in a time period. Some other important simulation parameters are listed in Table 2 [23, 44].

Figure 2 provides an example of topology evolutions at some consecutive time slots with the number of nodes  $n = 20$ . It can be seen that the network topology is varied from one time slot to another and is often disconnected in each time slot. Therefore, it is impossible to transmit data packet in a single time slot between two remote nodes. We should consider a sequence of network topologies in multiple time slots for the design of data transmission scheme.

The proposed EEDT-RL and EEDT-UL algorithms are evaluated according to the following performance metrics: energy cost, actual transmission count, and path reliability. Actual transmission count between source and destination

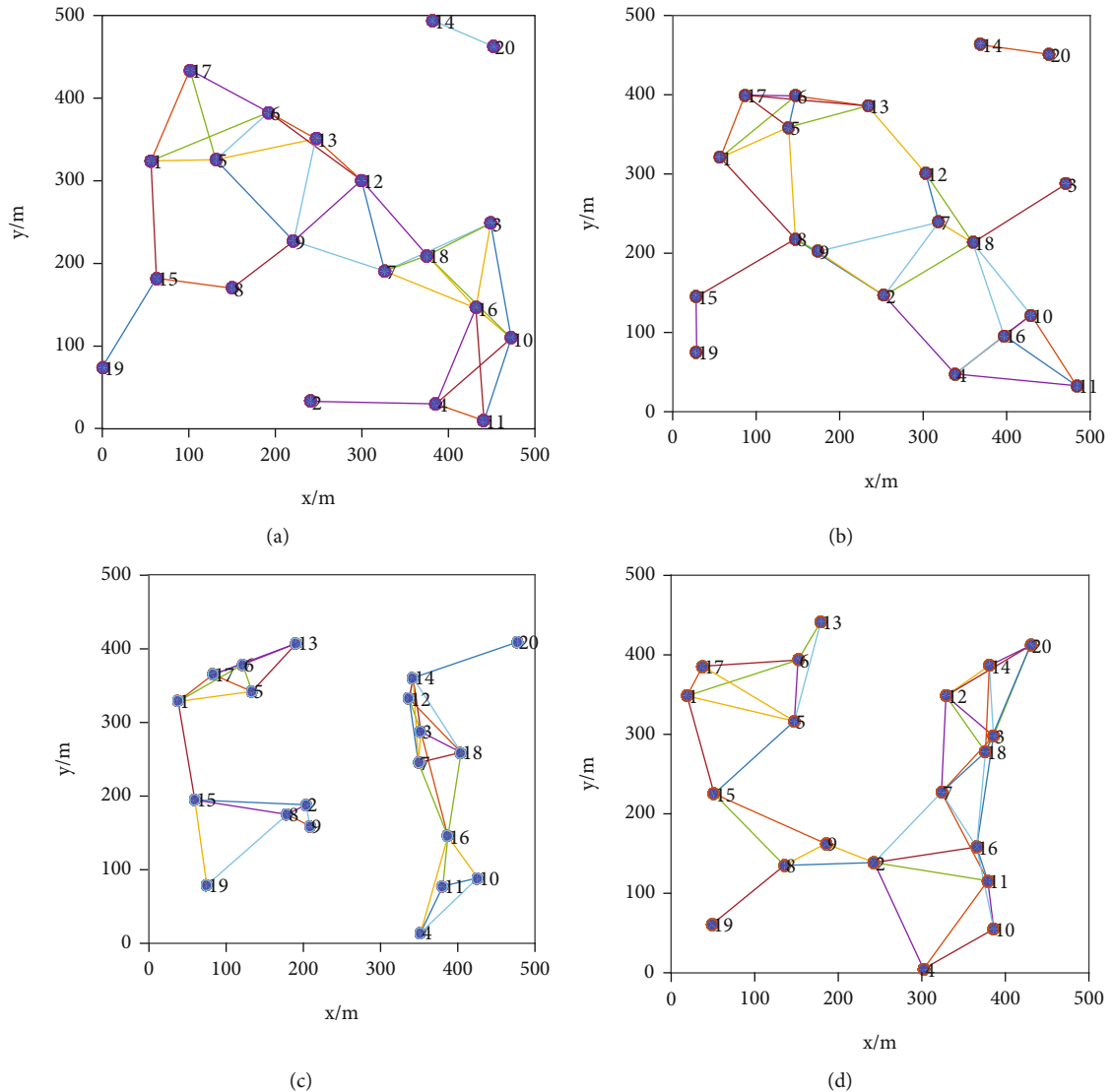


FIGURE 2: An example of topology evolutions at four consecutive time slots.

is equal to the number of spatial links in the space-time path, while path reliability is the production of all links' reliability, which is defined in Subsection 3.2. We compare these performance metrics with two common algorithms, which are often employed for data transmission in mobility networks, i.e., epidemic-based transmission and distance-based transmission. These two common algorithms have many versions in different literatures [3, 46]. In our simulation, we extract the main idea of these two algorithms and make appropriate modifications to fit the space-time graph model. In epidemic-based transmission, the nodes carrying data packet infect the nodes which do not receive the packet utilizing the communication opportunity at each time slot, until the destination node receives this packet. In distance-based transmission, a node, which does not receive the packet, with the minimum distance to the destination is selected as the next forwarding node.

6.1. Simulations on EEDT-RL. In this simulation, we first increase the number of nodes from 10 to 30 and keep

time period at 10 time slots. Figures 3(a) and 3(b) show the variation trend of energy cost and transmission count versus number of nodes. Before executing the proposed EEDT-RL and two comparison algorithms, a sequence of static topology graphs should be converted into a space-time graph, as shown in Subsection 3.1. These algorithms are implemented on 1000 various space-time graphs, and in each space-time graph, one source-destination pair is selected randomly. From Figure 3, the proposed EEDT-RL has the minimum energy cost and the minimum transmission count compared with the epidemic-based and distance-based algorithms. Actually, EEDT-RL can select better communication opportunities for data transmission and hold data packet (do not transmission) for saving energy when the communication opportunities is worse. With the increase of number of nodes, the two metrics of energy cost and transmission count are stable relatively in EEDT-RL and distance-based algorithm, while these two metrics are increasing rapidly in epidemic-based algorithm.

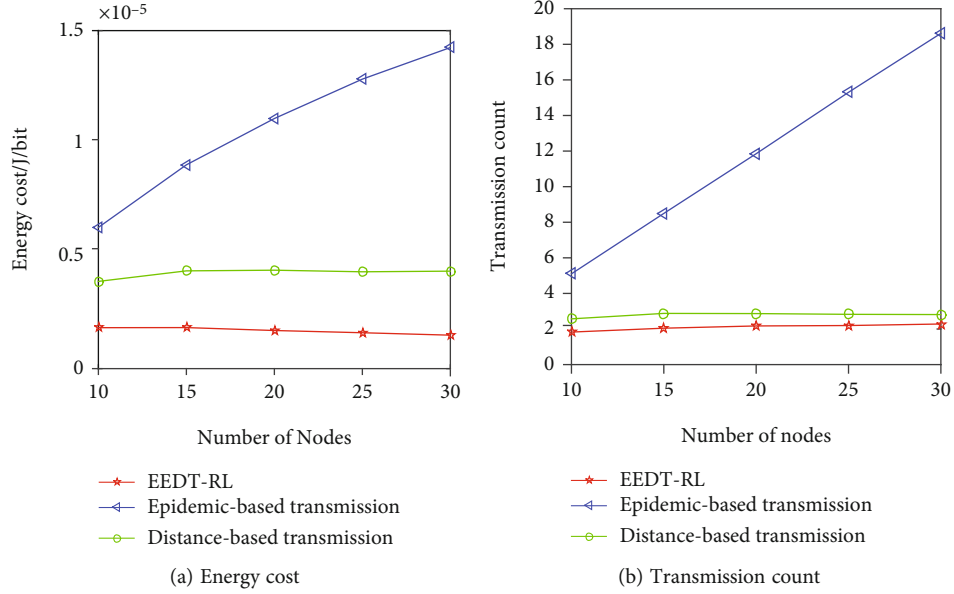


FIGURE 3: Performance metrics of EEDT-RL vs. number of nodes.

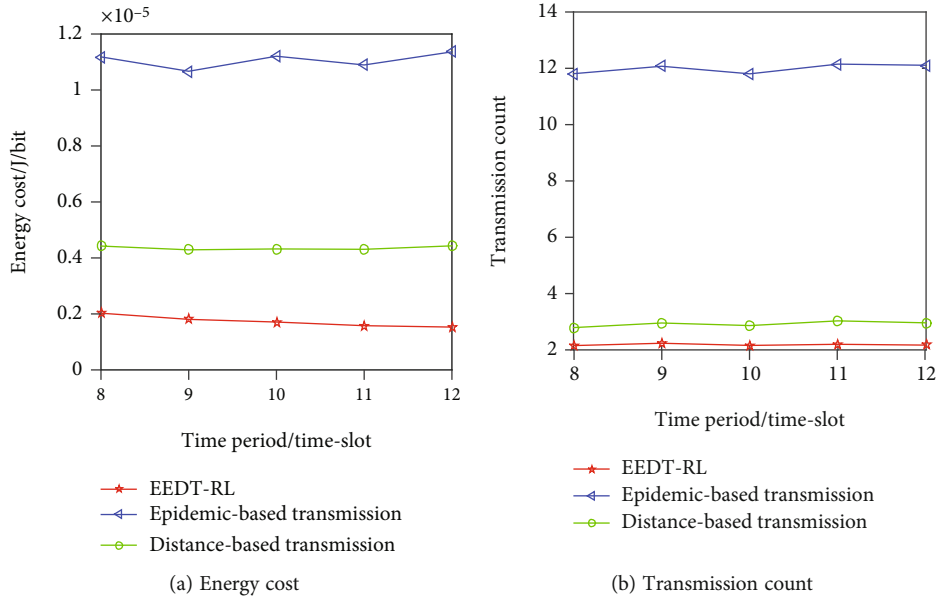


FIGURE 4: Performance metrics of EEDT-RL vs. time period.

The time period  $T$  is varied from 8 to 12 time slots to study the impact of time period on energy cost and transmission count. The curves of energy cost and transmission count vs. time period are reported in Figures 4(a) and 4(b), in which the number of nodes is 20. Similarly, the proposed EEDT-RL generates the most energy-efficient space-time path. From Figure 4(a), the energy cost decreases gradually with the increase of time period in EEDT-RL; that is, a longer time period will induce a more energy-efficient data transmission.

**6.2. Simulations on EEDT-UL.** For the unreliable link model, the reliability probability of a spatial link is assigned a ran-

dom value from 0.6 to 1 in this simulation. We first investigate the tradeoff between energy cost and path reliability of the proposed EEDT-UL by varying the weight factor  $\lambda$  from 0.1 to 0.9. The time period and number of nodes are set as  $T = 10$  time slots and  $n = 20$ . Figures 5(a) and 5(b) plot the curves of energy cost and path reliability vs. weight factor  $\lambda$ , respectively. The link weight changes from 0.1 to 0.9, reflecting the ever-increasing importance of energy cost in data transmission. From Figure 5, the reduction of energy cost comes at the expense of path reliability. We can see the tradeoff between energy cost and path reliability via the adjustment of weight factor  $\lambda$ . It should be noted that the energy cost of the space-time path generated by EEDT-UL

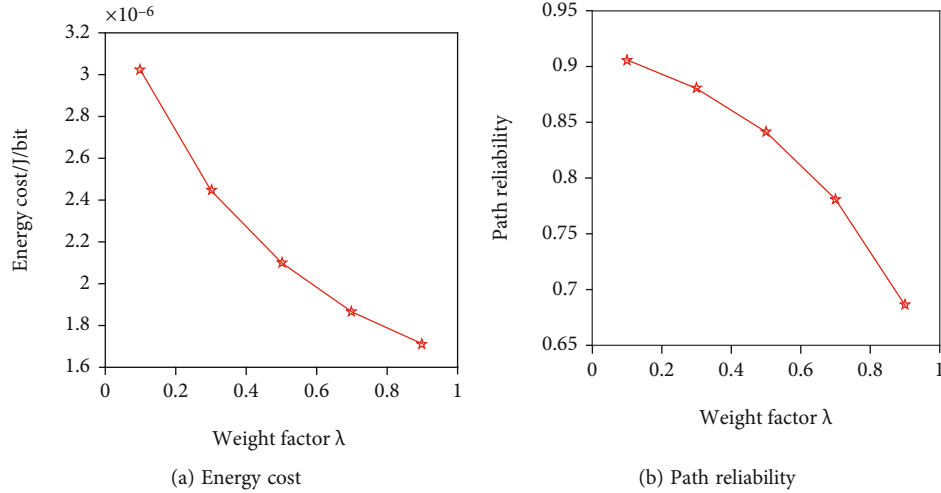


FIGURE 5: Tradeoff between energy cost and path reliability.

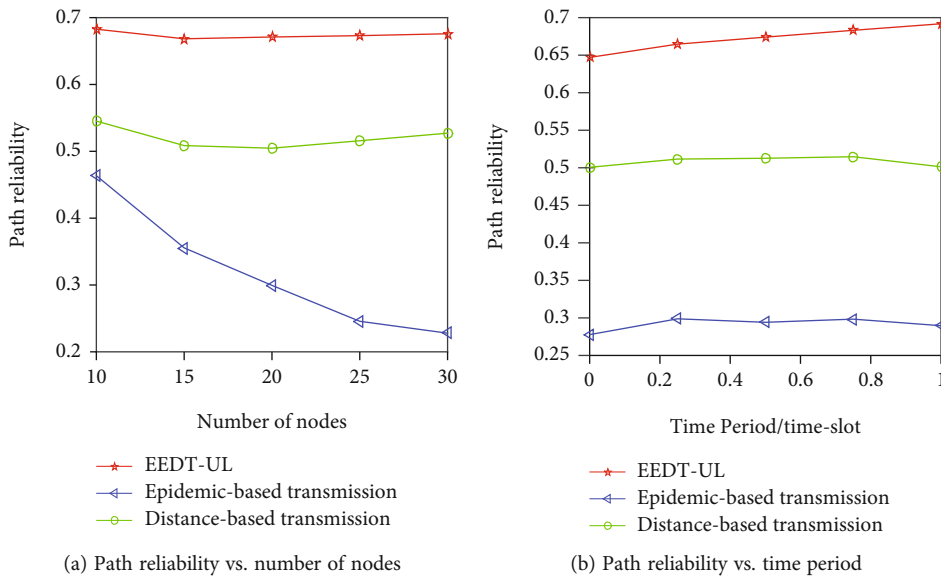


FIGURE 6: Path reliability of EEDT-UL vs. number of nodes and time period.

may not be the actual energy consumption. Because of the unreliable links, the data transmission may fail, and a new space-time path will be generated. If the path reliability is high, the failure probability of data transmission is low, and then the calculated energy cost is equal to the actual value.

The metrics of energy cost and transmission count in EEDT-UL are similar with these in EEDT-RL. Thus, we study the path reliability in EEDT-UL in contrast to epidemic-based and distance-based data transmission. Figures 6(a) and 6(b) show the simulation results of path reliability vs. number of nodes and time period, respectively. Here, we set time period as 10 time slots in Figure 6(a), number of nodes as 20 in Figure 6(b), and weight factor as 0.5 for both. From Figure 6, the proposed EEDT-UL has the maximum path reliability compared with the two common algorithms.

## 7. Conclusions

In this paper, the energy-efficient data transmissions over reliable links and unreliable links in mobility-aware wireless networks are investigated. A sequence of network topologies deduced by mobility prediction, each of which is usually disconnected, was modeled as a virtual space-time graph. Data transmission problems over reliable links and unreliable links on space-time graph were defined, in which a space-time path with the minimum energy cost would be found. Then, EEDT-RL was proposed to find the optimal space-time path under the reliable link model. Under the unreliable link model, we developed a heuristic data transmission method, named EEDT-UL, which could achieve the tradeoff between energy cost and path reliability. The simulation results showed that our proposed algorithms perform well in terms of energy consumption and transmission count



compared with some existing algorithms. Specially, we can further improve the energy-efficiency of data transmission by increasing the number of nodes and/or prolonging the time period, due to that these ways could result in more communication opportunities for selection.

### Data Availability

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

### Conflicts of Interest

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

### Acknowledgments

This research was supported in part by the National Natural Science Foundation of China (62172457), National Natural Science Foundation of Henan (202300410523), Innovation and Entrepreneurship Training Program for College Students of Henan (S202110478028), Key Scientific Research Project of Henan Educational Committee (21A510003, 22A510018), and Science and Technology Development Project of Zhoukou (2021GG02028).

### References

- [1] S. Batabyal and P. Bhaumik, "Mobility models, traces and impact of mobility on opportunistic routing algorithms: a survey," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1679–1707, 2015.
- [2] H. Lin, W. Shin, and B. C. Jung, "Multi-Stream opportunistic network decoupling: relay selection and interference management," *IEEE Transactions on Mobile Computing*, vol. 18, no. 10, pp. 2372–2385, 2019.
- [3] P. Sermpezis and T. Spyropoulos, "Delay analysis of epidemic schemes in sparse and dense heterogeneous contact networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 9, pp. 2464–2477, 2017.
- [4] P. Chen, S. Cheng, and M. Sung, "Analysis of data dissemination and control in social internet of vehicles," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2467–2477, 2018.
- [5] M. Xu, Q. Yang, and Z. Shen, "Joint design of routing and power control over unreliable links in multi-hop wireless networks with energy-delay tradeoff," *IEEE Sensors Journal*, vol. 17, no. 23, pp. 8008–8020, 2017.
- [6] L. Pei, J. Huilin, P. Zhiwen, and Y. Xiaohu, "Energy-Delay tradeoff in ultra-dense networks considering BS sleeping and cell association," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 734–751, 2018.
- [7] X. Ma, X. Zhang, and R. Yang, "Reliable energy-aware routing protocol in delay-tolerant mobile sensor networks," *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 5746374, 11 pages, 2019.
- [8] Y. Luo, M. Zeng, and H. Jiang, "Learning to tradeoff between energy efficiency and delay in energy harvesting-powered D2D communication: a distributed experience-sharing algorithm," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5585–5594, 2019.
- [9] Q. Han, L. Gong, W. Wu, and Y. Bai, "Link availability prediction-based reliable routing for mobile ad hoc networks," *IET Communications*, vol. 5, no. 16, pp. 2291–2300, 2011.
- [10] K. Namuduri and R. Pendse, "Analytical estimation of path duration in mobile ad hoc networks," *IEEE Sensors Journal*, vol. 12, no. 6, pp. 1828–1835, 2012.
- [11] S. Tseng, A. Tang, G. Choudhury, and S. Tse, "Routing stability in hybrid software-defined networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 2, pp. 790–804, 2019.
- [12] O. Younes and U. Albalawi, "Analysis of route stability in mobile multihop networks under random waypoint mobility," *IEEE Access*, vol. 8, pp. 168121–168136, 2020.
- [13] C. Song, Z. Qu, N. Blumm, and A. L. Barabási, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.
- [14] Y. Li, D. Jin, P. Hui, Z. Wang, and S. Chen, "Limits of predictability for large-scale urban vehicular mobility," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 6, pp. 2671–2682, 2014.
- [15] Z. Yang, J. Chen, J. Hu, Y. Shu, and P. Cheng, "Mobility modeling and data-driven closed-loop prediction in bike-sharing systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 12, pp. 4488–4499, 2019.
- [16] G. Qi, A. Huang, W. Guan, and L. Fan, "Analysis and prediction of regional mobility patterns of bus travellers using smart card data and points of interest data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1197–1214, 2019.
- [17] P. Sommer, K. Geissdoerfer, R. Jurdak et al., "Energy- and mobility-aware scheduling for perpetual trajectory tracking," *IEEE Transactions on Mobile Computing*, vol. 19, no. 3, pp. 566–580, 2020.
- [18] S. Sarkar and R. Datta, "Mobility-aware route selection technique for mobile ad hoc networks," *IET Wireless Sensor Systems*, vol. 7, no. 3, pp. 55–64, 2017.
- [19] S. S. Chaudhari, S. Maurya, and V. K. Jain, "MAEER: mobility aware energy efficient routing protocol for internet of things," in *Conference on Information and Communication Technology*, Gwalior, India, 2017.
- [20] Y. Yao, Y. Sun, C. Phillips, and Y. Cao, "Movement-aware relay selection for delay-tolerant information dissemination in wildlife tracking and monitoring applications," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 3079–3090, 2018.
- [21] R. Jiang, Y. Zhu, X. Wang, and L. M. Ni, "TMC: exploiting trajectories for multicast in sparse vehicular networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 1, pp. 262–271, 2015.
- [22] Y. Tang, N. Cheng, W. Wu, M. Wang, Y. Dai, and X. Shen, "Delay-minimization routing for heterogeneous VANETs with machine learning based mobility prediction," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3967–3979, 2019.
- [23] F. Li, S. Chen, M. Huang, Z. Yin, C. Zhang, and Y. Wang, "Reliable topology design in time-evolving delay-tolerant networks with unreliable links," *IEEE Transactions on Mobile Computing*, vol. 14, no. 6, pp. 1301–1314, 2015.
- [24] X. Zhao, Y. Zhang, C. Jiang, J. Yuan, and J. Cao, "Mobile-aware topology control potential game: equilibrium and connectivity," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1267–1273, 2016.
- [25] H. Ko, J. Lee, and S. Pack, "MALM: mobility-aware location management scheme in femto/macrocell networks," *IEEE*

- Transactions on Mobile Computing*, vol. 16, no. 11, pp. 3115–3125, 2017.
- [26] N. Xing, Q. Zong, L. Dou, B. Tian, and Q. Wang, “A game theoretic approach for mobility prediction clustering in unmanned aerial vehicle networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 9963–9973, 2019.
- [27] J. Li, X. Zhang, J. Zhang, J. Wu, Q. Sun, and Y. Xie, “Deep reinforcement learning-based mobility-aware robust proactive resource allocation in heterogeneous networks,” *IEEE Transactions on Cognitive Communications and Networks*, vol. 6, no. 1, pp. 408–421, 2020.
- [28] M. Xu, H. Zhu, H. Xu, J. Song, and Z. Luo, “Reliable routing design in predictable wireless networks with unreliable links,” in *16th International Conference on Computational Intelligence and Security*, Guangxi, China, 2020.
- [29] N. Basilico, M. Cesana, and N. Gatti, “Algorithms to find two-hop routing policies in multiclass delay tolerant networks,” *IEEE Transactions on Wireless Communications*, vol. 15, no. 6, pp. 4017–4031, 2016.
- [30] E. Altman, F. de Pellegrini, D. Miorandi, and G. Neglia, “Adaptive optimal stochastic control of delay-tolerant networks,” *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 1815–1829, 2017.
- [31] Z. Du, C. Wu, T. Yoshinaga et al., “A routing protocol for UAV-assisted vehicular delay tolerant networks,” *IEEE Open Journal of the Computer Society*, vol. 2, pp. 85–98, 2021.
- [32] D. Popescu, P. Jacquet, B. Mans, R. Dumitru, A. Pastrav, and E. Puschita, “Information dissemination speed in delay tolerant urban vehicular networks in a hyperfractal setting,” *IEEE/ACM Transactions on Networking*, vol. 27, no. 5, pp. 1901–1914, 2019.
- [33] R. Cavallari, S. Toumpis, R. Verdone, and I. Kontoyiannis, “Packet speed and cost in mobile wireless delay-tolerant networks,” *IEEE Transactions on Information Theory*, vol. 66, no. 9, pp. 5683–5702, 2020.
- [34] H. Zhang, H. Dai, Z. Zhang, and Y. Huang, “Mobile conductance in sparse networks and mobility-connectivity tradeoff,” *IEEE Transactions on Wireless Communications*, vol. 15, no. 4, pp. 2954–2965, 2016.
- [35] L. Bedogni, M. Fiore, and C. Glacet, “Temporal reachability in vehicular networks,” in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pp. 81–89, Honolulu, USA, 2018.
- [36] X. Kui, A. Samanta, X. Zhu, S. Zhang, Y. Li, and P. Hui, “Energy-aware temporal reachability graphs for time-varying mobile opportunistic networks,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 10, pp. 9831–9844, 2018.
- [37] T. Li, K. Xiong, P. Fan, and K. B. Letaief, “Service-oriented power allocation for high-speed railway wireless communications,” *IEEE Access*, vol. 5, pp. 8343–8356, 2017.
- [38] F. Li, H. Jiang, H. Li, Y. Cheng, and Y. Wang, “SEBAR: social-energy-based routing for mobile social delay-tolerant networks,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 7195–7206, 2017.
- [39] Y. Liu, X. Chen, Y. Niu, B. Ai, Y. Li, and D. Jin, “Mobility-aware transmission scheduling scheme for millimeter-wave cells,” *IEEE Transactions on Wireless Communications*, vol. 17, no. 9, pp. 5991–6004, 2018.
- [40] B. Du, X. Di, D. Liu, and H. Zhang, “Dynamic graph optimization and performance evaluation for delay-tolerant aeronautical ad hoc network,” *IEEE Transactions on Communications*, vol. 69, no. 9, pp. 6018–6036, 2021.
- [41] B. Baron, P. Spathis, M. Dias de Amorim, Y. Viniotis, and M. H. Ammar, “Mobility as an alternative communication channel: a survey,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 289–314, 2019.
- [42] L. Guntupalli, J. Martinez-Bauset, F. Y. Li, and M. A. Weitnauer, “Aggregated packet transmission in duty-cycled WSNs: modeling and performance evaluation,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 563–579, 2017.
- [43] M. Xu, H. Zhu, H. Xu, and X. Yang, “Energy-efficient routing under delay constraint in duty-cycle wireless sensor networks,” *International Journal of Sensor Networks*, vol. 33, no. 1, pp. 8–15, 2020.
- [44] Q. Zhao and L. Tong, “Energy efficiency of large-scale wireless networks: proactive versus reactive networking,” *IEEE Journal of Selected Areas on Communications*, vol. 23, no. 5, pp. 1100–1112, 2005.
- [45] T. Korkmaz, M. Krunz, and S. Tragoudas, “An efficient algorithm for finding a path subject to two additive constraints,” *Computer Communications*, vol. 25, no. 3, pp. 225–238, 2002.
- [46] D. Chen, M. Haenggi, and J. N. Laneman, “Distributed spectrum-efficient routing algorithms in wireless networks,” *IEEE Transactions on Wireless Communications*, vol. 7, no. 12, pp. 5297–5305, 2008.

## Research Article

# Research on Distributed Multisensor Spectrum Semantic Sensing and Recognition in Internet of Things Environment

Xiguo Liu <sup>1,2</sup>, Jing Zhang,<sup>3</sup> Min Liu,<sup>2</sup> Zhongyang Mao <sup>1,2</sup> and Changbo Hou<sup>3</sup>

<sup>1</sup>Aviation Communication Teaching and Research Section, Naval Aeronautic University, Yantai 264001, China

<sup>2</sup>State Key Laboratory of Signal and Information Processing of Shandong Province, Yantai 264001, China

<sup>3</sup>College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China

Correspondence should be addressed to Zhongyang Mao; [freedom\\_mzy@163.com](mailto:freedom_mzy@163.com)

Received 21 January 2022; Accepted 23 March 2022; Published 20 May 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Xiguo Liu 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 brought about a new era of Internet of Thing (IoT), and at the same time, electromagnetic spectrum monitoring and sensing have also ushered in huge challenges. Digital modulation recognition technology is an important content of electromagnetic spectrum sensing. In the increasingly complex wireless communication transmission environment, especially in noncooperative communication, it becomes more and more difficult to receive target signals and accurately extract effective semantic information from diverse modulation signals of the electromagnetic spectrum. At this stage, with the rapid development of network information and wireless communication technology, within a prescribed distance, the IoT built by many sensors has attracted wide attention from people in related fields. This paper proposes a distributed collaborative sensing spectrum semantic recognition architecture for communication signals based on feature fusion. Perform wireless communication and transmission between multiple sensors to form a self-organizing network to cooperatively sense signal semantic information, and extract the signal features of each sensor in the distributed network structure. Finally, the extracted sensor features are semantically analyzed and modeled, and the effective features are fused to complete the entire perception and recognition process. Even if the channel environment of a small number of receiving nodes deteriorates in a complex transmission environment, the signal quality features can still be accurately extracted, the classification and recognition effect like or higher than the best channel state performance can be achieved, and the fault tolerance of the system can be effectively improved. It can also enhance the performance of spectral semantic information sensing and recognition in the IoT environment.

## 1. Introduction

The rapid development of technology in the communication field has opened a new era of IoT with the emergence of 5G technology, followed by richer application scenarios and increasingly complex electromagnetic environments, which has brought spectrum monitoring management and electromagnetic spectrum sensing utilization huge challenge. In the complex electromagnetic environment where everything is interconnected, accurate perception of semantic spectrum information and identification of signal modulation methods can provide important information for communication networking, etc., thereby effectively improving spectrum utilization efficiency. Typical spectrum semantic

sensing and recognition (SMSR) methods are mainly divided into traditional likelihood function-based decision-making [1, 2], feature extraction-based pattern recognition [3, 4], and other methods, as well as deep learning (DL) methods that have emerged recently [5]. Traditional methods mainly extract specific features manually, and the recognition effect largely depends on manual experience, which leads to poor recognition performance and fewer recognition types. With the advent of AlphaGo in 2015, more and more researchers are focusing on DL. It has achieved good results in classification tasks with its outstanding feature extraction capabilities. O'shea and West [6] built a simulated communication model through GNU Radio and collected 11 communication signals, using a Convolutional Neural Network (CNN) to

extract signal features from In-phase and Quadrature ( $I/Q$ ) components. The DL method has no expert-derived features, showing a great accuracy improvement over traditional statistical methods. Wu et al. [7] improved the CNN and added a Long and Short-Term Memory (LSTM) structure, which improved the network's feature extraction ability for signal timing and increased the signal recognition accuracy to 80%. Wu et al. [8] combined the cyclic spectrogram and constellation diagram and simulated it on the public dataset. When the signal-to-noise ratio (SNR) is 0 dB, the accuracy rate reaches 80% and the training time is shortened. In the next few years, researchers used more complex DL architectures [9–13], using extracted feature inputs and neural network pruning to improve operating efficiency. With the increasingly complex electromagnetic environment of signals, DL has gradually become the mainstream algorithm in spectrum semantic sensing (SMS) and modulation recognition algorithms relying on powerful feature extraction capabilities and robustness. Although communication signal modulation recognition technology has gradually matured and the results have become more abundant [14], with the rapid development of wireless communication technology, signal transmission scenarios have become increasingly diversified, and application requirements have become increasingly updated [15], all of which promote the improvement of modulation methods. Therefore, modulation recognition technology always needs to be constantly updated according to changes in application scenarios and application requirements.

In the actual wireless communication environment, the single-node SMS technology is easily affected by factors such as multipath effects, hidden terminals, and path loss and cannot obtain correct sensing results. At this stage, with the rapid development of network information and wireless communication technology, within a prescribed distance, the IoT built by a large number of nodes has attracted wide attention from people in related fields, and the number of network devices and sensors deployed in the physical environment is rapidly increasing. The increase also brings new challenges to the wireless SMS and recognition, and the research on the distributed network architecture [16] of the combination of multiple receivers arises at the historic moment. Distributed multisensor node wireless SMSR technology can be divided into data layer-based fusion, feature layer-based fusion, and decision-making layer-based fusion schemes. Zhang et al. [17] proposed that the automatic modulation recognition scheme based on multisensor signal fusion can provide higher reliability than single-sensor signals. Dulek [18] proposed a classifier based on online and distributed expectation maximization, which can achieve a classification and recognition effect similar to the best channel state performance. Distributed recognition technology is widely used in optical fiber vibration sensing recognition [19, 20]; Sun et al. [21] developed an improved deep learning method based on a serial fusion feature extraction model for an optical fiber distributed vibration sensing system which can automatically extract and identify effective features. Distributed fusion schemes based on the feature layer mostly use artificial features to achieve [22–24], but in noncoopera-

tive communication scenarios, the received signal is usually a weak signal, which makes it difficult to obtain accurate feature expression. Although the fusion scheme based on the data layer can enhance the received signal strength to a certain extent, it is often necessary to perform centralized calculation and processing on the data of each node in the fusion center, which causes the fusion center to be overloaded. The distributed recognition architecture based on the decision-making layer needs to clarify the influence factors of each node on the final decision. Although the decision result can improve the recognition performance, it needs to know the prior information such as the SNR of the signal at the receiving end of each receiver node. It is not conducive to signal recognition in noncooperative communication scenarios. Therefore, in order to improve the performance of spectrum semantic perception and recognition in complex electromagnetic environments such as noncooperative communication scenarios, this study uses the outstanding feature extraction capabilities of DL methods to propose a distributed collaborative recognition scheme based on feature fusion. The main contributions of this study are as follows:

- (1) Build a distributed multisensor signal reception scene, and set up a transmitter and multiple receiver nodes for signal reception. Model the spectral semantic information in distributed scenarios, and simulate the transmission of communication signals in different state channels through simulation experiments. The specific scene settings are introduced in the next section. In addition, the method proposed in this paper can to a certain extent solve the problem of inability to perform accurate SMSR recognition modulation recognition in noncooperative communication confrontation scenarios due to weak received signals
- (2) The use of DL algorithms is to realize the feature semantic information extraction of multireceiver node signals, the more accurate feature expression can be obtained by fusion of multinode features' semantic information, and the dimensionality reduction of the fused features is performed through the classifier to complete the distributed and coordinated communication signal recognition. It can eliminate the uncertainty of communication signal recognition caused by poor channel conditions to a certain extent

## 2. Materials and Methods

Under the IoT, the rapid development and comprehensive use of digital communication technology have brought huge challenges to the task of electromagnetic spectrum sensing. Recognition of communication signals, as one of the key technologies for electromagnetic spectrum monitoring, is of great significance in both military and civilian fields. However, most of the current researches are limited to a single node. Channel conditions and received signal strength will directly affect recognition performance. A single node



is affected by environmental changes and has poor fault tolerance performance, which leads to its signal recognition effect; when the channel environment is poor, the effect will be poor. With the development of wireless sensor networks, spectral semantic sensing, signal estimation, and recognition algorithms have received more and more attention. Distributed deployment of multiple sensors in the monitoring area forms a wireless communication-based self-organizing network system, which can realize multisensor collaboration to sense the sensing objects in the detection area and finally send the collected semantic information to the control center for further processing. The use of multinode data fusion, feature fusion, and other methods can greatly eliminate the ambiguity of unknown signals. When the channel conditions of a small number of sensor nodes deteriorate, the recognition probability can still be finally maintained. In this study, the communication signal features extracted by each sensor node are fused, and the fused features' semantic information is used to identify the signal modulation mode, to complete the identification process of the entire distributed algorithm. Figure 1 shows the distributed cooperative signal sensing and recognition framework based on feature fusion in this study, which can be divided into three modules, the distributed signal receiving module, the CLDNN feature semantic extraction module, and the fusion classification and recognition module.

The specific process is shown in Algorithm 1

**2.1. Distributed Signal Reception.** Through the research and development of networking information systems and autonomous sensing and intelligent information equipment, the electronic equipment system is developing towards decentralization, networking, and distributed coordination, which greatly improves the level of electronic warfare. The network communication system based on the distributed concept is promoting the development of combat intelligence. The distributed scenario of this study is shown in Figure 2.

In the distributed scenario built in this paper, there are one transmitter and multiple receivers. The number of receivers needs to be determined according to the actual scene requirements, and different numbers of receivers often affect the final recognition performance. The transmitting end signal transmits the communication signal to each receiving end through different channels. After the signal features of each receiver are extracted, the features semantic information of different receivers can be analyzed and fused in the fusion classification center.

In the signal receiving module, the receiver converts the received signal into a baseband modulated signal through digital downconversion and other processing. The signal model received by each signal in the AWAG channel can be expressed as

$$x_k(n) = h_k s(n) + \text{noise}, \quad n = 1, 2, \dots, N. \quad (1)$$

Among them,  $s(n)$  is the signal sequence sent by the transmitter,  $h_k$  represents the channel coefficient,  $x_k(n)$  is the signal at the receiving end, and noise is Gaussian white noise. However, due to the influence of distance and other

factors in actual signal transmission, the signals of different receiver nodes have different delays and other factors. It can be further expressed as

$$x_k(n) = h_k s(n - D_k) + \text{noise}, \quad n = 1, 2, \dots, N, \quad (2)$$

where  $D_k$  represents the channel transmission delay of the  $k$ th receiver. The received signal quality of different receivers mainly depends on  $h_k$  and  $D_k$ .

For the popular digital receivers on the market, especially the software radio platform (SDR), the received communication signal is often a baseband  $I/Q$  complex sequence. Therefore, it is very necessary to start with the baseband  $I/Q$  data to perform modulation recognition on the signal. In this study, a vector is used to represent the received complex signal sequence with noise, and the signal model received by the  $k$ th receiver can be expressed as

$$X_k = [x(1), x(2), \dots, x(n)], \quad n = 1, 2, \dots, N. \quad (3)$$

Feature extraction is performed on the received signals of each receiver node and then sent to the fusion classification center for fusion analysis and modeling of feature semantic information to further complete the recognition.

**2.2. Feature Extraction and Analysis.** In recent years, DL methods have stood out among many machine learning methods by their neural network architecture, algorithms, and optimization technologies. They have been widely used in machine vision and speech recognition and have achieved a series of breakthroughs. DL is a method that effectively uses a data-driven approach to extract features and accurately identify it. Compared with manual feature design and extraction, DL algorithms can effectively extract the shallow features and implicit features of the data, while also saving time. The natural attributes of big data in the communication field have led scholars to explore the possibility of applying DL to the communication field, such as the use of deep neural networks for modulation recognition and radar waveform recognition. Therefore, this study will also use the method based on DL for the feature extraction of the distributed collaborative recognition of communication signals.

CNN is mostly used in the field of image processing. In recent years, they have also been widely used in the field of communication for automatic modulation recognition of signals. The convolutional layer extracts the local features of the data through the convolution kernel; the LSTM network is a special recurrent neural network designed to avoid long-term dependence problems. The difference between the deep neural network (DNN) and recurrent neural network (RNN) and CNN is that DNN specifically refers to a fully connected neuron structure and does not include convolutional units or temporal associations. It can convert the extracted features into a feature space making the output easier to classify.

CNN is good at reducing frequency changes, LSTM is good at time modeling, and DNN is suitable for mapping features to more separable spaces. Therefore, CNN, LSTM,

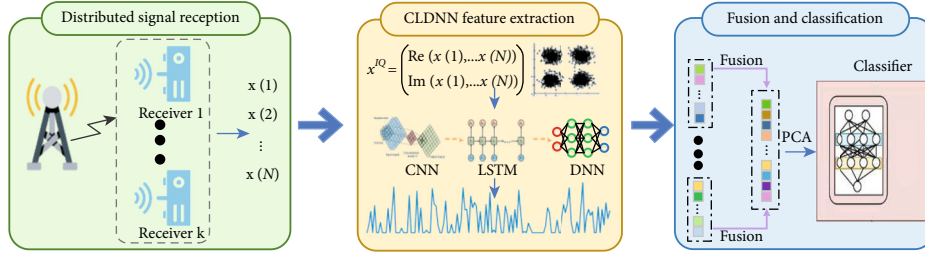
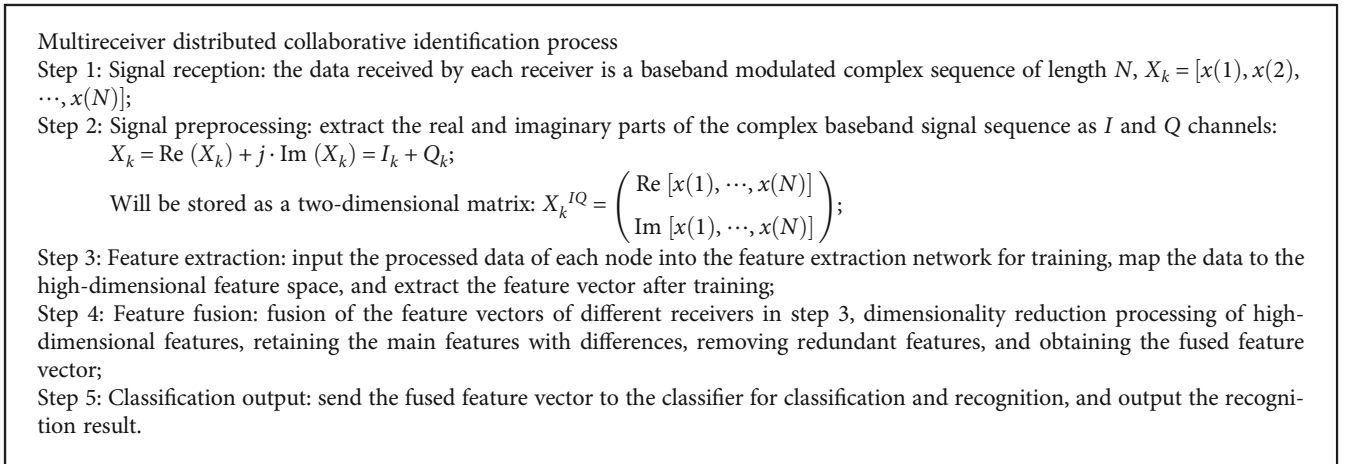


FIGURE 1: Distributed cooperative signal recognition framework based on feature fusion. This framework includes a distributed signal reception module, a CLDNN feature extraction module, and a feature fusion and classification module.



ALGORITHM 1: Multireceiver distributed collaborative identification process.

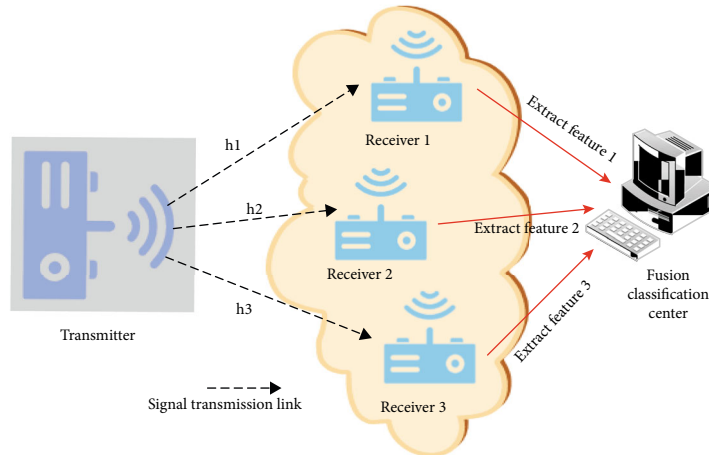


FIGURE 2: Distributed signal transmission. This scene includes one signal transmitter and multiple signal receivers (the number of receivers can be set according to the actual scene; this figure sets three for illustration). Parameter  $h$  represents the transmission parameters in each channel.

and DNN are complementary in modeling capabilities. Sainath et al. [25] use the complementarity of CNN, LSTM, and DNN to build a CLDNN network model for speech signal recognition, which has an improved effect compared to the three models used alone. There is a natural similarity between speech signal and communication signal, or it can be said that speech signal is a kind of communication signal.

In terms of data representation, it is a discrete correlation sequence in the time domain, but the digital modulation signal data is included in natural language processing. In addition to the same information that it carries, the more important thing is its modulation information. Different from natural language processing, its modulation information is only related to the current symbol and a few adjacent



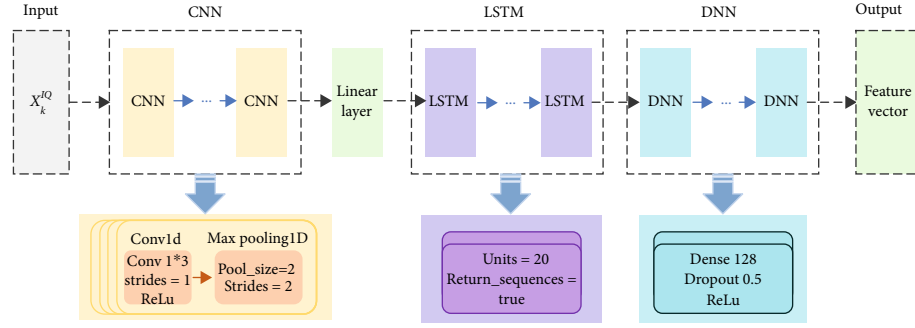


FIGURE 3: CLDNN feature extraction structure. The network contains 7 CNN layers, two LSTM layers, and two DNN layers.

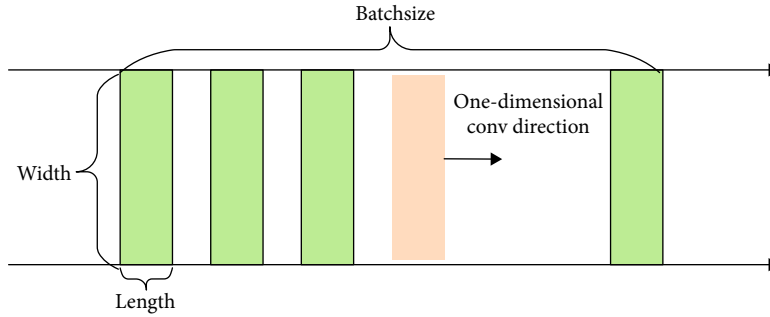


FIGURE 4: One-dimensional convolution diagram.

TABLE 1: LSTM and DNN network parameter settings.

Layers	Input: CNN feature ( $\mathbb{R}^{M \times 512}$ ) Kernel parameters	Output shape
LSTM	Units = 20	(None, $M$ , 20)
LSTM	Units = 20	(None, 20)
Dense+ReLU	128	(None, 128)
Dropout	0.5	(None, 128)
Dense+ReLU	128	(None, 128)
Dropout	0.5	(None, 128)

Output: CLDNN feature (Dimension = 128)

TABLE 2: Simulation parameter setting.

Dataset parameter setting	
Modulation	8 classes (BPSK, QPSK, 8PSK, 16PSK, 16QAM, 64QAM, 256QAM, PAM4)
Sample rate and length	100 kHz and 1024
SNR (dB) range	-20 : 2 : 18
Training dataset	$6400 \times 1024 \times 2$ (6400 instances)
Test dataset	$1600 \times 1024 \times 2$ (1600 instances)

symbols, not a real-time sequence. With obvious timing features, this paper builds a CLDNN network model that is more suitable for signal modulation recognition based on [21] for feature extraction. Its structure is shown in Figure 3.

In traditional feature extraction methods, only one channel of  $I/Q$  data is usually used for processing. In order to fully extract the subtle features' semantic information of the received signal, this study converts the received complex number sequence into  $I/Q$  two channels of data:

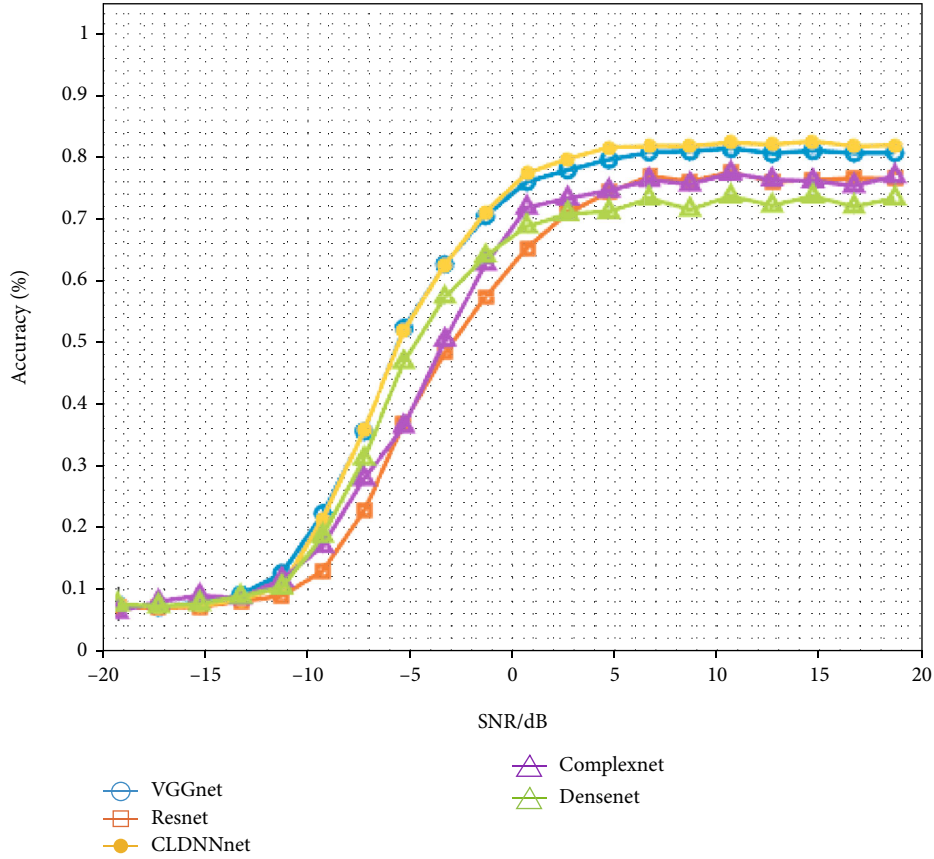
$$X_k = \text{Re}(X_k) + j \cdot \text{Im}(X_k) = I_k + Q_k, \quad (4)$$

where  $I_k$  is the real part of the complex modulated signal and  $Q_k$  is the imaginary part. The  $I/Q$  two-way data is stored as a two-dimensional matrix as the input of the feature extraction network, that is,  $\text{Input} = [N \times 2]$ , where  $N$  is the

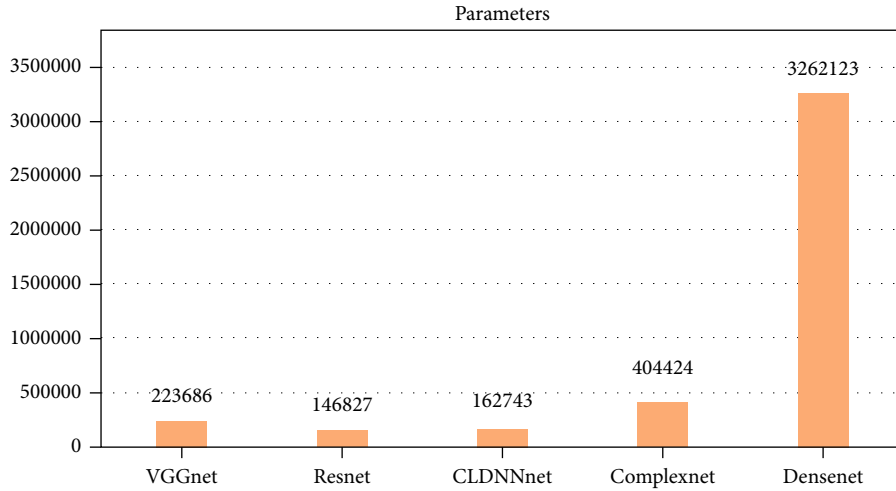
signal length, as shown in

$$X_k^{IQ} = \begin{pmatrix} \text{Re}[x(1), \dots, x(N)] \\ \text{Im}[x(1), \dots, x(N)] \end{pmatrix}. \quad (5)$$

Input the two-dimensional matrix sequence of the digital modulation signal into the CNN network, and complete the feature extraction and dimensionality reduction at this stage after convolution and pooling operations. To adapt to the features of the input  $I/Q$  sequence, this study uses the one-dimensional convolution kernel commonly used in sequences to replace the traditional two-dimensional convolution kernel to extract features, as shown in Figure 4, and the expression of the one-dimensional convolution kernel



(a) The recognition accuracy of different models



(b) Comparison of parameters of different models

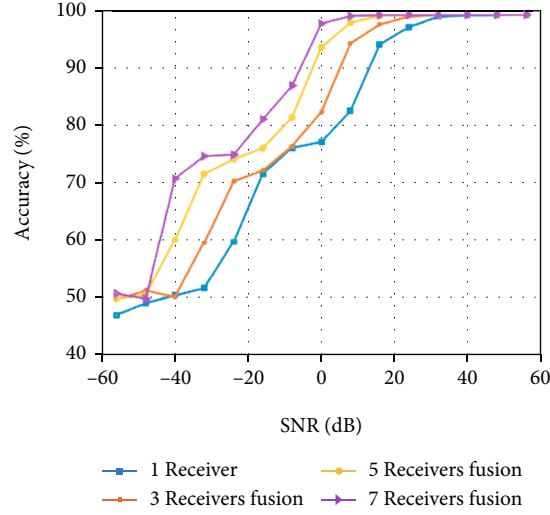
FIGURE 5: Comparison of classification performance of different network models. The experimental results are based on the RadioML2016.10a dataset.

is shown in

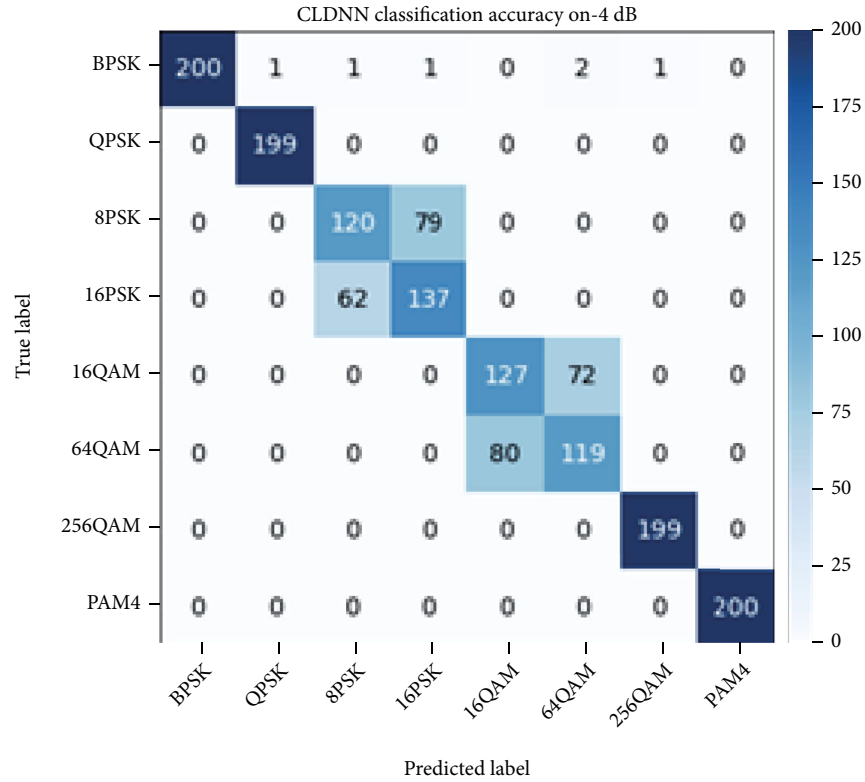
$$y_j^l = f\left(\sum_{i=1}^{M_j} \omega_{i,j} \times x_i^{l-1} + b_j^l\right), \quad j = 1, 2, \dots, N, \quad (6)$$

where  $y_j^l$  is the first feature of the layer,  $\omega_{i,j}$  is the weight

value of the  $j$  feature of the  $i$  layer,  $b_j^l$  is the offset of the  $j$  feature in the  $l$  layer,  $N$  is the number of feature maps in the  $l$  layer,  $M$  represents the size of the one-dimensional convolution kernel,  $f(\cdot)$  represents the activation function, and  $f(\cdot) = \max(0, \cdot)$ . In this study, the main features of the modulated signal are extracted by 7 layers of convolution kernels with a size of  $1 \times 3$ . The number of convolution kernels are 64, 128, 128, 256, 256, and 512, respectively.



(a) Recognition results under different SNRs



(b) 7 receivers cooperatively identify confusion matrix under -4 dB

FIGURE 6: Feature fusion recognition performance of different numbers of receivers: (a) the accuracy change curve of a single receiver and different numbers of receivers under cooperative recognition at -14-14 dB and (b) the effect of cooperative recognition by 7 receivers under -4 dB.

To reduce the size of the model and increase the calculation speed, this study uses a one-dimensional maximum pooling method with a step size of 2 to downsample the convolutional feature map. It should be referred to as

$$y_j^l = f\left(\text{down}\left(y_j^{l-1}\right)\right). \quad (7)$$

Among them,  $y_j^l$  and  $y_j^{l-1}$  represent the feature map of the  $l$  and  $l-1$  layer, respectively, and  $\text{down}(\cdot)$  represents the downsampling.

After the input data vector  $N \times 2$  is convolved and pooled, the feature space at this time can be expressed as  $\mathbb{R}^{M \times 512}$ . The long input information is converted into a shorter high-level feature sequence as the input of the LSTM

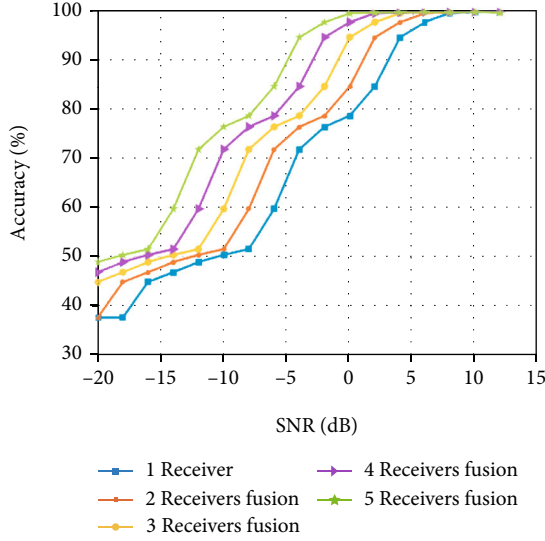


FIGURE 7: Recognition results under different SNRs (take the SNR of receiver 1 as the abscissa).

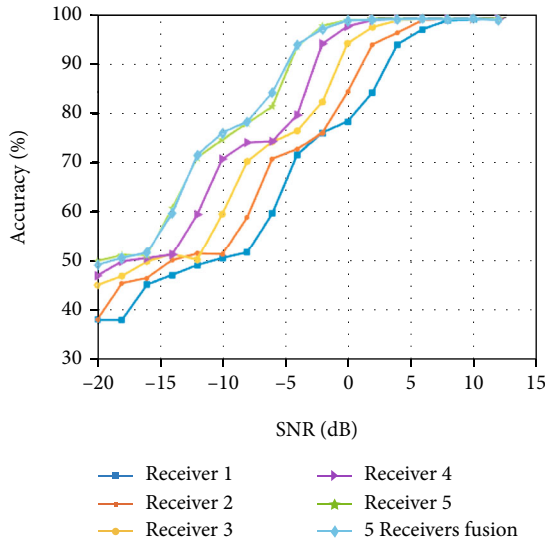


FIGURE 8: Comparison of collaborative recognition effect between different receivers and multireceivers.

module, using its features to learn the features of several adjacent symbols and input into the DNN network module, and the extracted features are mapped to a feature space that is easier to separate, and the final feature vector is obtained. The LSTM and DNN network parameter settings are shown in Table 1.

**2.3. Fusion and Classification.** The previous section introduced how to extract the features of the digital modulation signals of different receiver nodes. In order to make full use of the semantic information of different receivers in the communication network and further improve the accuracy of signal recognition, next, this section will introduce how to fuse feature semantics and collaborative identification of data from different receiver nodes.

Feature-level fusion refers to a fusion method that is completed by integrating or combining features from all nodes. Its purpose is to use the complementarity of each single node semantic information to synthesize the extracted features into a feature that is more discriminative than the input feature.

This study uses the aforementioned method to complete the feature extraction of the received data and finally extracts 128-dimensional features for each receiver node's data as the feature vector of the signal sequence of the node, and the resulting fusion feature vector is as follows:

$$F_{\text{fusion}} = F_1(X_1^{IQ}) \oplus F_2(X_2^{IQ}) \oplus \dots \oplus F_k(X_k^{IQ}). \quad (8)$$

Among them,  $F_{\text{fusion}}$  represents the feature vector after fusion, the size of each sample is  $1 \times 128$ ,  $F_k(\cdot)$  represents the data feature vector of each receiver obtained using the feature extraction method in this study, and  $\oplus$  represents the fusion operation on the feature vector of each node, and the size of the fused feature vector is  $1 \times (128 * k)$ .

After extracting effective feature vectors from single-node receiver data and analyzing them, perform feature vector fusion on multinode feature data. The obtained feature fusion vector can reduce the influence of channel quality and signal strength on the extracted features and can represent more modulation information about the signal than the vector extracted by a single node. But at the same time, multiple nodes also increase the complexity of problem analysis. Therefore, it is necessary to process the fusion features, so as to reduce the feature parameters while retaining the effective features to the greatest extent and complete the comprehensive analysis of the feature data. In summary, it is necessary to reanalyze the closely related feature parameters, eliminate redundant feature quantities, and finally realize the information contained in each feature with fewer comprehensive feature parameters. This study uses the same network model for feature extraction on different receiver node data, so the Principal Component Analysis (PCA) algorithm, which is often used in high-dimensional vector analysis, is used to process the fused feature data.

Assuming that the number of receiver nodes is  $k$ , the feature of each node has a dimension of  $m$ , the data of each node can be expressed as  $X^1, X^2, \dots, X^k$ , the data feature of each node can be expressed as  $X^k = [x_1^k, x_2^k, \dots, x_m^k]^T$ , and the error in the sample mapping process can be expressed as

$$\text{error} = \frac{1}{k} \sum_{i=1}^k \|X^i - X_{\text{map}}^i\|^2. \quad (9)$$

Among them,  $X_{\text{map}}^i$  represents the new feature after mapping, and the dimension remains unchanged. The process of PCA is as follows.

#### (1) Feature normalization

Normalize the training samples to obtain the training parameters and then normalize the test samples.

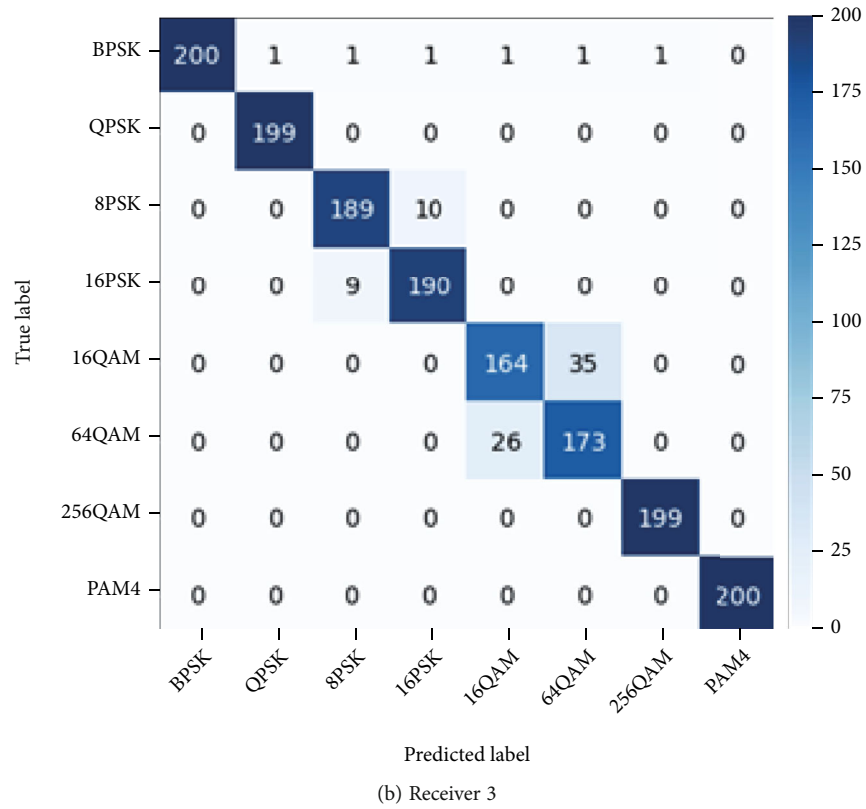
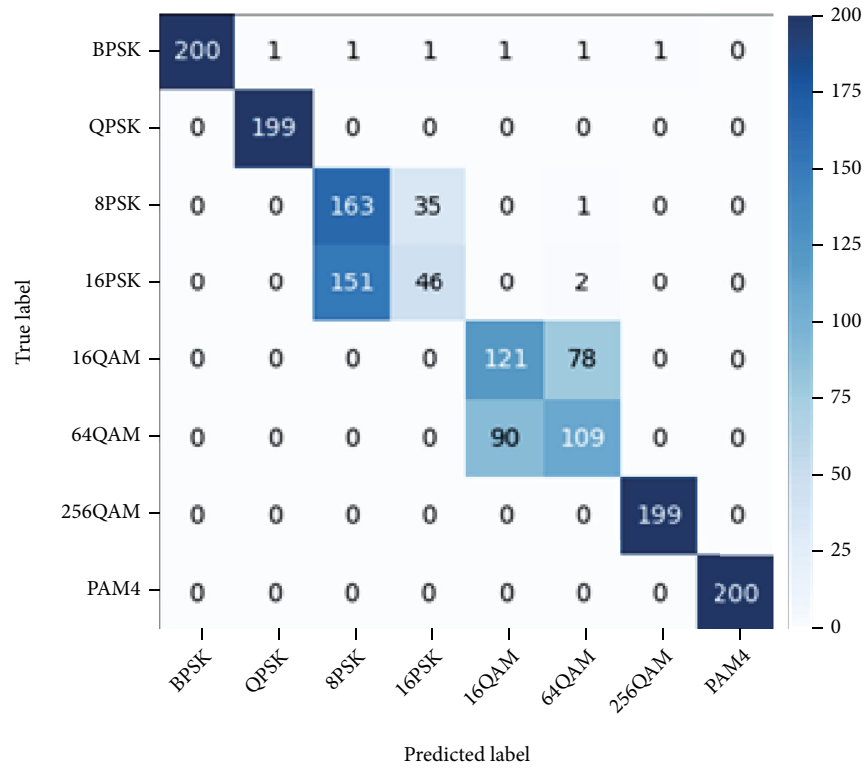


FIGURE 9: Continued.

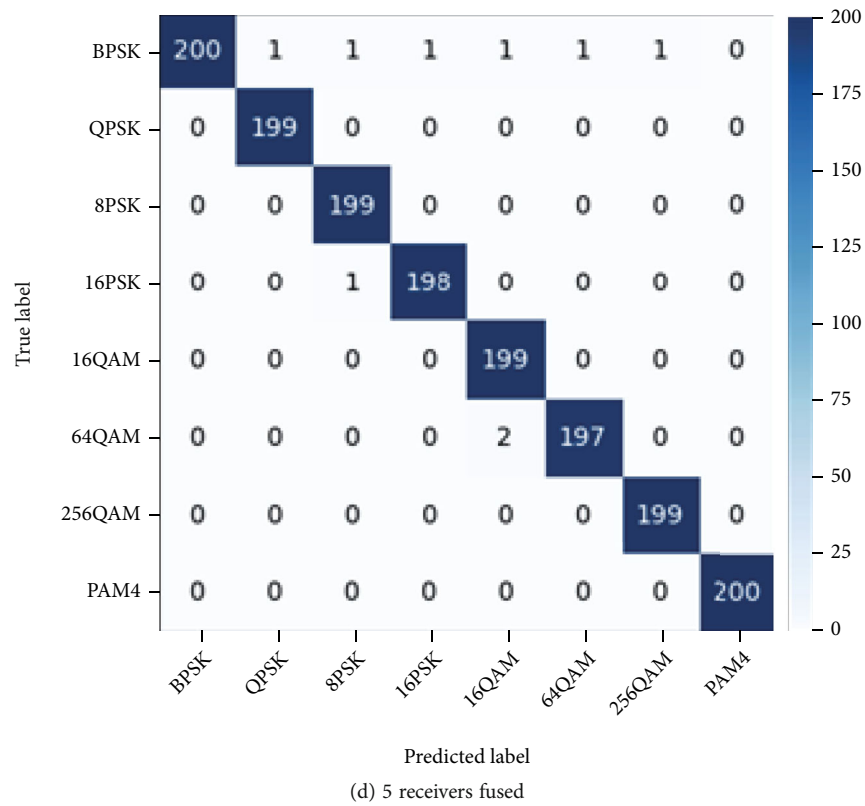
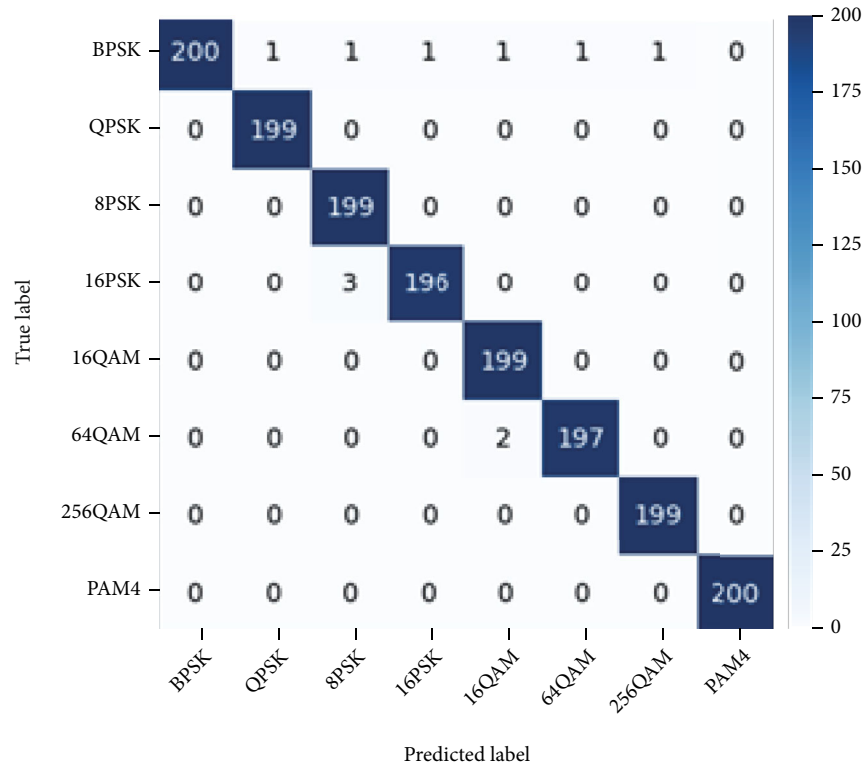


FIGURE 9: Recognition confusion matrix of different receivers under 0 dB. (a) is the recognition effect of receiver 1, receiver 1 is the transmission situation with the worst channel condition, (b) is the recognition effect of receiver 3, and (c) is the recognition effect of receiver 5. Receiver 5 is the best channel transmission situation, and (d) is the effect of collaborative identification by 5 receivers.



TABLE 3: Parameter settings for the new dataset.

Parameter	Dataset 1	Dataset 2
Path delays	[0 1e-4]	[0 1e-5]
AveragePathGains	[1 4]	[-3 3]
MaximumDopplerShift (Hz)	20	10
Phase offset (degree)	[-90 90]	[-90 90]
Frequency offset (Hz)	[0 100]	[0 100]
Fading model	Jakes	Jakes
Rician $K$ -factor	3	3
SNR (dB)	-20:2:20	-20:2:20

(2) Calculate the covariance matrix of the sample

$$\text{Cov} = \frac{1}{k \sum_{i=1}^m (X^i) \cdot (X^i)^T}. \quad (10)$$

Use the singular value decomposition method to calculate the eigenvalues and eigenvectors of the covariance matrix:

$$[U, S, V] = \text{SVD}(\text{Cov}), \quad (11)$$

wherein  $U$  is a dimensionality reduction matrix, which means that all the eigenvectors corresponding to the covariance matrix correspond to the eigenvalues one-to-one, and its dimension is  $m * m$ , and if the first  $d$  column of the matrix is selected, the sample features will be reduced to  $d$  dimensionality.

(3) Dimensionality reduction analysis

All nodes' data samples can be expressed as  $X = [X^1, X^2, \dots, X^k]$ , and the dimensionality reduction feature matrix is obtained according to the rules shown below:

$$Z = X \cdot U_d. \quad (12)$$

Among them, the dimension of  $X$  is  $k \times m$ , and the dimension  $U_d$  is  $m \times d$ ; then, the matrix  $Z$  dimension after dimensionality reduction analysis is  $k \times d$ . The size of the dimensionality reduction error mainly depends on the selection of  $d$ . The larger the value of  $d$ , the more the feature vectors in the representation of  $U$ , which can retain the features of the original features, and the smaller the error, but the redundant features will also be retained, and the amount of calculation will also be reduced. To retain the system's 99% uncertainty, the determination of the  $d$  can refer to

$$\frac{1/k \sum_{i=1}^k \|X^i - X_{map}^i\|^2}{1/k \sum_{i=1}^k \|X^i\|^2} \leq 0.01. \quad (13)$$

Through the above steps, the eigenvalues of the principal components can be determined, and the new feature space

after dimensionality reduction and the new fusion feature vector can be obtained.

The feature vector after the dimensionality reduction process reduces the redundancy of the feature semantics and retains the main semantic information. At this time, the feature vector dimension becomes 30. Input the fused dimensionality reduction feature vector into the classifier for classification. Then, the modulation method of the current sample can be obtained.

In machine learning, the function of the classifier is to judge the class to which a new observation belongs on the basis of the labeled training data. In this paper, after extracting the data features of different nodes through the deep learning model, the classifier can be used to complete the category judgment of the unknown sample data. Commonly used classifiers generally include  $K$ -nearest neighbors (KNN), decision tree classifiers, and support vector machines (SVM). SVM cannot rely on statistical methods, thus simplifying the usual classification and regression problems, and can find key samples that are critical to the task, so this method is used in this paper for the final classification and recognition.

### 3. Results and Discussion

**3.1. Experimental Dataset.** In this section, a number of experiments are carried out to evaluate the performance of this model and algorithm. The dataset used in the experiment is generated by simulation, and its parameters are shown in Table 2. In actual communication scenarios, low-order modulated signal features are easier to extract, but high-order modulated signals are usually used in practice. Recognize modulation modes including confusing modulation mode signals as dataset  $\Phi$ . Each modulation type under each SNR includes 1000 instances, 80% of which are selected as the training set and 20% as the test set.

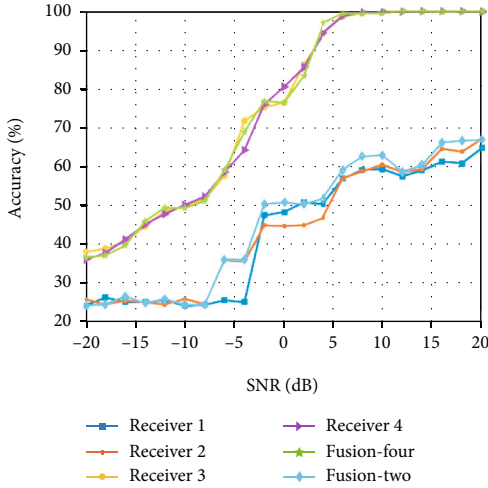
$$\Phi = \{\text{BPSK}, \text{QPSK}, \text{8PSK}, \text{16PSK}, \text{16QAM}, \text{64QAM}, \text{256QAM}, \text{PAM4}\}. \quad (14)$$

#### 3.2. Experimental Settings

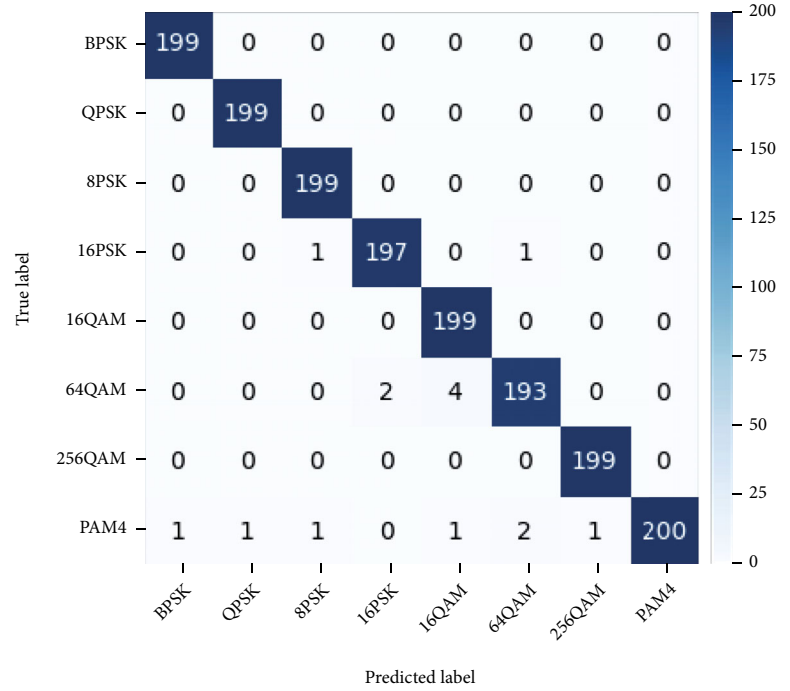
**3.2.1. Comparison of Feature Extraction Network Models.** All experiments in this study are carried out in the TensorFlow framework, and the GPU accelerator used is GeForce RTX 2060. In all simulation experiments, the training model adopts the adaptive moment estimation optimizer, and the learning rate is set to 0.0001 to evaluate the training parameters. In order to prevent overfitting, this paper adds an early stop mechanism during the training process. When the loss function is iterated 30 times and when it is not falling, it can be considered that the model training is completed and tested.

To verify the effectiveness of the feature extraction module in this study, several different network models are trained and tested on the RadioML2016.10a dataset. The test results are shown in Figure 5.

As can be seen from Figure 5(a), on the open-source dataset RadioML2016.10a dataset, the CLDNN network



(a) Recognition accuracy rate change curve



(b) Recognition of fusion-four under 8 dB

FIGURE 10: Recognition performance under different channel conditions. In (a), fusion-four is the variation curve of the cooperative recognition results of four receivers with different channel transmission status with SNR. Fusion-two represents the variation of the cooperative recognition results of receivers 1 and 2 with poor channel conditions.

model (CLDNNnet) network exhibits the best recognition performance. In terms of parameters, it can be seen from Figure 5(b) that ResNet and CLDNNnet have the least amount of network parameters, saving computing resources, and CLDNNnet is better than ResNet in terms of recognition performance. Therefore, considering comprehensively, the subsequent experiments in this paper extract different node features through the CLDNNnet.

**3.2.2. Distributed Collaborative Recognition Results and Analysis.** In the actual sensor network, due to the influence of factors such as the transmission distance and the distribution location of the monitoring nodes, the signal energy and quality at each receiving node are different. In the simulation experiment environment, it is mainly reflected in the different SNR of the received signal of each node. Therefore, in this experiment, each channel is set to be independent, and the signal quality of different receivers under different channel state transmission conditions is simulated with different sizes of Gaussian white noise. The experimental verification is divided into the following two cases:

- Keep the average SNR of each node the same, and test the recognition effect of feature fusion of different numbers of receiver signals
- Node 1 represents the node with the worst channel quality, and its SNR, namely, SNR1 varies from -20 to 12 dB, and the remaining nodes increase by 2 dB

on the basis of node 1, namely,  $\text{SNR}_2 = \text{SNR}_1 + 2$ ,  $\text{SNR}_3 = \text{SNR}_1 + 4$ , ...

Figure 6(a) illustrates the recognition performance of different numbers of receivers under the same average SNR. It can be seen from the figure that with the increase in the number of receivers, the recognition effect after feature fusion is continuously improving. The recognition performance is improved significantly under the low SNRs. In the case of 7-receiver cooperative recognition, the average recognition accuracy of the 8 kinds of modulated signals can reach 81.3% at -4 dB, which is nearly 10% higher than the single-receiver recognition accuracy. This is conducive to realize modulation recognition of weak and poor-quality communication signals in noncooperative communication scenarios. It can be seen from the confusion matrix in Figure 6(b) that the recognition error mainly comes from the confusion of 8PSK and 16PSK, 16QAM, and 64QAM which have similar features.

Under the setting of experiment (b), the variation curve of the cooperative recognition accuracy rate of different numbers of receivers under -20-12 dB is shown in Figure 7. With the increase in the number of receivers, the accuracy rate is continuously improving, and the improvement is obvious when the SNR is low. It can be found from Figure 8 that in the signal transmission process of different channels, under the conditions of low SNR or high SNR, the fusion node identification effect always tends to identify the receiver node with the highest performance. In a

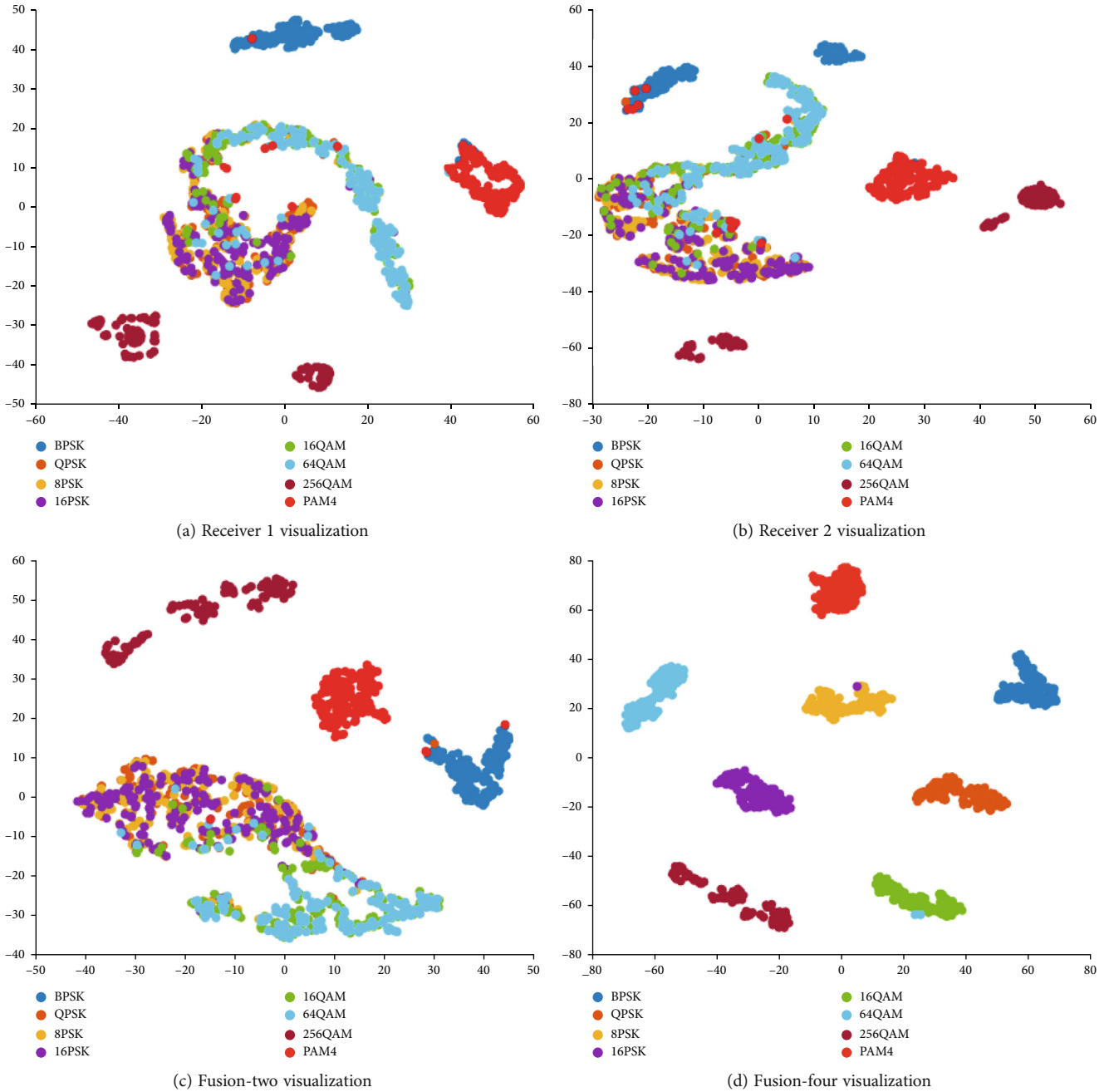


FIGURE 11: Feature cluster visualization.

distributed sensor network, the fusion recognition of the features of different receiver nodes can eliminate the ambiguity of unknown signals to a great extent. When the condition of the transmission channel of a small number of signals deteriorates, it can still maintain a high recognition probability in the end.

In Figure 9, receiver 1 is the transmission with the worst channel conditions. The recognition accuracy rate under 0 dB is only 77.31%. There is serious aliasing between 8PSK and 16PSK and between 16QAM and 64QAM. A good receiver 3 still has a serious aliasing phenomenon between 16QAM and 64QAM. This is because despite the improved

channel conditions, for signals with similar characteristics, the model is still unable to accurately discriminate between confusing samples. In receiver 5, due to good channel conditions and high received signal quality, an average recognition accuracy rate of 99.3% is achieved at 0 dB. In spite of the poor channel conditions, the average recognition accuracy rate of 0 dB still reaches 99.44% when five receivers are cooperatively recognized, and there is basically no recognition confusion. The distributed architecture of the sensor network uses the complementarity of the data features of each receiver to synthesize the extracted features into a feature that is more discriminative than the input feature,

thereby greatly improving the recognition effect in harsh environments without affecting performance of other receivers.

**3.2.3. Comparison of Different Channel Transmission Conditions.** In actual signal transmission on different channels, different magnitudes of frequency offset, phase offset, multipath fading, etc. are often generated. In order to further explore the identification method of distributed multireceiver node feature fusion in different channel environments, on the basis of the above dataset, different degrees of frequency offset, phase offset, and multiple fading delays are added to affect the channel. In order to facilitate comparison, experiments are carried out by adding noise of the same magnitude to each node. The parameters of the further generated dataset are shown in Table 3.

Datasets 1 and 2 are generated by simulation in Rayleigh fading channels. By adding different degrees of fading coefficients and delays, the situation of receivers placed at different distances from the transmitter is further simulated, representing the signals at both ends of receivers 1 and 2, respectively. In addition, two datasets with only frequency offset or phase offset are set as the channel simulation with better transmission status, receiver 3 and receiver 4, respectively.

Although there are a few cases of severe channel deterioration in the distributed system architecture, the recognition performance has been improved after fusion, and the result after fusion always tends to go to the node with the best performance. Even compared to the best receiving node, there is still a certain degree of improvement. As shown in Figure 10, the recognition performance of receiver 1 and receiver 2 is very unsatisfactory compared with receivers 3 and 4, but the fusion effect is significantly improved.

The extracted features of different receiver signals are visualized after dimensionality reduction, and the effect of feature extraction can be further observed. The effect of data feature visualization under 8 dB is shown in Figure 11. Because the transmitter signal is transmitted in a complex channel environment, it is affected by a variety of factors, resulting in poor signal quality received by receivers 1 and 2 and difficulty in feature extraction. In Figures 11(a) and 11(b), it can be found that the signal features of the two have serious aliasing phenomenon, and the clustering effect is poor. It is difficult to obtain accurate feature expressions, resulting in poor classification and recognition performance. When the features of the two nodes are fused, there is still serious aliasing, and it is difficult to achieve better recognition results, as shown in Figure 11(c). But when the signal features of the receiving end with better transmission are added, as shown in Figure 11(d), there is basically no aliasing. The signal features of each modulation type can be clearly distinguished, and the signal features of the same modulation are strongly aggregated, so that better recognition results can be achieved. This experiment further illustrates that although the distributed collaborative recognition framework based on feature fusion can improve the signal recognition performance to a certain extent, when the signal transmission channels are severely

deteriorated, the recognition performance will not be greatly improved.

## 4. Conclusions

Aiming at the problem that it is difficult to accurately extract signal features and perception signal semantic information from complex channel transmission in noncooperative communication scenarios, this paper uses the excellent feature extraction performance of DL algorithms to propose a recognition architecture for multisensor distributed cooperative sensing spectrum semantics based on feature fusion. The distributed wireless sensor network monitors the electromagnetic spectrum to realize the semantic information of wireless communication signals. The simulation experiment proves that the distributed collaborative sensing and recognition architecture in this paper can solve the problems of single-node signal transmission, which is difficult to accurately analyze the spectrum semantic information under complex channel conditions, poor adaptability to the environment, and low recognition performance. In particular, in the confrontation scenario of noncooperative communication, the algorithm in this paper is more conducive to the feature extraction of weak signals and realizes spectral semantic perception and recognition. However, this solution still cannot guarantee a good recognition effect in a more complex communication signal transmission environment with higher recognition accuracy requirements. Since the datasets in this experiment are all generated under simulation conditions, in order to further verify the effectiveness of the algorithm, the next research will be applied to the perception recognition of the measured datasets on this basis. In addition, this paper realizes the fusion scheme based on the feature layer. How to coordinate the recognition of the data layer, feature layer, and decision layer without increasing the data computing load is an important research direction.

## Data Availability

The RadioML2016.10a used in this paper is an open-source dataset in Reference [6]. Other datasets used in this article can be obtained by contacting the email zhangjing03@hr-beu.edu.cn.

## Conflicts of Interest

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

## Acknowledgments

This work is supported by the National Natural Science Foundation of China (grant number 62001137), the Natural Science Foundation of Heilongjiang Province (grant number JJ2019LH2398), and the Foundation of Key Laboratory of Signal and Information System of Shandong Province.



## References

- [1] J. L. Xu, W. Su, and M. Zhou, "Likelihood-ratio approaches to automatic modulation classification," *IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews)*, vol. 41, no. 4, pp. 455–469, 2011.
- [2] M. Mansouri, R. Baklouti, M. F. Harkat, M. Nounou, H. Nounou, and A. B. Hamida, "Kernel generalized likelihood ratio test for fault detection of biological systems," *IEEE Transactions on Nanobioscience*, vol. 17, no. 4, pp. 498–506, 2018.
- [3] A. Hazza, M. Shoaib, and S. A. Alshebeili, "An overview of feature-based methods for digital modulation classification," in *2013 1st international conference on communications, signal processing, and their applications (ICCSPA)*, pp. 1–6, Sharjah, United Arab Emirates, 2013.
- [4] M. L. D. Wong and A. K. Nandi, "Automatic digital modulation recognition using spectral and statistical features with multi-layer perceptrons," in *Proceedings of the Sixth International Symposium on Signal Processing and its Applications*, pp. 390–393, Kuala Lumpur, Malaysia, 2001.
- [5] C. Hou, Y. Li, X. Chen, and J. Zhang, "Automatic modulation classification using KELM with joint features of CNN and LBP," *Physical Communication*, vol. 45, no. 3, article 101259, 2021.
- [6] T. J. O'shea and N. West, "Radio machine learning dataset generation with gnu radio," *Proceedings of the GNU Radio Conference*, vol. 1, no. 1, 2016.
- [7] Y. Wu, X. Li, and J. Fang, "A deep learning approach for modulation recognition via exploiting temporal correlations," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications*, pp. 1–5, Kalamata, Greece, 2018.
- [8] H. Wu, Y. Li, L. Zhou, and J. Meng, "Convolutional neural network and multi-feature fusion for automatic modulation classification," *Electronics Letters*, vol. 55, no. 16, pp. 895–897, 2019.
- [9] Y. Lin, Y. Tu, and Z. Dou, "An improved neural network pruning technology for automatic modulation classification in edge devices," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5703–5706, 2020.
- [10] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *International conference on engineering applications of neural networks*, pp. 213–226, Cham, 2016.
- [11] Y. Lin, M. Wang, X. Zhou, G. Ding, and S. Mao, "Dynamic spectrum interaction of UAV flight formation communication with priority: a deep reinforcement learning approach," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 3, pp. 892–903, 2020.
- [12] S. Ramjee, S. Ju, D. Yang, X. Liu, A. E. Gamal, and Y. C. Eldar, "Fast deep learning for automatic modulation classification," 2019, <http://arxiv.org/abs/1901.05850>.
- [13] Y. Tu, Y. Lin, J. Wang, and J. U. Kim, "Semi-supervised learning with generative adversarial networks on digital signal modulation classification," *CMC-Computers Materials & Continua*, vol. 55, no. 2, pp. 243–254, 2018.
- [14] C. Hou, X. Zhang, and X. Chen, "Electromagnetic signal feature fusion and recognition based on multi-modal deep learning," *International Journal of Performability Engineering*, vol. 16, no. 6, pp. 941–949, 2020.
- [15] X. Li, F. Dong, S. Zhang, and W. Guo, "A survey on deep learning techniques in wireless signal recognition," *Wireless Communications and Mobile Computing*, vol. 2019, 12 pages, 2019.
- [16] Q. Cheng, Y. Yang, and X. Gui, "Disturbance signal recognition using convolutional neural network for DAS system," in *13th international conference on measuring technology and mechatronics automation (ICMTMA)*, pp. 278–281, Beihai, China, 2021.
- [17] Y. Zhang, N. Ansari, and W. Su, "Multi-sensor signal fusion based modulation classification by using wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 15, no. 12, pp. 1621–1632, 2015.
- [18] B. Dulek, "An online and distributed approach for modulation classification using wireless sensor networks," *IEEE Sensors Journal*, vol. 17, no. 6, pp. 1781–1787, 2017.
- [19] C. Lyu, Z. Huo, X. Cheng, J. Jiang, A. Alimasi, and H. Liu, "Distributed optical fiber sensing intrusion pattern recognition based on GAF and CNN," *Journal of Lightwave Technology*, vol. 38, no. 15, pp. 4174–4182, 2020.
- [20] J. Li, B. Lu, Y. Wang, X. Liu, Q. Bai, and B. Jin, "Distributed optical fiber vibration sensor for the identification of pipeline leakage using relevant vector machine," in *Optics Frontiers Online 2020: Distributed Optical Fiber Sensing Technology and Applications*, vol. 11607, p. 116070J, Virtual, Online, China, 2021.
- [21] Z. Sun, K. Liu, J. Jiang et al., "Optical fiber distributed vibration sensing using grayscale image and multi-class deep learning framework for multi-event recognition," *IEEE Sensors Journal*, vol. 21, no. 17, pp. 19112–19120, 2021.
- [22] T. Wimalajeewa, J. Jagannath, P. K. Varshney, A. Drozd, and W. Su, "Distributed asynchronous modulation classification based on hybrid maximum likelihood approach," in *MILCOM 2015-2015 IEEE Military Communications Conference*, pp. 1519–1523, Tampa, FL, United States, 2015.
- [23] J. L. Xu, W. Su, and M. Zhou, "Distributed automatic modulation classification with multiple sensors," *IEEE Sensor Journal*, vol. 10, no. 11, 2010.
- [24] B. Dulek, O. Ozdemir, P. K. Varshney, and W. Su, "Distributed maximum likelihood classification of linear modulations over nonidentical flat block-fading Gaussian channels," *IEEE Transactions on Wireless Communications*, vol. 14, no. 2, pp. 724–737, 2015.
- [25] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4580–4584, Brisbane, QLD, Australia, 2015.

## Research Article

# A Load-Aware Multistripe Concurrent Update Scheme in Erasure-Coded Storage System

Junqi Chen <sup>1,2</sup>, Yong Wang <sup>1,2</sup>, Miao Ye <sup>3,4</sup>, Qinghao Zhang <sup>3</sup> and Wenlong Ke <sup>3,4</sup>

<sup>1</sup>School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, China

<sup>2</sup>Guangxi Engineering Technology Research Center of Cloud Security and Cloud Service, Guilin University of Electronic Technology, Guilin 541004, China

<sup>3</sup>School of Information and Communication, Guilin University of Electronic Technology, Guilin 541004, China

<sup>4</sup>Ministry of Education Key Lab. of Cognitive Radio and Information Processing, Guilin University of Electronic Technology, Guilin 541004, China

Correspondence should be addressed to Yong Wang; [ywang@guet.edu.cn](mailto:ywang@guet.edu.cn)

Received 24 January 2022; Accepted 7 May 2022; Published 19 May 2022

Academic Editor: Ashish Bagwari

Copyright © 2022 Junqi Chen 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.

Erasure coding has been widely deployed in today's data centers for it can significantly reduce extra storage costs while providing high storage reliability. However, erasure coding introduced more network traffic and computational overhead in the data update process. How to improve the efficiency and mitigate the system imbalance during the update process in erasure coding is still a challenging problem. Recently, most of the existing update schemes of erasure codes only focused on the single stripe update scenario and ignored the heterogeneity of the node and network status which cannot sufficiently deal with the problems of low update efficiency and load imbalance caused by the multistripe concurrent update. To solve this problem, this paper proposes a Load-Aware Multistripe concurrent Update (LAMU) scheme in erasure-coded storage systems. Notably, LAMU introduces the Software-Defined Network (SDN) mechanism to measure the node loads and network status in real time. It selects nonduplicated nodes with better performance such as CPU utilization, remaining memory, and I/O load as the computing nodes for multiple update stripes. Then, a multiattribute decision-making method is used to schedule the network traffic generated in the update process. This mechanism can improve the transmission efficiency of update traffic and make LAMU adapt to the multistripe concurrent update scenarios in heterogeneous network environments. Finally, we designed a prototype system of multistripe concurrent updates. The extensive experimental results show that LAMU could improve the update efficiency and provide better system load-balancing performance.

## 1. Introduction

The scale of the distributed storage system is rapidly expanding to deal with the proliferation of the global datasphere. Meanwhile, the node failures and data loss caused by various reasons are increasing, such as system crashes, natural disasters, hacker attack, and power outages [1–3]. To avoid irreversible losses caused by these threats and improve the reliability of the storage system, the redundancy mechanism is indispensable in data centers. The two most typical redundancy mechanisms are *replications* and *erasure coding*. Rep-

lications copy each chunk of original data to other storage devices to improve the system redundancy. However, it considerably incurs extra storage costs, especially in today's data scale explosion and growth. As another alternative, erasure coding can provide better storage efficiency via encoding computations, meeting the same degree of fault tolerance as replications [4]. Specifically, erasure coding divides the original data into several data chunks, and then, these data chunks are encoded into a few redundant chunks (also called parity chunks). These data chunks and parity chunks together form an erasure-coding stripe. When data failure occurs, as long



as the number of failed chunks does not exceed the recovery threshold, the lost chunks can still be recovered from the living chunks. Since the erasure coding can significantly reduce the extra storage cost while providing high storage reliability, it has been widely deployed in today's data centers, such as Facebook [5], Azure [6], and Google GFS [7].

However, while providing high reliability with less extra storage cost, erasure coding introduces more network traffic and computation overhead during the data update process. When the data chunk is updated, all the parity chunks in the same stripe should be updated simultaneously to maintain the consistency of the stripe, which boosts the disk I/O load and the update time. In addition, various real trace analyses show that over 90% of writing in the storage system is data update [8–10], indicating that data update is prevalent. If the data failure occurs during the update process, the system cannot recover the failure data correctly. Therefore, the update efficiency of erasure coding affects not only the performance but also the reliability of the distributed storage system.

There are two major challenging factors impacting the erasure-coding update. Challenge 1 is the heterogeneity of the storage node and network status. For example, the storage nodes purchased in different periods during the expansion of storage system have different performance [11, 12]. Meanwhile, these storage nodes may also be processing various tasks in real time, such as MapReduce [13] and system heartbeat and data migration [14], making the status of network links dynamic and heterogeneous. In this case, the computational load and traffic caused by the update may significantly impact the system performance and reduce the update efficiency. Challenge 2 is the multistripe concurrent update. Due to the potential correlation between the data of each stripe [15, 16], the update of one erasure-coding stripe will result in the contemporary update of multiple correlation stripes [15], which amplifies the node load and the update time. Therefore, how to improve the update efficiency of erasure code storage and guarantee the system load balance is still a critical problem. However, the existing update scheme ignores the node and network heterogeneity and only focuses on the single stripe update scenario, which cannot sufficiently deal with the problems of update efficiency declines and system load imbalance caused by the multistripe concurrent update.

This paper proposed a Load-Aware Multistripe concurrent Update (LAMU) scheme. As we will explain in Section 3, LAMU adopts a centralized update architecture in which the data update is divided into the data-delta convergence, parity-delta computation, and parity-delta divergence. The centralized update architecture can mitigate the system overhead by preventing the separate connection between the data node and the parity node. Firstly, we introduce Software-Defined Networking (SDN) to measure and collect node load information (such as CPU utilization, residual memory, disk I/O load, and node access bandwidth) and network status (such as network topology, link residual bandwidth, and link transmission delay) in real time. Secondly, based on the node load information, we select the nonrepetitive computing nodes with a lower load for each update stripe. Finally, the TOP-

SIS method is used to tailor the best path for data-delta convergence and parity-delta divergence for each update stripe. The decision-making factor uses diverse weights for different network load scenarios so that LAMU could be suitable for various environments.

The main contributions of this paper can be summarized as follows:

- (1) Aiming to solve the problem that existing research cannot sufficiently deal with the efficiency decline of multistripe concurrent updates, this paper first establishes the optimization model of multistripe updates with multiple QoS constraints in the heterogeneous environment. The update efficiency can be improved by minimizing the cumulative weighted update delay of multistripe updates and balancing the link utilization. To the best of our knowledge, this is the first work attempt to improve the efficiency of multistripe concurrent updates with multiple QoS constraints
- (2) This paper introduces SDN to perceive the node load status and network status of the erasure-coded storage system in real time and proposes a Load-Aware Multistripe concurrent Update (LAMU) scheme. LAMU selected the nonrepetitive computing nodes with better capacity for each update stripe. Then, the TOPSIS method is used to schedule the update traffic generated in the update process to improve the efficiency of multistripe updates. As far as we know, this is the first work that considers the heterogeneity of nodes and network status simultaneously during the erasure-coding update process
- (3) We designed a prototype system of multistripe concurrent updates based on Containernet [17] to verify the effectiveness of LAMU. The extensive experimental results show that LAMU could improve the erasure-coding update efficiency and maintain better system load balancing

The rest of this paper is organized as follows: Section 2 presents the background and related work of the erasure-coding update. Section 3 describes the multistripe update problem in the erasure-coded system and provides the optimization model. Section 4 introduces the details of our LAMU scheme. We conduct extensive experiments to evaluate LAMU in Section 5. Section 6 concludes this paper.

## 2. Background and Related Work

*2.1. Basics of Erasure Coding.* In this paper, we concentrate on a well-known erasure code called the Reed-Solomon (RS) codes [18], which are widely used in today's commercial data centers [7]. To be precise, the system configures the RS codes with two parameters  $k$  and  $r$  and denote the code by RS  $(k+r, k)$  codes. In RS  $(k+r, r)$  codes, the original data  $D$  are divided into  $k$  data chunks  $\{d_1, d_2, \dots, d_k\}$ , and these  $k$  data chunks are encoded to  $r$  parity chunk  $s \{p_1, p_2, \dots, p_r\}$  through the linear operation of equation (1).

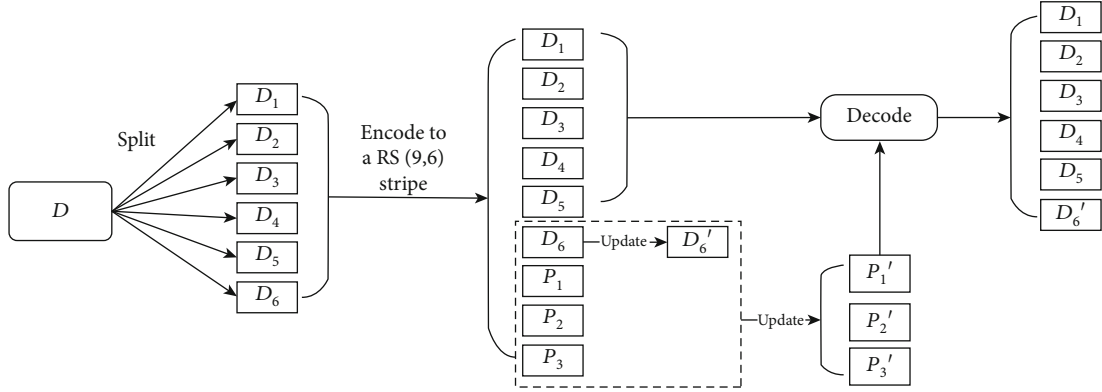


FIGURE 1: Example of RS (9, 6).

These  $k + r$  chunks distributed in different nodes of the storage system form an erasure code stripe  $S$ .

$$p_i = \sum_{j=1}^k c_{i,j} * d_j, \quad (1)$$

where  $c_{i,j}$  is the conversion coefficient from  $d_j$  to  $p_i$ ,  $1 \leq i \leq r$ ,  $1 \leq j \leq k$ . According to the linear characteristics of equation (1), as long as the number of surviving chunks in the stripe is larger than  $k$ , any  $k$  chunks can reconstruct the whole stripe.

Figure 1 depicts the process of encoding, updating, and decoding of RS (9, 6). First, the system divides the original data  $D$  into 6 data chunks  $\{D_1, D_2, D_3, D_4, D_5, D_6\}$ , and the data chunks are encoded by equation (1) to get 3 parity chunks  $\{P_1, P_2, P_3\}$ ; these 9 chunks form an erasure code stripe. When the data chunk  $D_6$  is updated to  $D_6'$ , the 3 parity chunks will be synchronously updated to  $\{P_1', P_2', P_3'\}$ . In decoding, through the linear operation of equation (1), the whole stripe can be reconstructed from any 6 surviving chunks (such as  $\{D_1, D_2, D_3, D_4, D_5, P_1'\}$ ).

As we can see from Figure 1, in the data update process of erasure coding, when the data chunk is updated, all the parity chunks in the same stripe must also be updated simultaneously to maintain the consistency of the stripe. Based on whether the complete data chunks need to be transmitted, there are 2 classes of update framework: *full-stripe* update and *delta-based* update. In the full-stripe update, the data chunk  $D_j$  (where  $1 \leq j \leq k$ ) is updated to  $D_j'$ , and then, equation (1) is used to calculate the new parity chunk  $P_i'$  (where  $1 \leq i \leq r$ ), which needs to transmit the whole chunks and consumes significantly large network resources. In the delta-based update, we can update each parity chunk  $P_i$  into  $P_i'$  by equations (2a) and (2b):

$$\Delta P_i = \sum_{j=1}^m c_{i,j} * (D_j' - D_j), \quad (2a)$$

$$P_i' = P_i + \Delta P_i, \quad (2b)$$

$$1 \leq i \leq r, 1 \leq j \leq m. \quad (2c)$$

To elaborate further, when  $m$  data chunks  $\{D_1 \dots D_j \dots D_m\}$  are updated to  $\{D_1' \dots D_j' \dots D_m'\}$ , each data chunk sends the data-delta  $D_j' - D_j$  to the computing node, which calculates  $\Delta P_i$  based on equation (2a) and then distributes  $\Delta P_i$  to parity nodes to complete the whole update process. Clearly, the delta-based update can save network resources and improve the update efficiency compared with the full-stripe update.

**2.2. Related Work.** As mentioned above, the data update is prevalent in the storage system. It has a significant impact on the performance of the distributed storage system. Therefore, various update schemes have been proposed to improve the erasure-coding update efficiency in recent years. T-Update [19] builds the minimum update tree using the Prim [20] algorithm to deal with the single-node update problem, but T-Update neglects the network status during construction of the update topology, which is prone to cause network congestion when the system load is high. TA-Update [21] adds a rollback-based failure handle method based on T-Update, making the update process more adaptive. To cope with the problem of multiple-node update in erasure coding, PUM-P [22] first proposed the centralized update architecture that collects the data-delta  $\Delta d$  to the middle node close to the data nodes and distributes the  $\Delta p$  by the random choice route path. Although the PUM-P can reduce the connection number between the data node and the parity node, PUM-P ignores the heterogeneity of nodes when selecting the compute node and ignores the link status when scheduling the update traffic. In order to improve the data transmission efficiency of multiple-node updates, ACOUS [23] constructs an update tree that considers the link delay provided by the commercial cloud service provider, which reduces the multiple-node update time. But ACOUS also neglects the heterogeneity of nodes when selecting the compute node, and it is difficult to obtain the delay parameter from the service provider in common storage clusters.

The work mentioned above improves the update efficiency by optimizing the data update process. Shen et al. [15] proposed the CASO that solves this problem by organizing data chunks with high correlation into the same

stripes to reduce the update traffic. Specifically, CASO mines the correlation of different stripes from the real storage system work trace [16] and then organizes the highly correlated data into the same stripe to reduce the number of concurrent update stripes and improve the update efficiency. CASO can only mitigate the correlation between stripes, but it cannot entirely eliminate the correlation of stripes. Consequently, the multistripe concurrent updates are still frequently triggered by association stripes, especially in the storage system that the stripe is organized without consideration for data correlations. Therefore, improving the multistripe concurrent update efficiency and maintaining the system load balance are still very challenging tasks.

In summary, most of the existing update schemes of erasure codes only focus on the single stripe update scenario and ignore the heterogeneity of the node and network status, which cannot sufficiently deal with the problems of low update efficiency and load imbalance caused by the multistripe concurrent update. To solve these problems, this paper introduces SDN and multiattribute decision-making methods and proposes the Load-Aware Multistripe concurrent Update (LAMU) scheme in heterogeneous erasure-coded storage systems.

### 3. Model and Formulation of the Multistripe Concurrent Update Problem

In this section, we first state the multistripe concurrent update problem in the erasure-coded system. Our motivation is to find the best computing nodes, convergence path, and divergence path for each strip. These computing nodes and route paths are combined to form an update forest. Then, we give the optimization model of multistripe updates with multiple QoS constraints in the heterogeneous environment.

*3.1. Problem Statement.* Figure 2 shows a simple distributed erasure-coded storage system including several racks, in which each rack contains multiple storage nodes and each node stores many chunks from diverse erasure-coding stripes. For example,  $S_1d_1$  denotes the first data chunk of stripe  $S_1$  and  $S_1p_1$  denotes the first parity chunk of stripe  $S_1$ . We can see from Figure 2 that the 4 data chunks and 4 parity chunks have been distributed in different racks or nodes in the system.

We use the centralized update architecture similar to PUM-P [22], which reduces the connection between the data and parity nodes by introducing middle computing nodes. We take the update process of 4 data chunks in stripe  $S_1$  described in Figure 2 as an example: Firstly, stripe  $S_1$  updates data chunks  $S_1d_1, S_1d_2, S_1d_3, S_1d_4$  to  $S_1d_1', S_1d_2', S_1d_3', S_1d_4'$  and then converges the data-delta  $S_1\Delta d_j = (S_1d_j' - S_1d_j)$  to the computing node selected by the controller. Secondly, the computing node calculates the parity-delta  $\Delta P_i$  by equation (2a). Finally, the computing node distributes the  $\Delta P_i$  to corresponding parity nodes.

The detailed mathematical model of multistripe concurrent updates with multiple QoS constraints in a heterogeneous environment is introduced in Section 3.2. The

network topology of an erasure-coded storage system can be modeled by a graph  $G(V, E)$ , in which  $V$  presents the set of switches and  $E$  denotes the set of links between adjacent switches. For easy reference, the notations used in this section are shown in Table 1.

#### 3.2. Problem Formulation

*3.2.1. Cumulative Weighted Update Delay for Multistripe Update.* The first objective function is aimed at minimizing the cumulative update delay of  $M$  stripes defined as  $f_1$  in equation (3a). Specifically, the update delay of stripe  $m$  is defined as  $d_{\text{update}}^m$  in equation (3b), which is composed of (a) the data-delta convergence delay  $d_{\text{convergence}}^m$ , (b) the parity-delta compute delay  $d_{\text{compute}}^m$ , and (c) the parity-delta divergence delay  $d_{\text{divergence}}^m$ .

$$\min f_1 = \sum_{m \in M} d_{\text{update}}^m, \quad (3a)$$

$$d_{\text{update}}^m = d_{\text{convergence}}^m + d_{\text{compute}}^m + d_{\text{divergence}}^m. \quad (3b)$$

(1) *The Data-Delta Convergence Delay.* The data-delta convergence delay  $d_{\text{convergence}}^m$  is formulated as follows:

$$d_{\text{convergence}}^m = \sum_{i \in N_m} \max_{p \in i} \{d_{mp}\} x_{mi}, \quad (4a)$$

$$\text{s.t. } \sum_{m \in M} b_m x_{mi} \leq \min_{e \in i} \{b_e\}, \quad \forall i \in N_m, \quad (4b)$$

$$\sum_{i \in N_m} x_{mi} = 1, \quad \forall m \in M, \quad (4c)$$

$$x_{mi} \in \{0, 1\}, \quad \forall m \in M, \forall i \in N_m, \quad (4d)$$

where  $N_m$  denotes the set of all possible convergence paths from the updated data nodes to the computing node of stripe  $m$ .  $i$  is an element in the set  $N_m$ , and it is composed of multiple point-to-point path  $p$  from data nodes to computing node.  $\max_{p \in i} \{d_{mp}\}$  denotes the convergence delay of

stripe  $m$  selecting  $i$  as the convergence path. The constraint in formula (4b) ensures that the total bandwidth requirement for all convergence paths through path  $i$  does not exceed the bottleneck bandwidth. The constraint (4c) is met to ensure that only one divergence path will be assigned to stripe  $m$ . In constraint (4d),  $x_{mi}$  denotes a binary variable: it is 1 if stripe  $m$  selects  $i$  as the convergence path and 0 otherwise.

(2) *The Parity-Delta Computing Delay.* This section adopts the definition of node computing capacity in the erasure-

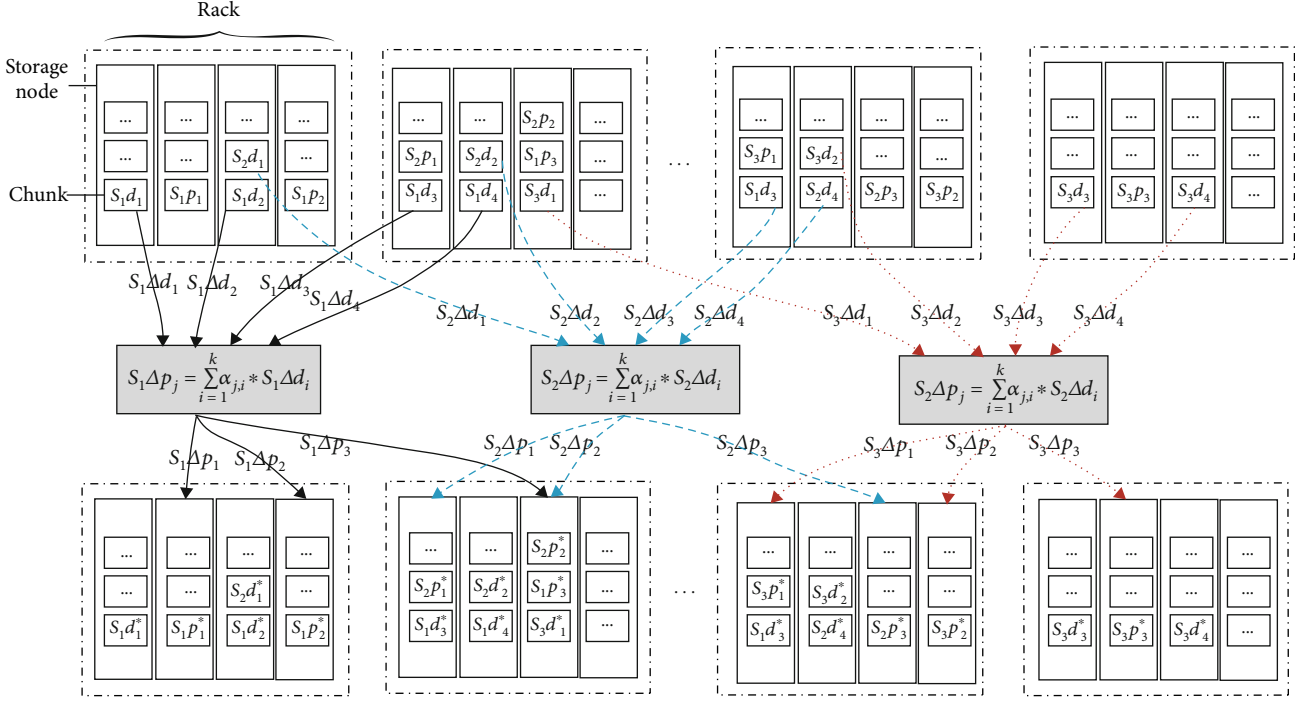


FIGURE 2: Example of multistripe concurrent update in erasure-coded storage system.

TABLE 1: Symbols in the problem formulation.

Symbol	Description	Symbol	Description
$G$	Network topology	$E$	Link set
$V$	Switch set	$M$	Number of update stripes
$m$	Update stripe index	$e$	Link between two adjacent switches
$N_m$	Convergence paths set of update stripe $m$	$L_m$	Divergence paths set of update stripe $m$
$i$	Convergence path	$j$	Divergence path
$p$	Unicast path in convergence path $i$	$q$	Unicast path in divergence path $j$
$d_{mp}$	Convergence delay of stripe $m$ via path $p$	$d_{mq}$	Divergence delay of stripe $m$ via path $q$
$b_m$	Bandwidth cost of update stripe $m$	$b_e$	Bandwidth capacity of link $e$
$x_{mi}$	Binary variable	$x_{mj}$	Binary variable
$N$	Number of node load decision factors	$D_m$	Update data volume of stripe $m$

coded system proposed by Fenglin et al. [24]. It uses the sequence  $c_1, c_2, \dots, c_k, \dots, c_n$  to denote the factors which affect the processing ability of a node, such as CPU utilization, remaining memory, and disk I/O, and the corresponding weight factors are  $\varphi_1, \varphi_2, \dots, \varphi_k, \dots, \varphi_n$ . Therefore, the computing capacity of each node in the erasure-coding update can be expressed as

$$\text{Capacity}_m = \sum_{k=1}^N c_k \varphi_k. \quad (5a)$$

It is assumed that  $D_m$  represents the update volume of stripe  $m$ ; the parity-delta computing delay  $d_{\text{compute}}^m$  is formulated as follows:

$$d_{\text{compute}}^m = \gamma \frac{D_m}{\text{Capacity}_m} = \gamma \frac{D_m}{\sum_{k=1}^N c_k \varphi_k}, \quad (5b)$$

where  $\gamma$  indicates the capacity conversion coefficient.

(3) *The Parity-Delta Divergence Delay.*

$$d_{\text{divergence}}^m = \sum_{j \in L_m} \max_{q \in j} \{d_{mq}\} x_{mj}, \quad (6a)$$

$$\text{s.t. } \sum_{m \in M} b_m x_{mj} \leq \min \{b_e\}, \quad \forall j \in L_m, \quad (6b)$$

$$\sum_{j \in L_m} x_{mj} = 1, \quad \forall m \in M, \quad (6c)$$

$$x_{mj} \in \{0, 1\}, \quad \forall m \in M, \forall j \in L_m, \quad (6d)$$

where  $L_m$  denotes the set of all possible divergence paths from the computing node to the parity nodes of stripe  $m$ .  $j$  is an element in the set  $L_m$ , and it is composed of multiple point-to-point path  $q$  from the computing node to parity nodes.  $\max_{q \in j} \{d_{mq}\}$  denotes the divergence delay of stripe  $m$  selecting  $j$  as the divergence path. The constraint in formula (6b) ensures that the total bandwidth requirement for all convergence paths through path  $j$  does not exceed the bottleneck bandwidth. The constraint (6c) is met to ensure that only one divergence path will be assigned to the stripe  $m$ . In constraint (6d),  $x_{mj}$  denotes a binary variable: it is 1 if stripe  $m$  selects  $j$  as the convergence path and 0 otherwise.

(4) *The Proposed Objective Function.* According to formulas (3)–(6), the objective function of the cumulative weighted delay of the multistripe update can represent as

$$\min f_1 = \sum_{m \in M} \left( \sum_{i \in N_m} \max_{p \in i} \{d_{mp}\} x_{mi} + \gamma \frac{D_m}{\sum_{k=1}^N c_k \varphi_k} + \sum_{j \in L_m} \max_{q \in j} \{d_{mq}\} x_{mj} \right). \quad (7)$$

3.2.2. *Network Load-Balancing Performance for Multistripe Update.* While improving the update efficiency, the load balance of the network is also critical. The objective function of minimizing the maximum link bandwidth utilization is defined as follows:

$$\min f_2 = \max_{e \in E} \sum_{m \in M} \frac{b_m x_{me}}{b_e}, \quad (8a)$$

$$\text{s.t. } \sum_{m \in M} b_m x_{me} \leq b_e, \quad \forall e \in E, \quad (8b)$$

$$x_{me} \in \{0, 1\}, \quad \forall m \in M, \forall e \in E. \quad (8c)$$

The constraint (8b) ensures that the used bandwidth of the link  $e$  cannot be in excess of the link capacity;  $x_{me}$  is a binary variable for the link selection of the update traffic.

3.3. *The Proposed Multiobjective Optimal Model of Multistripe Update.* Our goal is to minimize the cumulative weighted delay of multistripe updates denoted as  $f_1$  and

minimize the maximum link bandwidth utilization represented as  $f_2$ . However, it is difficult to achieve the minimum values of  $f_1$  and  $f_2$  at the same time. The overall objective function of this paper is defined as follows:

$$\text{minimize } Z = [f_1, f_2], \quad (9a)$$

$$\text{s.t. } \sum_{m \in M} b_m x_{mi} \leq \min_{e \in i} \{b_e\}, \quad \forall i \in N_m, \quad (9b)$$

$$\sum_{m \in M} b_m x_{mj} \leq \min_{e \in j} \{b_e\}, \quad \forall j \in L_m, \quad (9c)$$

$$\sum_{m \in M} b_m x_{me} \leq b_e, \quad \forall e \in E, \quad (9d)$$

$$\sum_{i \in N_m} x_{mi} = 1, \quad \forall m \in M, \quad (9e)$$

$$\sum_{j \in L_m} x_{mj} = 1, \quad \forall m \in M, \quad (9f)$$

$$x_{mi} \in \{0, 1\}, \quad \forall m \in M, \forall i \in N_m, \quad (9g)$$

$$x_{mj} \in \{0, 1\}, \quad \forall m \in M, \forall j \in L_m, \quad (9h)$$

$$x_{me} \in \{0, 1\}, \quad \forall m \in M, \forall e \in E. \quad (9i)$$

## 4. SDN-Based Load-Aware Multistripe Concurrent Update Scheme

To solve the objective functions (9), we propose the LAMU scheme. Figure 3 presents the system architecture of LAMU, which includes four main modules: the Node Monitor (NodeM) module, Network Monitor (NetM) module, Compute Node Selection (CNS) module, and Path Selection (PS) module. In the process of LAMU, seeking the best computing node and transmission path for the multistripe erasure code data update is briefly described as follows: Firstly, the NodeM and NetM modules update the real-time node load information and network information. Then, the CNS module is used to select the computing nodes for update stripes according to the network and node status. Last, LAMU employs the PS module to find an appropriate convergence path between data nodes and the compute node and an appropriate divergence path between the compute node and parity nodes. The combination of the computing node, convergence path, and divergence path forms the update tree. Multiple update trees constitute an update forest.



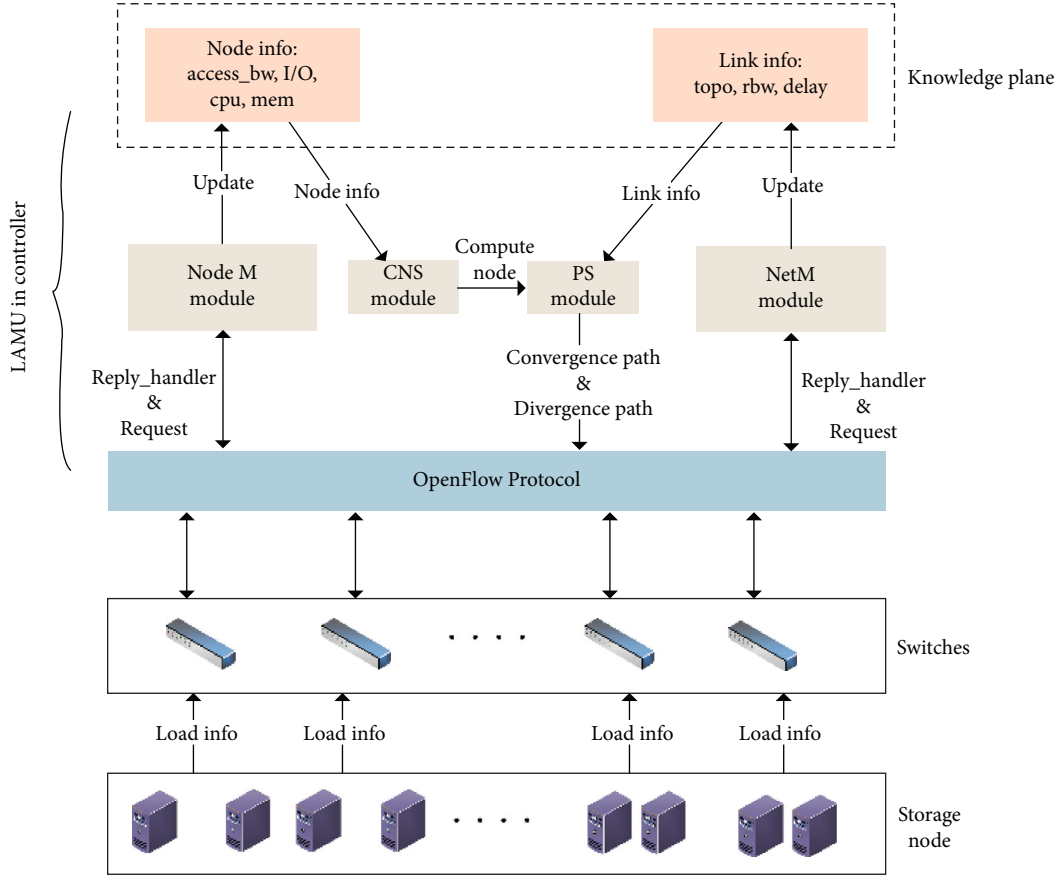


FIGURE 3: The architecture of LAMU.

4.1. *The NodeM Module and NetM Module.* The Software-Defined Network (SDN) can significantly simplify the network configuration and alleviate the measurement overhead compared with the traditional network architecture. For example, SDN can provide a flexible and efficient monitor strategy through the centralized control plane. In the LAMU scheme, the NodeM and NetM modules interact with switches through the OpenFlow protocol of SDN to discover the global network topology. It updates the load information of the storage node and the network link status in real time to provide a knowledge plane for LAMU. The node load information recorded by the NodeM is as follows:

$$C_n = \{c_1, c_2, \dots, c_k\}, \quad (10)$$

$$M_n = \{m_1, m_2, \dots, m_k\}, \quad (11)$$

where  $C_n$  denotes the set of CPU utilization of each storage node and  $M_n$  denotes the set of the residual memory capacity of each storage node. Both the basic functions of compute nodes and the calculation of parity increment require CPU and memory resources.

$$L_n = \{l_1, l_2, \dots, l_k\}, \quad (12)$$

where  $L_n$  denotes the set of the I/O load of each node. The I/O load performance represents the reading and writing performance of the storage node. Since computing nodes receive and forward data involving data reads and writes, a more accurate node selection weighting factor can be obtained by considering the I/O load.

$$A_n = \{a_1, a_2, \dots, a_k\}, \quad (13)$$

where  $A_n$  denotes the set of access bandwidths of each storage node. In the scenario of the multistripe concurrent update, the compute node, as the convergence node of  $\Delta d$  and the divergence node of  $p$ , has a relatively larger demand for access bandwidth. Therefore, larger access bandwidth means less possibility of congestion, thus improving the overall update efficiency.

The set of CPU utilization  $C_n$ , residual memory capacity  $M_n$ , and I/O load  $L_n$  can be obtained by periodically requesting status information from storage nodes. The set of the node access bandwidth can be calculated by using the SDN-based network measurement method in our previous work [25].

The NetM module follows the OpenFlow protocol of SDN to obtain the global network topology and update the real-time network information. The network information obtained by the NetM module is as follows:



$$P = \{p_1, p_2, \dots, p_n\}, \quad (14)$$

where  $P$  is the set of point-to-point paths between all nodes of each stripe. This path set can be obtained by the Dijkstra [26] algorithm.  $P$  is calculated during LAMU initialization, and  $P$  can be accessed directly in the subsequent path calculation, without repeated path calculation, which reduces the cost of the algorithm.

$$B_p = \{b_1, b_2, \dots, b_n\}, \quad (15)$$

where  $B_p$  represents the set of residual bandwidths for each path  $P$ . It can be calculated using the SDN-based network measurement method in our previous work [27].

$$D_p = \{d_1, d_2, \dots, d_n\}, \quad (16)$$

where  $D_p$  represents the set of transmission delays for each path. It can be obtained by using the SDN-based network measurement method [27].

$$H_p = \{h_1, h_2, \dots, h_n\}, \quad (17)$$

where  $H_p$  denotes the hops of data from the start node to the end node. It can be calculated from the length of each path  $P$ .

We use NodeM and NetM to obtain the storage node load and discover the network global status information mentioned above (10)–(17), which provides data support for the following computing node selection, aggregation path, and divergence path selection.

**4.2. The CNS Module.** As shown in Figure 3, LAMU selects nonduplicate computing nodes with better performance for multiple stripes by the CNS module. Firstly, when computing nodes are assigned to multiple update stripes, it is necessary to avoid numerous stripes selecting the duplicated computing node. Otherwise, the efficiency of parity-delta computing will be reduced and network congestion will occur. Secondly, according to Section 3.2, the parity-delta computing efficiency is positively correlated with the computing capacity of nodes, so the load status of heterogeneous nodes should be considered when selecting the computing node. Specifically, the CNS module uses the node load information obtained by the NodeM module to select computing nodes with better capacity by equation (20). Then, it deletes the selected nodes from the candidate computing node set to avoid concurrent update stripes from selecting the duplicate computing node. The entire process of the CNS module is as follows.

**4.2.1. Normalizing the Load Attributes.** To eliminate the dimension of each node load factor, a min–max normaliza-

tion method is used. Equation (18) is used for the node CPU utilization and disk I/O load factor, which can achieve better performance with smaller values. Equation (19) is used for the node residual memory and node access bandwidth factor, achieving better performance with larger values.

$$u(x) = \frac{x^{\max} - x}{x^{\max} - x^{\min}}, \quad (18)$$

$$v(x) = \frac{x - x^{\min}}{x^{\max} - x^{\min}}. \quad (19)$$

Then, the normalization decision factor vector can be obtained as follows:  $C_j^* = u(C_j)$ ,  $M_j^* = v(M_j)$ ,  $L_j^* = u(L_j)$ ,  $A_j^* = v(A_j)$ , and  $j \in \{1, 2, \dots, n\}$ ;  $j$  represents the sequence number of the candidate computing node.

**4.2.2. Calculating the Capacity of the Node.** The capacity of each candidate node can be calculated using the following equation:

$$\text{Capacity}_j = w_C C_j + w_M M_j + w_L L_j + w_A A_j, \quad (20)$$

where  $W = [w_C, w_M, w_L, w_A]$  is the vector of weighted coefficients for the node CPU utilization, residual memory, disk I/O load, and node access bandwidth.  $\text{Capacity}_j$  represents the weighted summation of the normalized factor of candidate node  $j$ . A node  $j$  with a larger  $\text{Capacity}_j$  value is a better computing node.

**4.2.3. Selecting the Computing Node.** To prevent severe network congestion and excessive node load, we need to avoid multiple update stripes selecting the same computing nodes. The entire process of the computing node selection is summarized in Algorithm 1.

**4.3. The Path Selection (PS) Module.** As described in Figure 3, when processing the multistripe concurrent update request, after LAMU selects the computing node with the CNS module, the system uses the PS module to schedule the update traffic, which includes the convergence traffic between the data nodes and computing node and the divergence traffic between the computing node and parity nodes. Specifically, LAMU uses the real-time network status and the multiattribute decision-making method based on TOPSIS to schedule the update traffic. In order to improve the update efficiency and maintain better system load balancing, we adjust the weight of decision factors under different network loads. The entire process of the PS module is as follows:

*Step 1.* Obtain candidate path.

The PS module firstly filters the existing path set  $P$  according to the network bandwidth requirements of the

1. Inputs:  
 $N$ : candidate computing nodes set  
 $S$ : concurrent update stripes set  
 $C_n$ : CPU utilization  
 $M_n$ : remaining memory  
 $L_n$ : I/O load  
 $A_n$ : access bandwidth  
 Output: best computing nodes for concurrent update stripes
2. **For**  $s$  in stripe set  $S$  **do**
3.   **For**  $j$  in node set  $N$  **do**
4.     Obtain the load parameters  $C_n, M_n, L_n, A_n$
5.     Normalize load parameter to  $C_n^*, M_n^*, L_n^*, A_n^*$  according to (18) and (19)
6.     Calculate the capacity of node  $j$  according to (20)
7.     Set the node has largest capacity as computing node  $R$  for stripe  $s$
8.     Delete  $R$  from  $N$  to ensure that the computing nodes selected by multiple stripes are not duplicated
9.   **End for**
10. **End for**

ALGORITHM 1: Computing node selection module.

update traffic and then obtains the candidate path set  $P^*$ , where  $b_\phi$  is the network bandwidth requirements of the update traffic  $\phi$ .

$$P^* = \{p_1, p_2, \dots, p_n\}, \quad (21a)$$

$$b_e \geq b_\phi, \quad \forall e \in P^*, \quad (21b)$$

$$P^* \subseteq P. \quad (21c)$$

$$\mathbf{M} = \begin{bmatrix} b_1 & b_2 & \dots & b_n \\ d_1 & d_2 & \dots & d_n \\ h_1 & h_2 & \dots & h_n \end{bmatrix}, \quad (22)$$

where  $\mathbf{M}$  is the decision-making matrix for finding the best path for each point-to-point update traffic; each column in  $\mathbf{M}$  presents a candidate path. The symbols  $b$ ,  $d$ , and  $h$  in each column denote each path's residual bandwidth, transmission delay, and network hops, respectively. These network attributes are obtained by the NetM module.

*Step 2.* Construct and normalize the decision-making matrix.

To eliminate the influence of dimensions between each network attribute, the min-max normalization method is used, as shown in equations (18) and (19) in Section 4.2. Equation (18) is used for the path delay and network hop attribute, which can achieve better performance with smaller values. Equation (19) is used for the residual bandwidth, achieving better performance with larger values.

Then, the normalization decision-making matrix  $\mathbf{M}^*$  is described as follows:

$$\mathbf{M}^* = \begin{bmatrix} b_1^* & b_2^* & \dots & b_n^* \\ d_1^* & d_2^* & \dots & d_n^* \\ h_1^* & h_2^* & \dots & h_n^* \end{bmatrix}, \quad (23)$$

where  $b_j^* = v(b_j)$ ,  $d_j^* = u(d_j)$ ,  $h_j^* = u(h_j)$ , and  $j \in \{1, 2, \dots, n\}$ ;  $j$  is the sequence number of the matrix column, corresponding to the sequence number of the candidate path of the update traffic.

$$W = [w_b, w_d, w_h]^T, \quad (24)$$

where  $W$  is the vector of weighted coefficients;  $w_b$ ,  $w_d$ , and  $w_h$  are the weight coefficients of the residual bandwidth, path delay, and network hops, respectively. The value of the weight coefficient set is usually determined through experiment [28] and will be introduced in Section 5. The weighted decision matrix can be obtained using the following equation:

$$Z_{ij} = W_i \times M_{ij}^*, \quad (25)$$

where  $i \in \{1, 2, 3\}$  and  $j \in \{1, 2, \dots, n\}$ .

$$P_i^+ = \max_j \{Z_{ij} \mid i = 1, 2, 3\}, \quad j \in \{1, 2, \dots, n\}, \quad (26a)$$

$$P_i^- = \min_j \{Z_{ij} \mid i = 1, 2, 3\}, \quad j \in \{1, 2, \dots, n\}, \quad (26b)$$

*Step 3.* Construct the weighted decision matrix.

*Step 4.* Obtain the positive and negative ideal solutions.

1. **Inputs:**  
Candidate path set  $P$ ; path residual bandwidth set  $B_p$ ; path delay set  $D_p$ ; path hop set  $H_p$ ; source node  $n_{src}$  and destination node  $n_{dst}$  of convergence flow or divergence flow; bandwidth requirement  $b_\emptyset$ ; of update flow; vector of weighted coefficients for the residual bandwidth, end-to-end delay, and network hops  $W = [w_b, w_d, w_h]^T$   
Output: the best path from update traffic source  $n_{src}$  to update traffic destination  $n_{dst}$
2. **for** path  $p$  in path set  $P$  **do**
3.   **if**  $p$  is from  $n_{src}$  to  $n_{dst}$  and  $b_p > b_\emptyset$  **do**
4.     add  $p$  to path set  $P_n$
5.   **end if**
6. **end for**
7. Build the decision matrix  $M$  based on  $P_n$  according to Equation (22)
8. Normalize  $M$  to  $M^*$  according to Equation (23)
9. Construct the weighted decision matrix  $Z$  based on  $M^*$  according to Equation (25)
10. Calculate the positive  $P^+$  and negative ideal solution  $P^-$  of weight matrix  $Z$  according to (26)
11. Calculate the Euclidean distance  $D^+, D^-$  from each candidate path to the positive and negative ideal solutions according to (27)
12. Calculate the relative closeness  $C_j^+$  between each candidate path and the best candidate path according to (28)
13. **Return** the candidate path  $p$  with the largest relative closeness as the route path

ALGORITHM 2: Path Selection module.

where  $P_i^+$  represents the positive ideal solution of the  $i_{th}$  attribute value, which is composed of the maximum value of each decision factor, and  $P_i^-$  represents the negative ideal solution of the  $i_{th}$  attribute value, which is composed of the minimum value of each decision factor.

$$D_j^+ = \sqrt{\sum_{i=1}^3 (Z_{ij} - P_i^+)^2}, \quad j \in \{1, 2, \dots, n\}, \quad (27a)$$

$$D_j^- = \sqrt{\sum_{i=1}^3 (Z_{ij} - P_i^-)^2}, \quad j \in \{1, 2, \dots, n\}, \quad (27b)$$

where  $Z_{ij}$  is an element in the candidate path  $[Z_{1j}, Z_{2j}, Z_{3j}]^T$  and  $D_j^+$  and  $D_j^-$  are the distances from each candidate path to the positive and negative ideal solutions, respectively.

$$C_j^+ = \frac{D_j^-}{D_j^+ + D_j^-}, \quad j \in \{1, 2, \dots, n\}. \quad (28)$$

*Step 5.* Calculate the distance from the candidate path to the positive and negative ideal solutions.

*Step 6.* Calculate the relative closeness  $C_j^+$  between each candidate path and the optimal candidate path.

When the relative closeness  $C_j^+$  is larger, the path is more suitable for the update traffic.

The entire process of the PS module is summarized in Algorithm 2.

## 5. Implementation and Evaluation

*5.1. Experiment Environment.* The performance of the proposed erasure-coding update scheme is evaluated in this section. We implement the prototype of LAMU on Container-

net [17], a fork of the famous Mininet [29] network emulator. Different from Mininet, Containernet uses the Docker [30] containers as hosts in emulated network topologies. This feature allows Containernet to better simulate distributed storage systems. Ryu [31] is used as the SDN controller that supports the OpenFlow protocol. The entire experimental environment is deployed on an Ubuntu 18.04 system on a Sugon A840r-G, which has  $64 * 2.1$  GHz AMD processors and 128 GB of memory. In terms of the experimental topology, we use Containernet 3.1.0 to simulate the fat-tree topology [32]. As shown in Figure 4, the bandwidth capacity of each link in the fat tree is set to 200 Mbps because the simulation experiment assumes limited resources. Storage nodes in the fat-tree topology are heterogeneous; when selecting the computing node for each update stripe, we set the weight of the access bandwidth, CPU utilization, residual memory, and I/O load to  $[0.6, 0.1, 0.1, 0.2]$ .

To evaluate the performance of LAMU in a more realistic environment, we use the real distributed storage system background traffic pattern, which was measured in our previous work [33], to reproduce the realistic network condition, as shown in Table 2. According to [33], the speed of the heartbeat traffic is set to 1 Mbps to reduce the packet loss rate in the experimental environment; all the background traffic is maintained for a long time to ensure that the background traffic exists throughout the whole update process. To further evaluate the efficiency of our LAMU method under different network loads, three kinds of traffic load scenarios are set in the evaluation, as follows:

- (i) *Low-load (LL) scenario:* 10 heart beating flows, 10 user data flows, and 10 migration flows
- (ii) *Middle-load (ML) scenario:* 20 heart beating flows, 20 user data flows, and 20 migration flows
- (iii) *High-load (HL) scenario:* 30 heart beating flows, 30 user data flows, and 30 migration flows

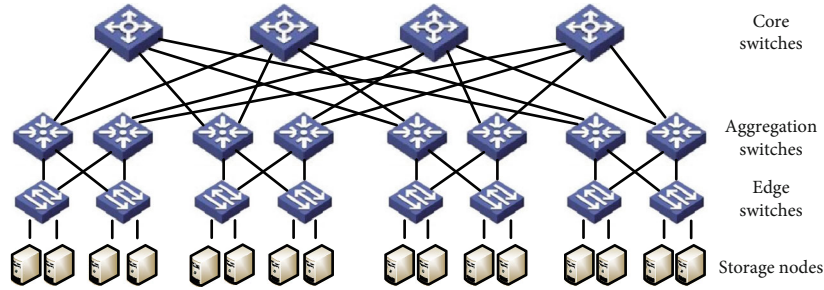


FIGURE 4: Experimental fat-tree topology.

TABLE 2: The statistical information of the different types of traffic in the distributed storage system.

Traffic type	Speed (Mbps)	Duration (s)	Packet number	Avg. packet size (bytes)
Heartbeat traffic	92.87	0.006554	54	1477.42
User data traffic	12.93	39.31	67073	1508.65
Migration traffic	4.36	654.12	340354	1505.56

The value of the weight coefficient set is usually determined through experiments [28]. As the system load increase, the network bandwidth resources become more limited. Therefore, we increase the bandwidth weight with the load increase. We set  $W = [w_b, w_d, w_h]^T$  mentioned in equation (24) to  $[0.25, 0.6, 0.15]$ ,  $[0.5, 0.4, 0.1]$ , and  $[0.7, 0.2, 0.1]$  for the LL, ML, and HL scenarios, respectively.

In the evaluation, we compare LAMU with PUM-P [22] and DelaySelect. For PUM-P, it also improves the update efficiency by introducing the computing node. Yet, PUM-P ignores the heterogeneity of computing nodes and network status; all nodes and route paths have an equal probability of being selected. DelaySelect is extended from [23], which also adopts a centralized update framework and improves the update efficiency by selecting the path with the least delay as the routing path for update traffic. The experimental comparison items include the average update time, the standard deviation of bandwidth, and the link maximum bandwidth utilization of the system.

We focus on the update performance between different update schemes under various system load scenarios. In terms of experimental parameters, we use the parameters that may impact the update performance, including the number of update data nodes, the number of parity nodes, the size of data-delta, and the number of update stripes. The range of these parameters is listed in Table 3. To get a more convincing experiment result, each experiment was done 10 times, and the average value of these experiments was taken as the result.

## 5.2. Update Efficiency

**5.2.1. Average Update Time with Varying Numbers of Parity Nodes in Different Load Scenarios.** This subsection presents extensive comparisons of the average update time of three update schemes under different experimental parameters and load scenarios. Figure 5 shows the average update time

increase along with  $p$  when  $d = 8$ . As the number of parity nodes increases, more parity-delta needs to be transmitted, increasing the average update time. While the load becomes higher, the update time between different schemes begins to present differences. As we can see, in the high-load (HL) scenario, LAMU reduces the average update time by 17.9% and 43.1% compared with DelaySelect and PUM-P, respectively.

**5.2.2. Average Update Time with Varying Numbers of Update Data Nodes in Different Load Scenarios.** Figure 6 presents that the average update time is generally stable with the increase of update data nodes in different load scenarios. This is because, on the premise that the data volume is constant, the increasing number of update data nodes will reduce the average data-delta sent by each data node. Therefore, the extra time caused by connecting more data nodes is offset. While the load becomes higher, the update time between different schemes begins to show more significant differences. As we can notice, in the low-load (LL) scenario, the three update schemes achieve a comparable average update time. In the middle load (ML) scenario, LAMU starts to show better update efficiency. In the HL scenario, LAMU reduces the average update time by 18.8% and 49.5% compared with DelaySelect and PUM-P, respectively.

**5.2.3. Average Update Time with Varying Sizes of Update Data Volume in Different Load Scenarios.** Figure 7 illustrates how the average update time increases along with the update data volume in different load scenarios. The three update schemes achieve comparable average update times in the LL scenario. In the ML scenario, LAMU starts to show better update efficiency. Compared with DelaySelect and PUM-P, LAMU reduces the average update time by 12.1% and 26.5% under the ML scenario, respectively, and 19.7% and 43.4% under the HL scenario, respectively.

**5.2.4. Average Update Time with Varying Update Stripes in Different Load Scenarios.** Figure 8 illustrates the average update time variation with the number of concurrent update stripes. As we can see, with the increase of the number of concurrent update stripes in all three scenarios, the average update time of LAMU is only increasing a little. It illustrates that LAMU is more efficient in dealing with the multistripe concurrent update. However, the update time increases significantly when adopting the DelaySelect and PUM-P schemes with the increase of the number of update stripes. Specifically,

TABLE 3: Experiment parameters.

Parameter	Ranges	Default
The number of data node for update of each stripe, $d$	6, 7, 8, 9, 10	8 update data nodes
The number of parity node of each stripe, $p$	3, 4, 5, 6, 7	6 parity nodes
The size of update data volume (MB)	1, 2, 4, 8, 16	8 MB
The number of update stripe	3, 4, 5, 6, 7	5 update stripes

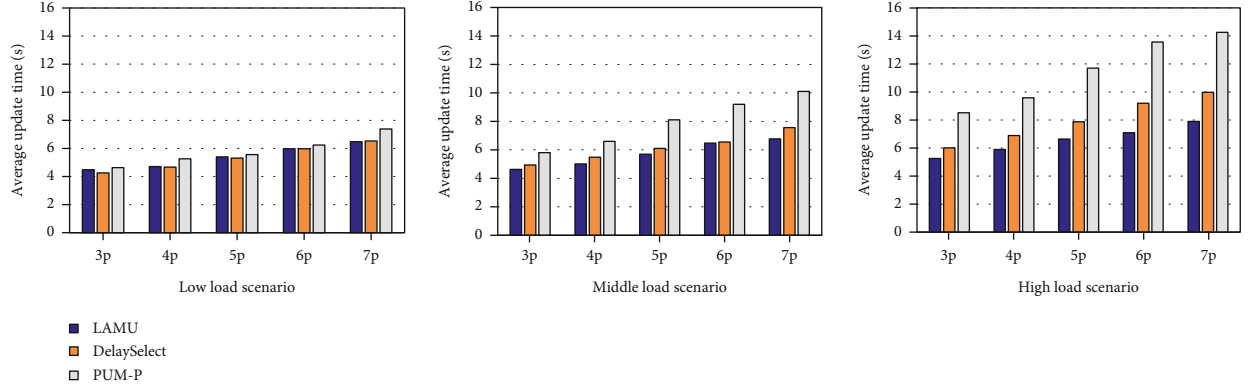


FIGURE 5: Average update time comparison with the variation of the number of parity nodes in different load scenarios.

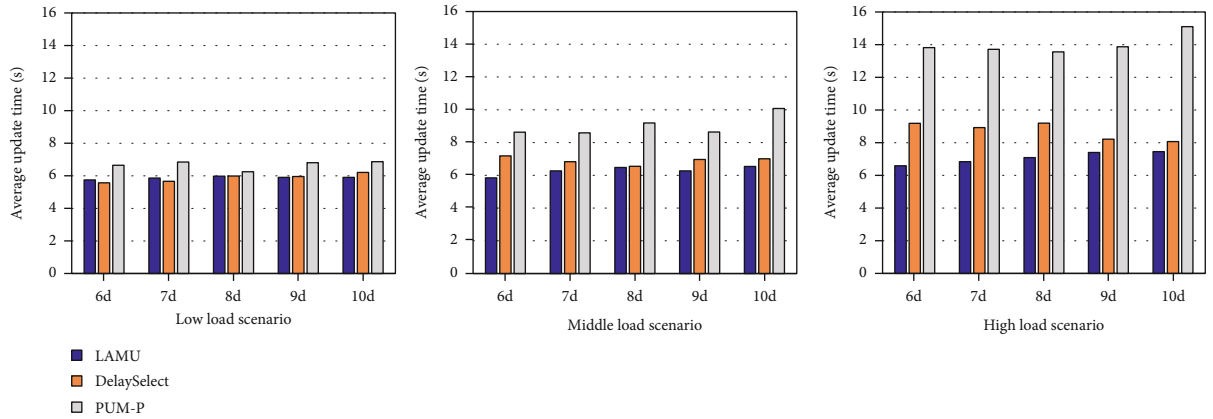


FIGURE 6: Average update time comparison with the variation of the number of data nodes in different load scenarios.

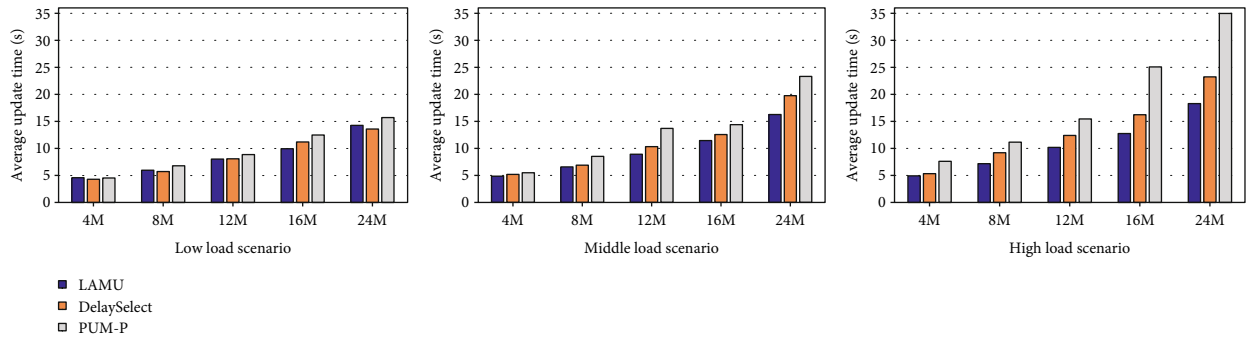


FIGURE 7: Average update time comparison with the variation of the update data volume in different load scenarios.

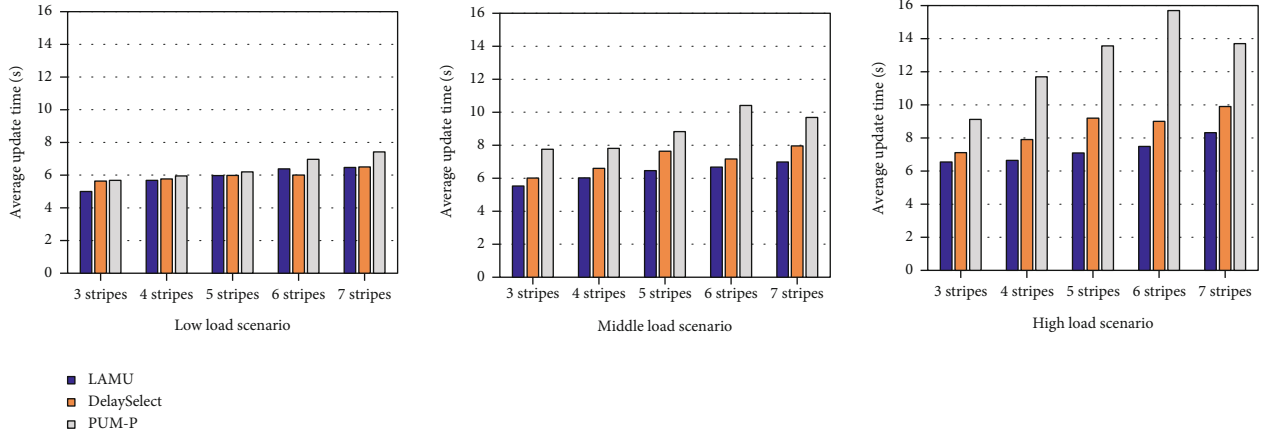


FIGURE 8: Average update time comparison with the variation of the update stripe number in different load scenarios.

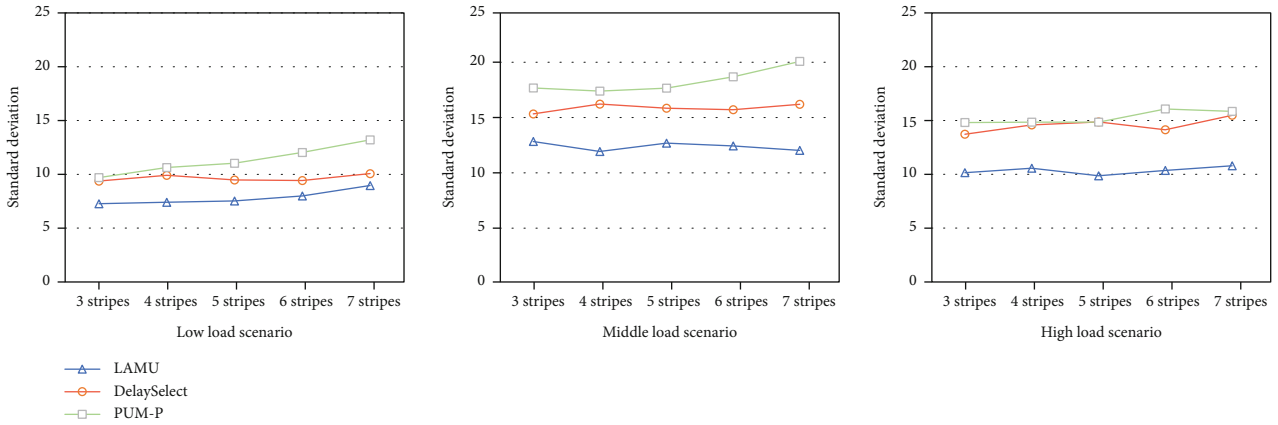


FIGURE 9: Standard deviation of link utilization with the variation of the number of update stripes in different load scenarios.

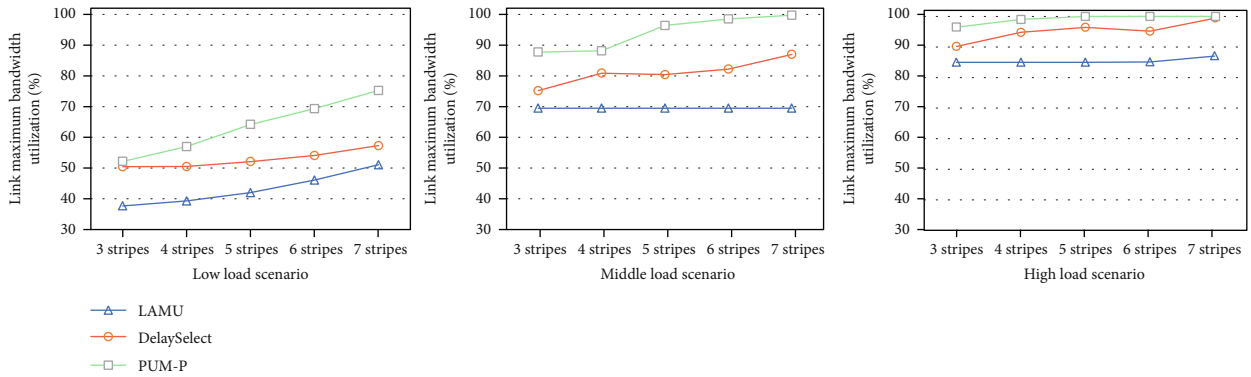


FIGURE 10: Maximum link bandwidth utilization with the variation of the number of update stripes in different load scenarios.

compared with DelaySelect and PUM, LAMU reduces the average update time by 10.4% and 28.8% under the ML scenario, respectively, and reduces the average update time by 16.3% and 43.4% under the HL scenario, respectively.

### 5.3. Network Load-Balancing Performance

#### 5.3.1. Standard Deviation of Link Bandwidth Utilization with Varying Update Stripes in Different Load Scenarios.

To verify the load-balancing performance of the three update schemes, we evaluate the standard deviation of link utilization, as presented in Figure 9. The lower the standard deviation of link utilization, the more balanced the link loads are. PUM-P has the largest standard deviation in all three scenarios. This finding is because PUM-P does not consider the network status when scheduling the update traffics, which easily leads to load imbalance, while we can notice that in the HL scenario, the standard deviation of PUM-P



is a little lower than that of the ML scenario. The reason is that, with the load increasing, PUM-P makes more and more links saturated. Thus, the standard deviation is decreased. The DelaySelect has a lower standard deviation than PUM-P for DelaySelect uses the link delay to schedule the update traffic, which has better load-balancing performance. The LAMU has the lowest standard deviation in all three scenarios. This result is because LAMU comprehensively considers link bandwidth, delay, and path hop when scheduling the update traffic; LAMU achieves better load balancing and avoids network congestion caused by several links reaching the full load.

*5.3.2. Network Maximum Link Bandwidth Utilization with Varying Update Stripes in Different Load Scenarios.* The maximum link bandwidth utilization represents the utilization of the most congested link in the system, and the larger it is, the more unbalanced the system is. As shown in Figure 10, in the ML scenario, full load links have already appeared in PUM-P and have also appeared in DelaySelect in the HL scenario. It means that some links in the system are highly congested. The LAMU method has the lowest maximum link bandwidth utilization in all three scenarios, which means LAMU can achieve better load balancing and avoids network congestion caused by links reaching the full load.

## 6. Conclusions

Erasure coding has become an indispensable redundancy mechanism in today's large-scale distributed storage system. However, the data update of erasure coding introduces additional computing load and network traffic, reducing the efficiency of data updates and affecting the system load balancing. Most of the existing erasure-coding update schemes ignore the heterogeneity of node and network status and the multistripe concurrent update caused by data correlation. To solve this problem, this paper establishes the optimization model of multistripe updates with multiple QoS constraints in the heterogeneous environment and then proposes LAMU, a load-aware multistripe concurrent update scheme. Firstly, LAMU introduces SDN to measure the node load and network status in real time, and then, the obtained nodes and network information are used to select nonduplicated computing nodes with better capacity for multiple update stripes. Finally, the multiattribute decision-making method is used to schedule the network traffic between data nodes, computing nodes, and parity nodes. Extensive experimental results show that LAMU can reduce the average update time while providing better load-balancing performance.

Moreover, we'll consider implementing LAMU in a real erasure-coded storage system in the future. Another direction for future work is to use reinforcement learning to adjust the decision parameter weight when scheduling the update traffic and making a trade-off between the number and the location of the computing nodes to achieve better results.

## Data Availability

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

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research work obtained the subsidization of the National Natural Science Foundation of China (Nos. 61861013 and 62161006), the Science and Technology Major Project of Guangxi (No. AA18118031), and the Innovation Project of Guangxi Graduate Education (No. YCSW2022271).

## References

- [1] D. Ford, F. Labelle, F. I. Popovici et al., "Availability in globally distributed storage systems," in *Proceedings of the 9th USENIX conference on Operating systems design and implementation*, pp. 61–74, Vancouver, BC, Canada, 2010.
- [2] C. A. Rincón, J.-F. Pâris, R. Vilalta, A. M. Cheng, and D. D. Long, "Disk failure prediction in heterogeneous environments," in *International Symposium on Performance Evaluation of Computer and Telecommunication Systems*, pp. 1–7, Seattle, WA, July 2017.
- [3] S. S. Arslan and E. Zeydan, "On the distribution modeling of heavy-tailed disk failure lifetime in big data centers," *IEEE Transactions on Reliability*, vol. 70, no. 2, pp. 507–524, 2021.
- [4] H. Weatherspoon and J. D. Kubiatowicz, "Erasure coding vs. replication: a quantitative comparison," in *International Workshop on Peer-to-Peer Systems*, pp. 328–337, Springer, Cambridge, MA, USA, 2002.
- [5] S. Muralidhar, W. Lloyd, S. Roy et al., "f4: Facebook's warm BLOB storage system," in *11th USENIX Symposium on Operating Systems Design and Implementation*, pp. 383–398, Broomfield, CO, USA, 2014.
- [6] C. Huang, H. Simitci, Y. Xu et al., "Erasure coding in windows Azure storage," in *Proceedings of the USENIX conference on Annual Technical Conference*, p. 2, Boston, MA, 2012.
- [7] M. Sathiamoorthy, M. Asteris, D. Papailiopoulos et al., "XORing elephants," *Proceedings of the VLDB Endowment*, vol. 6, no. 5, pp. 325–336, 2013.
- [8] D. Narayanan, A. Donnelly, A. Rowstron, and Usenix, "Write off-loading: practical power management for enterprise storage," in *6th USENIX Conference on File and Storage Technologies*, pp. 253–267, San Jose, CA, 2008.
- [9] J. C. Chan, Q. Ding, P. P. Lee, and H. H. Chan, "Parity logging with reserved space: towards efficient updates and recovery in erasure-coded clustered storage," in *12th {USENIX} Conference on File and Storage Technologies*, pp. 163–176, Santa Clara, CA, USA, 2014.
- [10] D. J. Ellard, *Trace-Based Analyses and Optimizations for Network Storage Servers*, Harvard University, 2004.
- [11] Z. Fengyan, W. Yan, and L. Nianshuang, "Survey of heterogeneous-based data repair strategies for erasure codes," *Application Research of Computers*, vol. 36, no. 8, pp. 2249–2255, 2019.

- [12] M. Ye, R. Wei, W. Guo, Q. Jiang, H. Qiu, and Y. Wang, "A new method for reconstructing data on a single failure node in the distributed storage system based on the MSR code," *Wireless Communications & Mobile Computing*, vol. 2021, pp. 1–14, 2021.
- [13] S. Maitrey and C. K. Jha, "MapReduce: simplified data analysis of big data," in *3rd International Conference on Recent Trends in Computing*, pp. 563–571, Delhi, India, 2015.
- [14] Z. Yuan, X. You, X. Lv, and P. Xie, "SS6: online short-code RAID-6 scaling by optimizing new disk location and data migration," *The Computer Journal*, vol. 64, no. 10, pp. 1600–1616, 2021.
- [15] Z. R. Shen, P. P. C. Lee, J. W. Shu, and W. Z. Guo, "Correlation-aware stripe organization for efficient writes in erasure-coded storage: algorithms and evaluation," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 7, pp. 1552–1564, 2019.
- [16] Z. Li, Z. Chen, S. M. Srinivasan, and Y. Zhou, "C-Miner: mining block correlations in storage systems," in *3rd Conference on File and Storage Technologies*, pp. 173–186, Usenix Assoc, San Francisco, CA, 2004.
- [17] "Contanernet," 2022, <https://containernet.github.io/>.
- [18] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *Journal of the society for industrial applied mathematics*, vol. 8, no. 2, pp. 300–304, 1960.
- [19] X. Pei, Y. Wang, X. Ma, and F. Xu, "T-update: a tree-structured update scheme with top-down transmission in erasure-coded systems," in *IEEE INFOCOM - The 35th Annual IEEE International Conference on Computer Communications*, pp. 1–9, San Francisco, CA, USA, April 2016.
- [20] S. Manen, M. Guillaumin, and L. Van Gool, "Prime object proposals with randomized Prim's algorithm," in *IEEE International Conference on Computer Vision*, pp. 2536–2543, Sydney, Australia, 2013.
- [21] Y. Wang, X. Pei, X. Ma, and F. Xu, "TA-update: an adaptive update scheme with tree-structured transmission in erasure-coded storage systems," *IEEE Transactions on Parallel Distributed Systems*, vol. 29, no. 8, pp. 1893–1906, 2017.
- [22] F. Zhang, J. Huang, and C. Xie, "Two efficient partial-updating schemes for erasure-coded storage clusters," *IEEE Seventh International Conference on Networking, Architecture, and Storage*, 2012, pp. 21–30, Xiamen, China, 2012.
- [23] L. Qian, H. Yupeng, Y. Zhenyu, X. Ye, and Q. Zheng, "An ant colony optimization algorithms based data update scheme for erasure-coded storage systems," *Journal of Computer Research and Development*, vol. 58, no. 2, p. 305, 2021.
- [24] Q. Fenglin, G. Qingyuan, Z. Yangfan, and X. Wang, "Heterogeneity-aware node selection for data repair in distributed storage systems," *Journal of Computer Research and Development*, vol. 52, no. 2, pp. 68–74, 2015.
- [25] Y. Wang, M. Ye, Q. He, Y. Huan, and W. Kang, "A new node selecting approach in Ceph storage system based on software defined network and multi-attributes decision-making model," *Chinese Journal of Computers*, vol. 42, no. 2, pp. 93–108, 2019.
- [26] M. Barbehenn, "A note on the complexity of Dijkstra's algorithm for graphs with weighted vertices," *IEEE Transactions on Computers*, vol. 47, no. 2, pp. 263–263, 1998.
- [27] W. Ke, Y. Wang, M. Ye, and J. Chen, "A priority-based multi-cast flow scheduling method for a collaborative edge storage datacenter network," *IEEE Access*, vol. 9, pp. 79793–79805, 2021.
- [28] Y. Luo, J. Xia, and T.-p. Chen, "Comparison of objective weight determination methods in network performance evaluation," *Journal of Computer Applications*, vol. 29, no. 10, pp. 2624–2626, 2009.
- [29] "Mininet," 2022, <http://mininet.org/>.
- [30] "Docker," 2022, <https://www.docker.com/get-started>.
- [31] "Ryu," 2022, <https://osrg.github.io/ryu/>.
- [32] C. Zhang, S. Zhang, B. Jin, W. Li, Z. Wang, and Y. Wang, "A3: an automatic malfunction detection and fixation system in FatTree data center networks," in *Conference of the ACM-Special-Interest-Group-on-Data-Communication*, pp. 24–26, Beijing, China, 2019.
- [33] W. Ke, Y. Wang, and M. Ye, "GRSA: service-aware flow scheduling for cloud storage datacenter networks," *China Communications*, vol. 17, no. 6, pp. 164–179, 2020.

## Research Article

# Lightweight Security Wear Detection Method Based on YOLOv5

Sitong Liu,<sup>1</sup> Nannan Zhang,<sup>2</sup> and Guo Yu <sup>3</sup>

<sup>1</sup>Department of Software Engineering, Northeastern University, Shenyang 110000, China

<sup>2</sup>Department of Physical Education, Northeastern University, Shenyang 110000, China

<sup>3</sup>Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, East China University of Science and Technology, Shanghai 200000, China

Correspondence should be addressed to Guo Yu; guoyu@ecust.edu.cn

Received 14 March 2022; Accepted 28 April 2022; Published 13 May 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Sitong Liu 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.

Given a large number of network parameters of the existing security wear detection, it is difficult to run on the embedded platform. Based on the idea of deep separable convolution, a lightweight security wearable target detection network based on improved YOLOv5 is proposed. Specifically, the feature extraction network structure of YOLOv5 is lightweight improved to reduce computation of the proposed model, including the increase of the number of network layers and decrease of the number of parameters. In addition, an attention mechanism is introduced to weigh different channels of the feature map to improve detection accuracy. The model has been tested on PASCAL VOC dataset and security wear dataset. The experimental results show that the size of the proposed model is 8.0 MB, the number of parameters is  $7.5 \times 10^5$ , and the number of FLOPs is  $7.5 \times 10^5$ . Compared with the YOLOv5 model, the required memory is reduced by 44.8%, and the number of parameters decreased by 45.58%, FLOPs decreased by 54.54%. Accordingly, the results have demonstrated that the proposed method can significantly improve the detection speed while maintain the accuracy. Especially, we have successfully deployed the proposed model in the high-speed detection of security wear.

## 1. Introduction

Target detection is a classical task in the field of computer vision, such as scene content analysis and understanding [1]. In recent years, more and more artificial intelligence research has been applied to various intelligent systems to obtain greater benefits [2]. For example, radio frequency identification networks are widely used in the applications of the Internet of things (IOT) [3], biomedical domains [4], including retail production monitoring, supply chain management, localization, and navigation. However, for engineering industries such as steel mills, the production site is relatively fixed. Most of the work and business of construction enterprises take place on the construction site, but there are always many problems in the management and supervision of the construction site, such as difficult safety management, difficult monitoring of accidents that

hide dangers, and difficult monitoring of work progress on the construction site. Generally, the larger the scale and the more complex the process of the project, the greater the requirements for the supervision and management of the construction site, and the greater the difficulty of management. With the development of science and technology, intelligent factories have become an important part of intelligent city construction and a typical application of AI, IOT, and other technologies in traditional engineering industries [5]. Many companies promote this technology and products on social networks, which can help users realize more intelligent and standardized management of personnel, mechanical equipment, materials, and materials on the construction site [6].

An intelligent factory changes passive monitoring into active monitoring in the form of visualization and data. As a result, the multilevel and all-around real-time monitoring

of project progress and standardized civilized construction will be carried out. This way can provide prewarning of the abnormal accidents and quickly analyze the accident, which will keep the factory safe and improve the efficiency of the management of the factory. As a part of smart construction sites, security wear detection is becoming the standard configuration of AI+ smart construction sites. In areas prone to falling objects, such as steel plants, power plants, construction sites, coal mines, and smelting, operators must wear safety helmets and protective clothes. Due to the lack of standard wearing of protective clothing and safety helmets, accidents such as falling object impacts, collisions, scalds, and smashes will cause great harm to the lives and safety of workers, while safety accidents on the construction site due to the lack of safety helmets and protective clothing occur frequently. At the beginning of the 20th century, safety helmets were gradually used on construction sites, reducing the annual accident death toll to less than 2.5%, which proved that wearing safety helmets correctly can effectively reduce risks [7]. However, for a long time, people in construction areas in China generally have had a weak awareness of self-safety protection [8] and often fail to realize the importance of wearing safety helmets. According to the data survey, among the 42 major accidents in the construction industry in 2016, the casualties caused by collapse accounted for 84% of the total casualties, a large part of which was due to the failure to wear protective measures such as safety helmets and protective clothing. Helmets and protective clothing are important protective tools for factory workers at the construction site, but many workers choose not to wear them because of the lack of comfort in helmets and cumbersome wearing of protective clothing, which will endanger the lives and safety of workers [9]. Therefore, it is important to check whether factory workers are wearing safety protection equipment correctly in real time. However, the working environment in some factories is very dangerous, so it is not suitable to use the manual area for real-time inspection and management. Therefore, consider using machines instead of manpower for security wear detection. These protective tools can prevent safety accidents to a certain extent and ensure the physical safety of factory workers. With the development of powerful computing power ES (Edge Storage) (ES means the edge storage, which is a tool to directly store data during the data collection.) [10] and computer vision, unmanned intelligent security detection methods have attracted people's attention because of their advantages of low detection cost and high efficiency [11]. Security wear detection has been studied by many scholars at home and abroad because of its complex working environment, shooting angle, distance from the target, and so on. Most researchers use the helmet as the primary research object. There are many types of research on helmet-wearing detection at home and abroad, mainly including the following two ideas.

The first method is through machine learning or deep learning algorithms. During the traditional helmet-wearing detection, the position of workers, pedestrians, faces in the image, or the position of the image foreground information is firstly extracted. Then, the extracted information is used to

infer whether the target exists in the approximate area of the image. Finally, the circular Hough transform or SVM is applied to judge whether there is a helmet in this area. The traditional helmet-wearing detection algorithm mainly recognizes color and shape features. As discussed by Li [12], it was proposed to study how to locate the head area and calculate the color characteristics of the helmet to detect it. Liu and Ye[13] proposed using skin color to locate the face and intercept the area above the face, then taking the extracted Hu moments (Hu moment is a two-dimensional moment invariant theory [13] with a good invariance and anti-interference on the rotation and scaling changes of targets in an image, which is commonly used to effectively reflect the essential characteristics of the image [14].) [14] as the feature, and finally using the Hu moment feature extracted by SVM training to obtain the classifier that can detect the helmet. As discussed by Park et al. [15], they used the hog feature in the pedestrian detection stage, the color histogram in the helmet detection stage, and the spatial matching relationship between the human body and the helmet to judge whether the personnel were wearing the helmet. Feng et al. [8] proposed detecting the foreground with a Gaussian mixture model, then dealing with the connected domain, detecting the human body with a model-based method, and detecting the helmet with a SIFT feature and color statistical feature. Rubaiyat et al. [16] proposed combining the hog feature and the frequency domain information of the image to detect the workers, and then using the circular Hough transform and color information to judge whether a person is wearing a hat. Hence, the learning mechanisms of the individuals played a significant role in the algorithm's performance [17].

In a second way, the object detection algorithm based on a deep convolution network is used to train on the dataset to establish the model and to detect and identify the security wear of construction personnel. Huang and Pan [18] proposed using a parallel network to detect the human body on LeNet and then detect helmets through color features. Fang et al. [19] realized the detection of personnel's helmet-wearing in surveillance video by improving Faster R-CNN. As discussed by Zhang et al. [20], who proposed a helmet-wearing detection algorithm combining OpenPose and Faster-RCNN. First, OpenPose is used to detect the head and neck of people, and then, Faster-RCNN is used to detect the helmet in the image. Finally, the spatial relationship between the head, neck, and the helmet is analyzed to judge whether people wear the helmet. Zhang and Xu[21] improved SSD, used VGG as the backbone, chose the Adam optimizer to accelerate convolutional neural network convergence, and improved helmet-wearing detection accuracy by using characteristic diagrams of different scales. Fang et al. [22] improved its network structure based on the YOLO v2 target detection algorithm. By adding dense blocks to the original YOLO v2, the sensitivity of the network to small target detection is improved. Then, the deep separable convolution is used to compress the network, which increases the availability of the model. As discussed by Zhang et al. [23], they realized the detection of workers' helmet-wearing in the monitoring video through Faster-



RCNN. Wang [24] obtained the construction worker identification network and helmet identification network by improving the YOLO network structure, trained the network to obtain the generalization model, and then conducted semisupervised online learning on the obtained model, to obtain the helmet-wearing detection algorithm with high accuracy.

Compared with traditional algorithms, deep learning image recognition can extract more accurate image features and has a stronger recognition ability. However, the image recognition algorithm of deep learning needs to train on a large number of training datasets to learn the model weight, so an important premise of deep learning is to label the dataset. To solve the problems of large consumption of human and material resources and easy inspection gaps in manual supervision in the special working environment of some factories, after selecting the data collected by a factory for labeling and adding some filtered data from the open-source dataset HWD, the labeling data is added again to form a new security wear detection dataset. The target detection algorithm YOLOv5 is used to train and learn on the security wear detection dataset, and the security wear detection baseline model is established. In recent years, deep learning has become one of the most popular research methods for target detection because of its high accuracy and strong robustness. As discussed by Li et al. [11], they use the SSD [25] model to detect whether the helmet is worn. At present, the target detection algorithm based on deep learning mostly lays anchor frames of different sizes on the image and realizes target detection through regression and classification anchor frames. According to the generation method of the regression box, it is mainly divided into two stages and a single stage. The two-stage detector, such as Faster-RCNN [26], has high accuracy, but the detection speed is slow. Single-stage detectors, such as the YOLO deep learning algorithm, are particularly attractive because of their good recognition performance. Since the YOLO v1 [27] model in the field of target recognition was proposed by Redmon in 2016, the YOLO series has been constantly innovating. The YOLO v2 [28] model, YOLO v3 [29] model, YOLO v4 [30] model, and YOLO v5 are the new versions of the YOLO series. The innovative products are continuously integrated based on the YOLO series. In many current application scenarios, a problem usually involves multiple conflicting targets and is usually subject to a given set of constraints [31]. This paper selects the detection speed as the evaluation. Among them, the YOLO v5 model has the best performance and is suitable for practical engineering applications. The official YOLO v5 target detection network has given four network models: YOLO v5s, YOLO v5m, YOLO v5l, and YOLO v5x. The YOLO v5s network model is a YOLO network with the smallest depth and feature map width among the four sizes. Therefore, this paper uses the YOLO v5s detection network as the benchmark for security wear detection.

## 2. Related Work

*2.1. The YOLO v5 method.* In 2017, the YOLO algorithm was optimized by Redmon and Farhadi [29]. Specifically, the

convolution layer is used to replace the full connection layer on the basis of the idea of an anchor box in Faster-RCNN [26]. In addition, a BN (Batch Normalization) [32] layer is added to the convolution layer to further improve the detection effect, such as accuracy and speed. The architecture of YOLO v5 is shown in Figure 1. The YOLO v5 network consists of three parts: the backbone network, the neck, and the output. The main part focuses on extracting the feature information image from the input, fusing the extracted feature information to generate three scale feature maps, and the output part detects the object from these generated feature maps.

In the process of target detection and processing, the YOLO v5 algorithm adds a mosaic data enhancement function in this part of the data input. The backbone mainly adopts the Focus structure, SPP structure, and BottleneckCSP structure. Add the FPN + PAN (path aggregation network) structure to the neck. In the new version of the YOLOv5 network, the author transforms the bottleneck CSP (bottleneck layer) module into the C3 module. Its structure and function are the same as that of CSP, which includes three standard convolution layers and multiple bottleneck modules.

The difference between the C3 and CSP modules is that the Conv module after residual output is removed, and the activation function in the standard convolution module after the Concat module is also changed from LeakyRelu to SiLU. This module is the main module for learning the residual characteristics. Its structure is divided into two branches. One uses multiple bottleneck stacks and three standard convolution layers, while the other passes through only one basic convolution module, finally concatenating the two branches. In the YOLOv5s network model, the size of the image input is  $3 \times 640 \times 640$ , and the characteristic image with a size of  $12 \times 304 \times 304$  is converted through one focus slice operation, and then, it is converted into the characteristic image with a size of  $32 \times 304 \times 304$  through the ordinary convolution operation of 32 convolution cores.

*2.2. Mosaic Data Enhancement.* Mosaic [33] data augmentation is an advanced data augmentation method. The basic strategy of Mosaic data augmentation is to stitch together four security wear images and then perform data augmentation operations such as random scaling, random cropping, and random placement.

The advantages of mosaic data enhancement are as follows: there is no need to increase the size of the minibatch because Mosaic data augmentation can enrich the background and target of the security wear inspection object when calculating batch normalization. The GPU can calculate the data of four security wear images at a time, so that it can obtain a better detection effect. The security wear dataset used for detection can be significantly increased, making the network more robust. The function of the GPU can be simplified, and the performance can be greatly improved. An example of a security wear image enhanced by Mosaic data is shown in Figure 2. The images enhanced by Mosaic data are beneficial for better fitting the images in the training set during the training process. Mosaic data enhancement

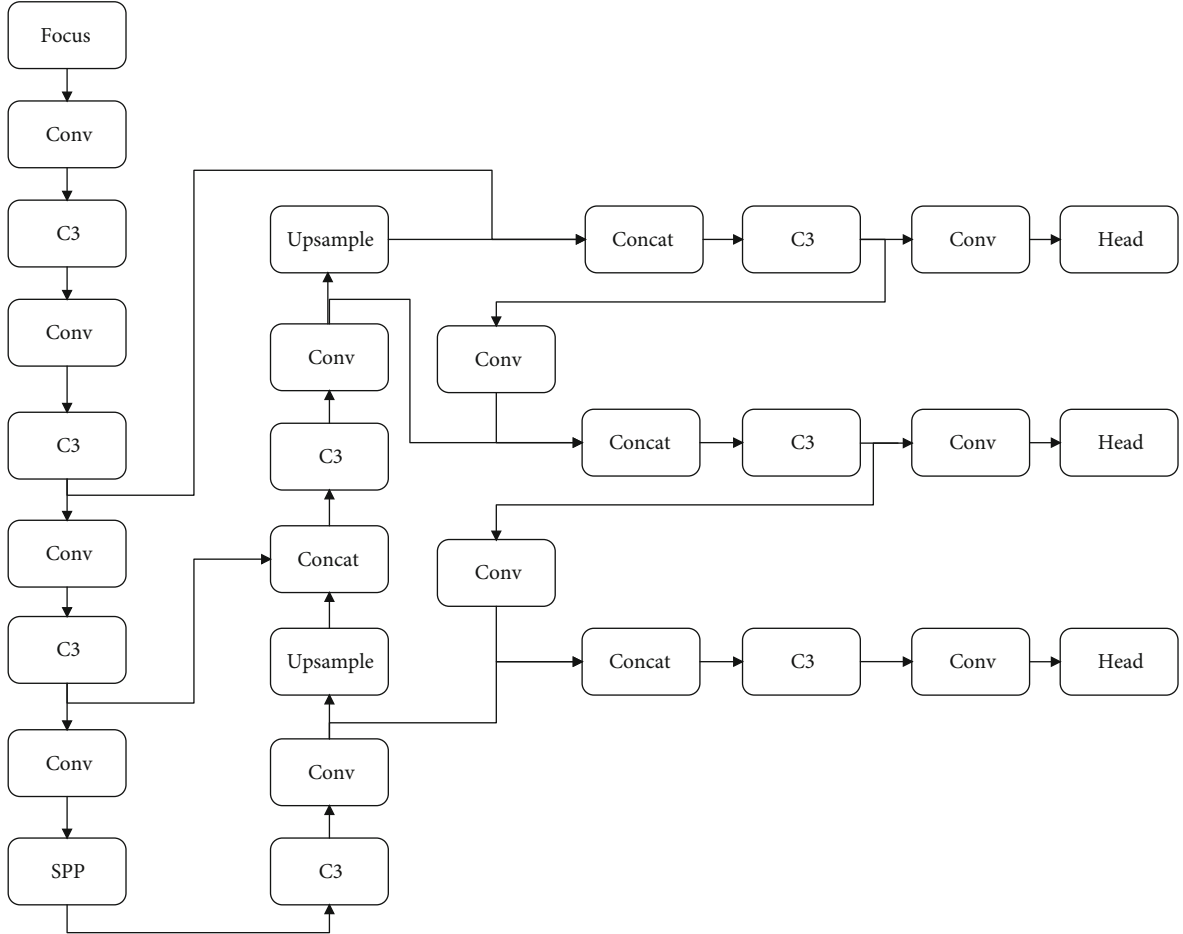


FIGURE 1: The architecture of the YOLOv5 method.

strategy is a training strategy that can improve the performance of the model, but it only needs to consume a little cost.

**2.3. Loss Function.** The calculation of loss in the YOLO series is based on objective score, class probability score, and bounding box region score. In this prediction, the loss function of the boundary anchor box is improved from CIOU (complete IOU) loss to a generalized IOU loss. The weighted NMS (Nonmaximum Suppression) operation is used to filter multiple target anchor frames. YOLO V5 uses GIOU loss as the loss of bounding box. The code uses nn.BCEWithLogitLoss or FocalLoss evaluates the class loss and confidence loss of the target frame and prediction frame. The change of a parameter will have a great impact on the loss function [34]. The formula for BCELoss is as follows.

$$\text{BCELoss} = -\frac{1}{n} \sum (y_n \times \ln x_n + (1 - y_n) \times \ln (1 - x_n)). \quad (1)$$

In Equation (1),  $y$  is the target and  $x$  is the output value of the model.

The YOLO v5 code uses the IOU index to evaluate the position loss of the target frame and prediction frame. The

YOLO v5 code selects the prediction box corresponding to the real box with the aspect ratio, and each real box corresponds to three prediction boxes. The YOLO v5 code uses the IOU value to evaluate the position loss between the prediction frame and the real frame. This paper introduces the CIOU index. Equation (2) is as follows.

$$\text{GIOU}_{\text{Loss}} = 1 - \text{CIOU} = 1 - \left( \text{IOU} - \frac{\text{Distance}_c^2}{\text{Distance}_c^2} - \frac{v^2}{(1 - \text{IOU}) + v} \right). \quad (2)$$

In Equation (2), IOU is the call union ratio of the prediction frame and the real frame.  $v$  is a parameter to measure the consistency of the aspect ratio.

### 3. Lightweight YOLOv5 Model

**3.1. Depth-Wise Separable Convolution.** Deep separable convolution is one of the methods that can miniaturize the network model at present. Depth-wise separable convolution is to decompose the standard convolution into two steps. In the first step, a convolution check should be applied to a channel, and a channel is extracted by only one convolution kernel. In the next step,  $n \ 1$  by  $1$  convolution kernels are



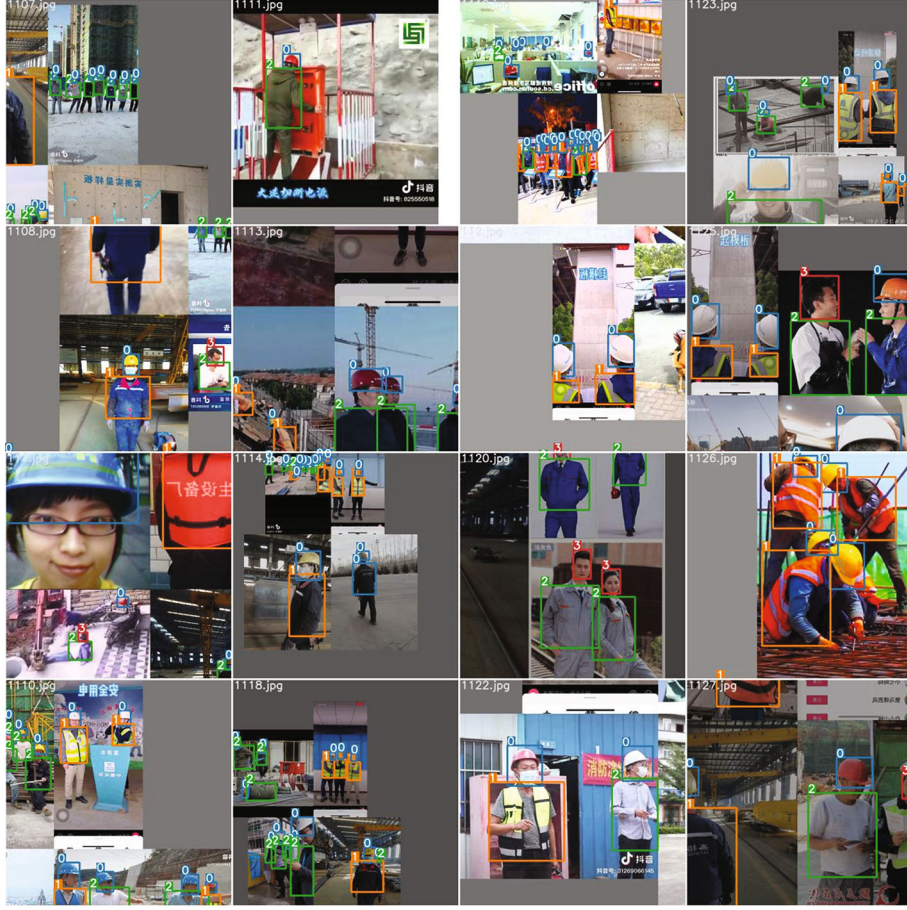


FIGURE 2: Mosaic data-enhanced image example.

used to connect the feature mapping obtained from the previous step to maintain the integrity of features [35]. The structure of deep separable convolution can reduce output channels and realize cross-channel information integration at the same time, keeping the algorithm performance unchanged while reducing the number of parameters [36]. Figure 3 shows the standard convolution and depth-wise separable convolution.

The ratio of parameters of depth-wise separable convolution to traditional standard convolution is shown in Equation (3).

$$\frac{D_k \times D_k \times M \times 1 + 1 \times 1 \times M \times N}{D_k \times D_k \times M \times N} = \frac{1}{N} + \frac{1}{D_k^2}. \quad (3)$$

In the formula,  $D_k$  is the size of the convolution kernel.  $M$  is the input channel.  $N$  is the output channel. The deep separation convolution layer is used to replace the standard convolution layer in the convolution network model, which shows the amount of calculation required to convolute the same image to obtain the same dimensional image features is greatly reduced (see Figure 3). The advantage of deep separable convolution over conventional

convolution is that it can significantly reduce the number of parameters.

In the new version of the YOLOv5 backbone, the author uses four-slice operations in the upper structure of feature extraction to form the Focus layer. The structure diagram of the Focus Layer is shown in Figure 4. For the Focus layer, every four adjacent pixels in a square generate a feature map with four times the number of channels, which is similar to the downsampling of the upper layer four times and concatenating the results. The main function is to reduce the parameters and accelerate the model without reducing the feature extraction ability of the model. When the parameters are reduced, the model is accelerated. However, there is a prerequisite for this acceleration, which can only be reflected by the use of the GPU. For this processing method of cloud deployment, the GPU does not need to consider the occupation of cache. That is, the method of fetching and processing makes the Focus layer very friendly on GPU devices. However, for chips, especially those without GPU and NPU acceleration, frequent slice operations will only seriously occupy the cache and increase the burden of computing processing. At the same time, during chip deployment, the transformation of the Focus layer is extremely unfriendly to novices. Therefore, the Focus layer is removed in this algorithm to avoid multiple slice operations.

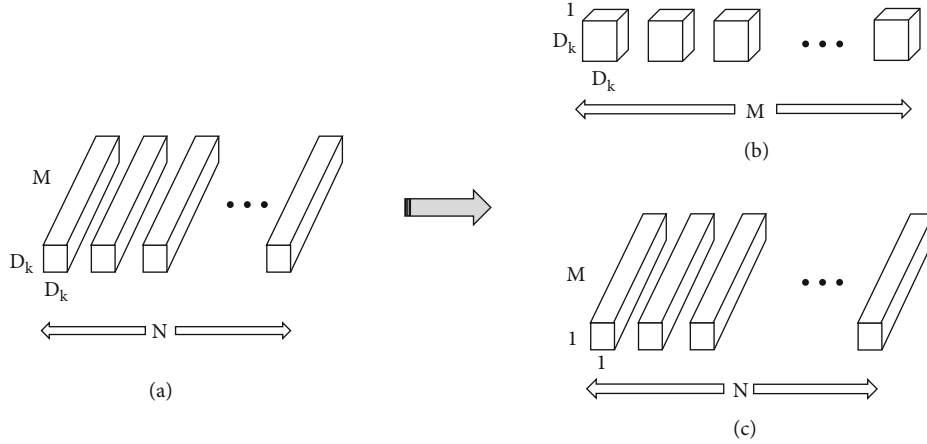


FIGURE 3: Standard convolution and depth-wise separable convolution.

C3 Layer is an improved version of CSPBottleneck proposed by YOLOv5 authors, which is simpler, faster, and lighter and achieves better results with nearly similar losses. However, the C3 Layer uses multiplexed convolution. Tests have shown that the frequent use of the C3 Layer and the C3 layer with a higher number of channels will occupy more cache space and reduce the running speed. Because of the G1 criterion of shufflenetv2 [37], the same channel size can minimize the amount of memory access. The higher the number of channels, the greater the step gap between hidden channels and c1 and c2. Imagine jumping down one step and ten steps. Although ten steps can be reached at one time, you need to run, adjust, and accumulate energy to jump up, which may take longer.

Therefore, lightweight YOLOv5 replaces the backbone of YOLOv5s with the DW network, modifies the model structure, increases the number of network layers, and improves the detection accuracy. The multiscale prediction structure of YOLOv5 is used to better detect different types of objects and improve detection accuracy. The improved network structure is shown in Figure 5. The reason for feature stitching is that the network can learn deep and shallow features at the same time, and the expression effect is better. Plots that show “CBR\_BLOCK” are used to represent Conv2d + BN + LeakyRelu6. “DW\_BLOCK” indicates deep separable convolution. The crosslink operation will be performed in “DW\_BLOCK”. If it is not the first layer, the cross-layer connection will be made. The output port of the network continues to use the output of YOLOv5s, and the  $20 \times 20$ ,  $40 \times 40$ ,  $80 \times 80$  images of three different scales are predicted.

Using SPP [38] instead of SPP [39] can reduce flops, run faster, and realize local features as well as all other features. The SPPF network structure is shown in Figure 6. The fusion of local features and full moment features is conducive to the large difference in target size in the image to be detected in the security wear recognition image and can enrich the expression ability of the feature map. Especially for the complex multitarget detection in this paper, the detection accuracy can be greatly improved.

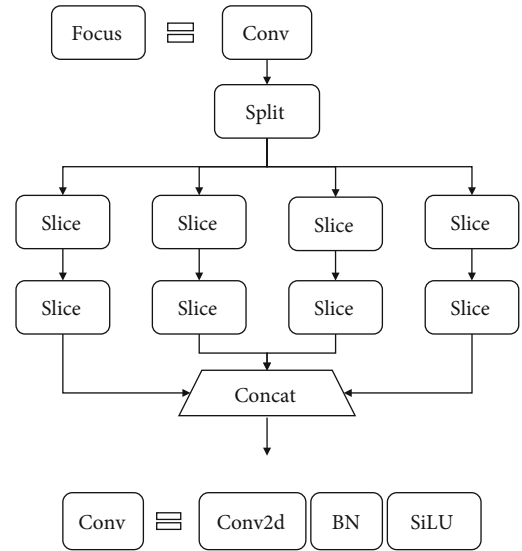


FIGURE 4: Focus structure diagram.

**3.2. Attention Mechanism Module.** The attention mechanism module [40] (convolution block attention module, CBAM) is a lightweight general module at present, focusing on channel dimension and spatial dimension and integrating two independent dimensions. It can be divided into two parts according to the spatial dimension and channel dimension. The first part is the channel attention module (CAM), and the second part is the spatial attention module (SAM). The attention mechanism module is a very simple module that can carry out end-to-end training with potential convolution at the same time to achieve good results. In two independent dimensions, the attention mechanism module can infer the attention map along different dimensions and carry out adaptive optimization based on the extraction of the feature map. In the process of attention inference, the lightweight  $1 \times 1$  convolution is used to mark the position information ignored in the feature extraction process on the convolution

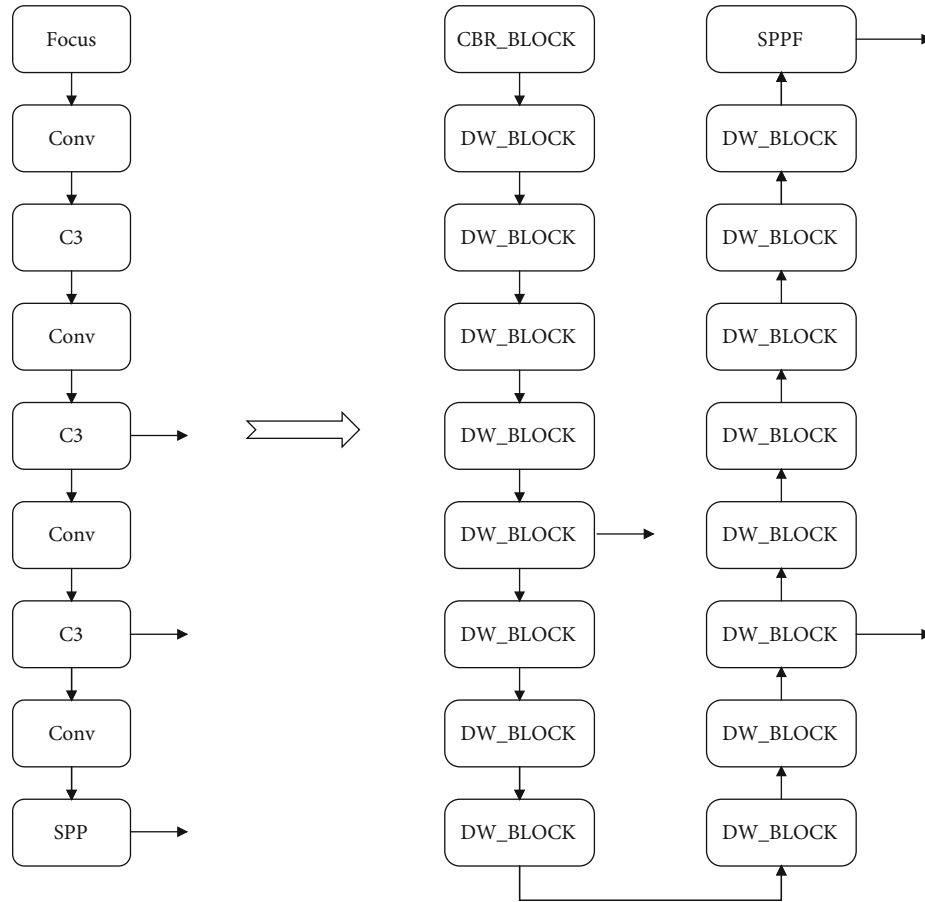


FIGURE 5: DW-YOLOv5s-BackBone.

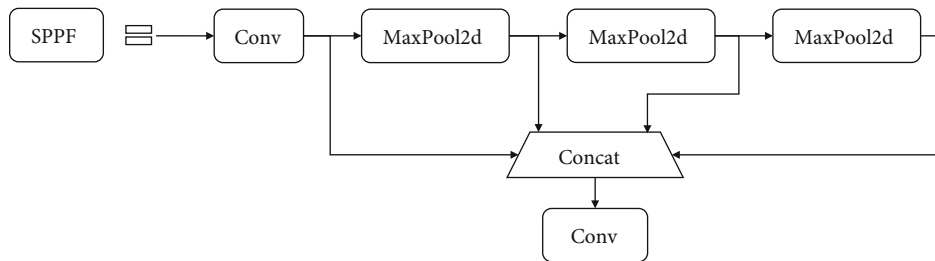


FIGURE 6: The SPPF network structure.

channel and strengthen the positioning ability of the model in the spatial dimension through the direction and feature information.

Firstly, this model uses the improved lightweight backbone network to obtain the coarse-grained target feature map, extracts its fine features in the feature fusion module, and embeds the attention mechanism CBAM to generate the attention map. Through the combination of attention map and coarse-grained features, it can enhance the feature information of security wear and realize the attention to the area of interest, that is, safety helmet and protective clothing, reducing the interference of irrelevant information on feature extraction [41].

Plots that show using DW-YOLOv5 as the basic network and adding a CBAM attention module in the neck of the network (between the backbone network and the detection layer) can better integrate the spatial features and channel features of small targets in the feature map, so as to enhance the feature information (see Figure 7). After model training, the test results are shown (see Table 1). It can be seen that after adding CBAM, the mAP of the model is increased from 80.5% to 82.1%.

After adding the attention mechanism module, there are some small gaps in training time, memory size, and detection speed of the model, but the mAP of the DW-YOLOv5-attention model is improved by nearly 1.6%.

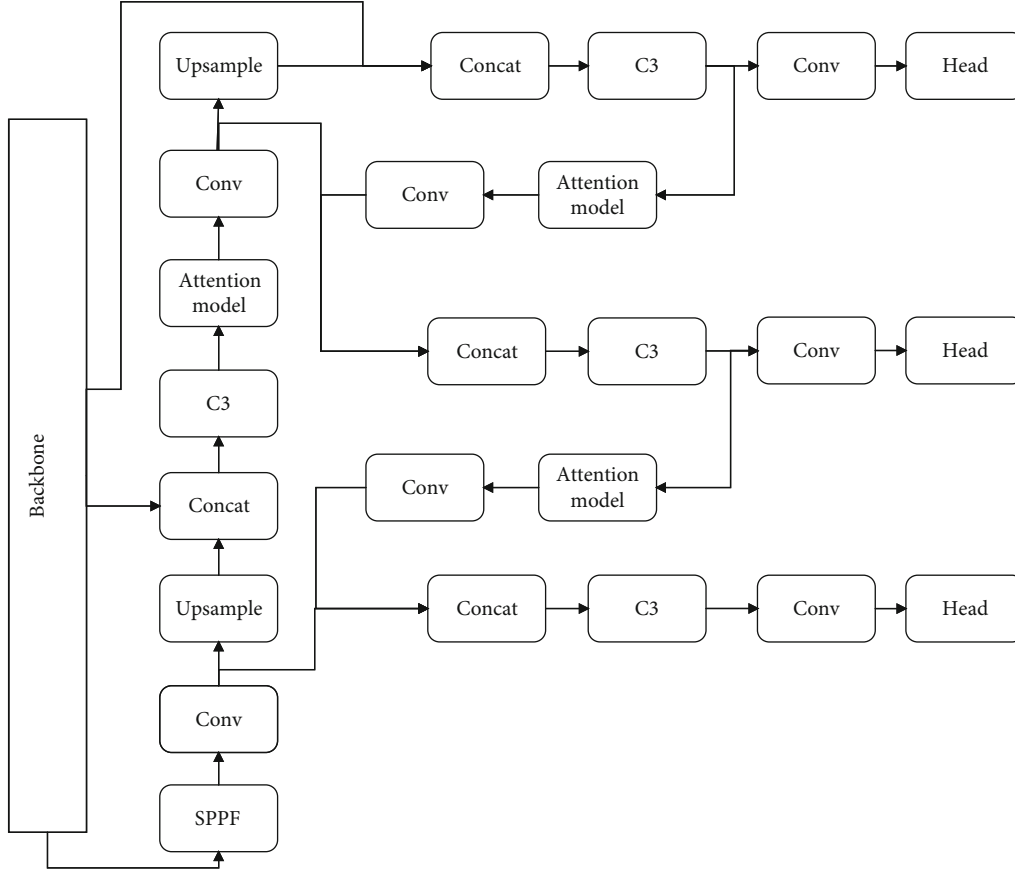


FIGURE 7: YOLOv5s- Attention-Neck.

TABLE 1: Comparison of results of multiple detection models on SWD dataset.

Model	Precision	Recall	mAP	FPS
Faster-RCNN	0.559	0.700	0.681	18.03
MobileNet-SSD	0.857	0.258	0.464	60.11
YOLO v5	0.877	0.809	0.849	83.33
DW-YOLO	0.869	0.760	0.805	90.90
DW-YOLO-attention	0.892	0.752	0.821	100.00

Experiments show that attention is applied to the model and improves the classification and detection performance of the model.

The security wear detection steps are shown in Figure 8. Using YOLO Auto Learning Bounding Box Anchors, the prior frames of these 9 scales are set in the DW-YOLO-attention module for training. Input the mosaic-enhanced image and adjust the image size, then enter the designed DW-YOLO attention module, and finally output the detected security wearing an image.

#### 4. The Datasets and Evaluation Index

In the previous research on target detection, most of the public dataset PASCAL VOC dataset was used for experiments, but the object size in this dataset is mostly 300 to

500, which is quite different from the detection object size in this paper. If only the PASCAL VOC public dataset is used for experiments, it may have a certain impact on the practical application of this algorithm. Therefore, in this experiment, the PASCAL VOC 2007 public dataset and the self-made security wear dataset (SWD) are used for training and testing. The specific dataset is described as follows.

*4.1. PASCAL VOC.* Select the training set of PASCAL VOC 2007 and VOC 2012 datasets as the training set and test set. This dataset is the benchmark dataset for evaluating image classification and target detection, including 20 types of label objects. The Train and Val datasets of VOC 2007 and 2012 are used for the training pictures, including 40025 objects of 16551 pictures, and the test dataset of the PASCAL VOC 2007 is used for the test pictures, a total of 12032 objects containing 4952 pictures.

*4.2. Security Wear Dataset (SWD).* Firstly, the monitoring video data of a factory's monitoring room for a week is collected. Considering the influence of the environment, the image information including different conditions, such as day, night, and inside and outside the factory, was selected in the construction process of the dataset. The selected scenes can ensure that the data has rich background information and that the model trained on this basis has good environmental generalization ability. By using openCV2 to



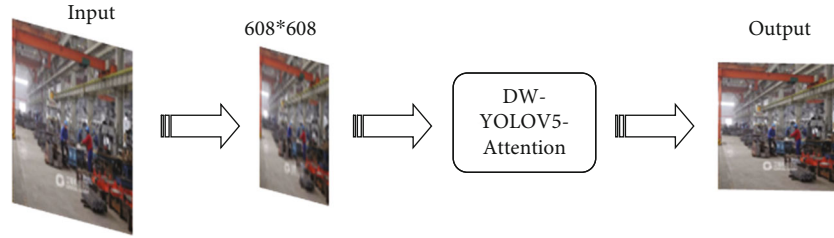


FIGURE 8: Security wear detection steps.

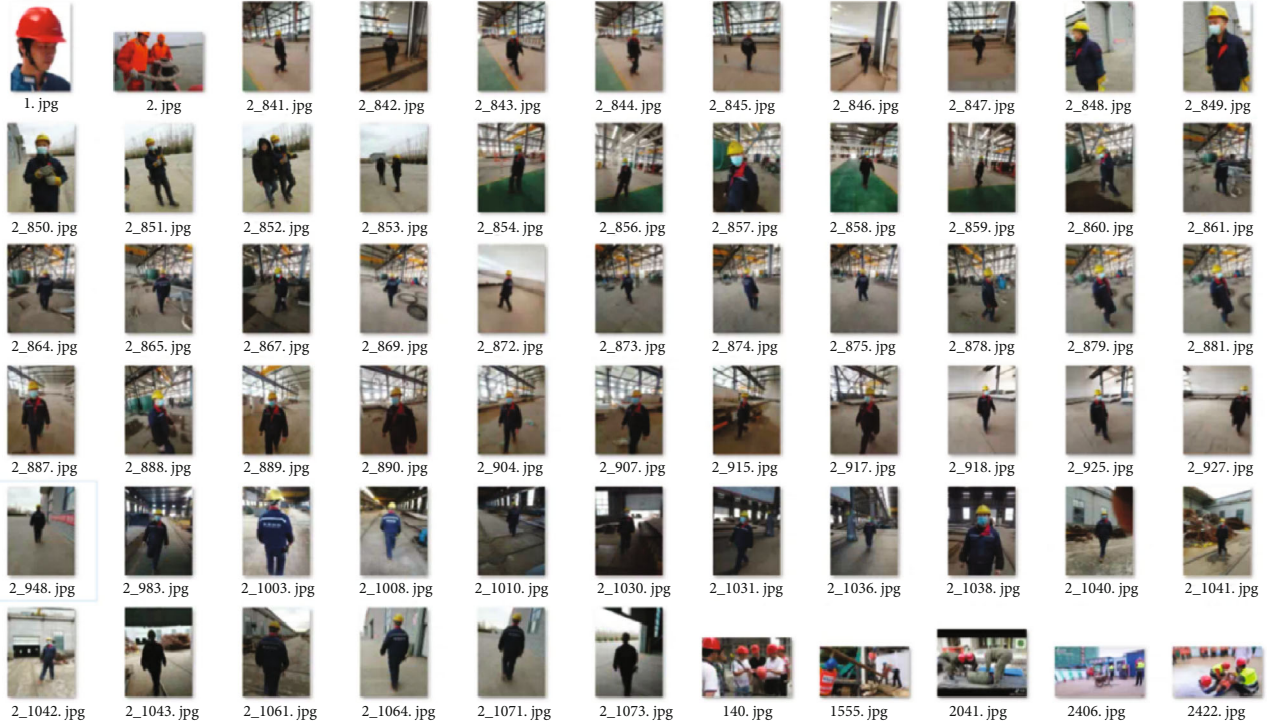


FIGURE 9: The sample of security wear datasets (SWD).

convert the video into image frames, the captured video is divided into several pictures. Considering the target size, illumination, and other factors, 3600 pictures, including helmets or heads, wearing protective clothing, and ordinary clothes, are selected. For the identification and detection of security wear, the main detection is whether the workers are wearing safety helmets and protective clothing. When the helmet is not worn, the detection category is the head. If you do not wear protective clothing, identify the category in which you are wearing ordinary clothes. According to the theory of literature [42], the sample data containing small targets can be added by replication. 110 images containing smaller targets were cut and copied from the training set. Select the open-source helmet dataset SHWD [43] (site scene helmet dataset), comprehensively consider screening 1250 pictures and adding them to the training set. After retagging with the open-source tagging tool, a dataset of 4950 security wear datasets is formed, which is made into the required YOLO dataset format. Some sample examples of security wear datasets are shown in Figure 9.

TABLE 2: Details of security wear datasets (SWD).

Label	Total
Safety_hat	8561
Reflective_cloth	4831
Other_cloth	2869
Head	926
Small( $\text{area} \leq 32 \times 32$ )	3970
Medium( $32 \times 32 < \text{area} \leq 96 \times 96$ )	7866
Large( $\text{area} > 96 \times 96$ )	5414

The SWD datasets for personnel safety protection wear on special occasions are formed, with a total of 4950 pieces, which are divided into four categories: normal wearing a safety helmet, wearing protective clothing, wearing ordinary clothes, and head without a safety helmet. The dataset contains 3907 small targets ( $\text{area} \leq 32 \times 32$ ),



TABLE 3: Details of the three datasets.

Dataset	Number of pictures	Object type	Function
PASCAL VOC 2007+12	16551	20	Training/testing
Security wear dataset (SWD)	4950	4	Training/testing

7866 medium targets ( $32 \times 32 < \text{area} < 96 \times 96$ ), and 5414 large targets ( $\text{area} > 96 \times 96$ ). The training set and verification sets are randomly allocated in the proportion of 7: 3. The number of instances include in each category is shown in Table 2.

The detailed information of VOC dataset and SWD data involved in this experiment (see Table 3).

4.3. *Evaluation Indexes.* In this paper, Precision and Recall [44] were selected as the evaluation indexes of this experiment.

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (4)$$

$$\text{Recall} = \frac{TP}{TP + FN}, \quad (5)$$

In Equations (4) and (5),  $TP$  is the probability that positive examples can be divided into pairs.  $FP$  is the probability of misclassifying negative cases into positive cases.

mAP is used to measure the accuracy of the detection model of personnel safety protection wear on special occasions. The mAP refers to the average value of the accuracy rate of all categories, the average value of AP of each category. For the test results of each specific category, sort according to the confidence level, and select each recall rate  $r_i$  and its corresponding maximum precision  $P_i$  in the recall rate set  $\{r_0, r_1, \dots, r_n\}$ . The definition of AP is shown in Equation (6):

$$\text{AP} = \sum_{i=1}^n P_i(r_i - r_{i-1}), \quad (6)$$

$$\text{mAP} = \frac{\sum_{i=1}^m \text{AP}_i}{m}. \quad (7)$$

In Equation (7), the numerator is the average accuracy of each category, and  $m$  is the total number of categories of image detection.

FPS (frames per second) is used to measure the running efficiency of the model, that is, the number of pictures processed per second. Size measures the storage space occupied by the model, and FLOPs (floating-point operations) measures the complexity of the model.

## 5. Experimental Results and Discussion

5.1. *Experiment Operating Environment.* The experimental environment of this experiment is mainly carried out under the computer configuration of the Windows 10 operating

system, Intel Core i7-8700k, 3.70 GHz, and 16 G RAM. The GPU adopts NVIDIA RTX 2080 and 16G video memory. The experimental conclusions of this model and its comparative model are drawn in this experimental configuration environment.

Hyperparameters	Default
Input size	640
Lr	0.01
Lr_f	0.2
Momentum	0.937
Weight_decay	0.0005
Warmup_epochs	3.0
Warmup_momentum	0.8
Warmup_bias_lr	0.1

TABLE 5: Object detection model performance comparison.

Model	FLOPs	Parameters	MB
YOLOv5s	16.5	7114785	14.5
DW-YOLO	7.4	3820257	7.9
DW-YOLO-Attention	7.5	3871751	8.0

TABLE 6: Comparison of different algorithms on PASCAL VOC dataset.

Algorithms	Input size	mAP/%
MobileNet-SSD	$300 \times 300$	68.0
faster-RCNN	$1000 \times 600$	70.0
YOLOv5s	$640 \times 640$	81.6
DW-YOLO	$640 \times 640$	77.8
DW-YOLO-attention	$640 \times 640$	77.5

system, Intel Core i7-8700k, 3.70 GHz, and 16 G RAM. The GPU adopts NVIDIA RTX 2080 and 16G video memory. The experimental conclusions of this model and its comparative model are drawn in this experimental configuration environment.

5.2. *Hyperparameter Setting.* The super parameter setting of the lightweight YOLO v5model is given. The training algebra of this experiment is 300 generations and the batch size is 18, the input size is 640. The initial momentum and initial learning rate (lr) are set to 0.937 and 0.01, respectively. Before the formal training in this paper, three generations of preheating learning are carried out, in which the preheating learning momentum is 0.8 and the Warmup lr is 0.1. The purpose is to make the model gradually stabilize after preheating learning and then carry out formal training. The effect of security wear recognition is better. The other super-parameter settings are shown in Table 4.

5.3. *Experimental Results and Analysis.* Firstly, it analyzes the size of the model and compares it with the original YOLOv5 model. There are three measurement indexes

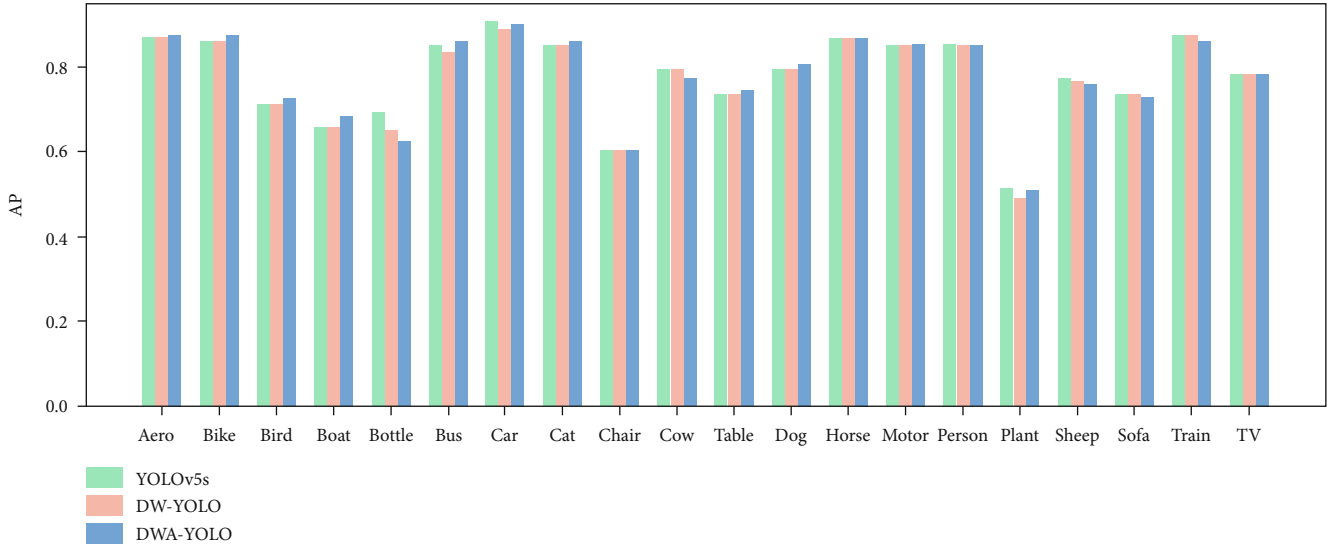


FIGURE 10: Comparison of different types of AP indexes in VOC datasets.

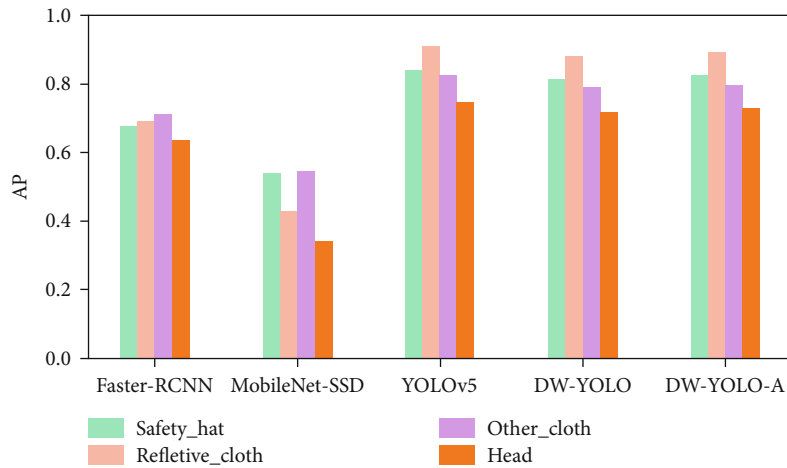


FIGURE 11: Comparison of different types of AP indexes in SWD datasets.

for the improved models (see Table 5). The memory required for lightweight DW-YOLO-Attention is 8.0 MB, the parameters are  $7.5 \times 10^5$ , and the FLOPs are 7.5. Compared with the YOLOv5 model, the memory required is reduced by 44.8%, respectively. The parameters were reduced by 45.58%. FLOPs are reduced by 54.54%, which reduces the complexity of the model and shows the superior performance of the lightweight DW-YOLO-Attention model in the field of security wear.

To fully evaluate the lightweight DW-YOLO-Attention detection algorithm in this paper, three commonly used detection algorithms are used for comparative experiments, and the mAP is selected as the evaluation index of different detection algorithms. The open datasets for model training are PASCAL VOC 2007+2012, and the test is PASCAL VOC 2007. The results are shown in Table 6.

Table 6 shows that compared with the advanced model algorithm, the mAP of the lightweight DW-YOLO model

algorithm is good. The detection mAP can reach 77.8%, only 3.8% points lower than the baseline, 7.8% higher than the Faster-RCNN, and 9.8% points higher than the MobileNet-SSD. By improving the feature extraction backbone of DW-YOLOv5, the number of layers of the improved lightweight backbone network is deepened, and the extracted security wear image features become more detailed. The deep separation convolution ensures that the detection accuracy of the model is improved, but the number of model parameters is not increased, so a better detection effect is achieved.

Then, it compares the AP of DW-YOLOv5s and DW-YOLO-Attention in each category of the VOC dataset (see Figure 10). Plots that show DW-YOLO have high detection accuracy for cat, dog, table, bus, bird, etc., which shows that this model has a good recognition effect for objects of different sizes, especially small and medium-sized objects. It applies to the security wear detection in this paper.

In this paper, the improved lightweight YOLOv5 algorithm is applied to the identification of safety wear protection. To verify the better effect of the method proposed in this paper, under the same equipment and network parameter configuration, the same number of test sets is used, and several popular one-stage and two-stage target detection networks are used for the experiments: Faster-RCNN and MobileNet-SSD. The experimental results were evaluated by four evaluation indexes: Precision, Recall, mAP, and FPS. The experiment is shown in Table 1.

The table shows the results of different models in security wear recognition. It can be seen that on the SWD dataset, the mAP of the lightweight DW-YOLO-Attention proposed can reach 82.1%. Compared with Faster-RCNN and MobileNet-SSD increases by 20.55% and 76.93%, and decreases by about 3.52% compared with YOLO v5. The FPS of the lightweight DW-YOLO-Attention model is 100. Compared with Faster-RCNN and MobileNet-SSD, the FPS of DW-YOLO-attention is increased by 81.97 and 39.89. Therefore, the lightweight DW-YOLO improves the real-time performance of security wear recognition while maintaining a high mAP of 82.1%, which has a certain practical application value for security wear recognition. Compared with other algorithms, the lightweight DW-YOLO-Attention model in this paper has certain generalization and robustness.

Then, we compared the AP of Faster-RCNN, MobileNet-SSD, YOLOv5, DW-YOLO, and DW-YOLO-Attention in the security wear dataset (see Figure 11). The recognition performance of these five target models is not satisfied in the case of occlusion in the factory scene. It may even be misjudged due to factors such as ambient light. The accuracy of the above methods decreases when they detect small targets. By contrast, both the lightweight DW-YOLO and DW-YOLO-Attention are able to detect small objects in the images on the same test set. Both of them have higher detection accuracy on objects with larger targets and more obvious features like the protective clothing (Reflective\_cloth) but have relatively lower accuracy on smaller targets with complex features like the head. The reason lies in the fact that DW-YOLO-Attention which updates YOLO with model lightweight is able to improve running speed and reduce the model training time. The above experimental results have demonstrated that the proposed DW-YOLO-Attention algorithm has competitive performance in reducing the missed detection rate of small and medium-sized targets. Notably, the accuracy and detection speed of the method are not degraded when DW-YOLO-Attention is used to detect small and medium-sized targets. Consequently, the proposed method can be applied in practical scenarios, and we have successfully utilized it in the factory for object detection.

## 6. Conclusions

In existing security wear detection, the model transplantation is challenging to perform on the embedded platform since the number of network parameters is huge, and the low computation power of the platform also triggers the

degradation of the detection accuracy. Accordingly, we have modified the YOLOv5 network structure to handle the safety wear detection with three scales. In other words, a lightweight target detection network based on deep separable convolution is proposed. In our model, we replaced ordinary convolution with depth-separable convolution in order to reduce the number and scale of parameters. The attention part is used to weigh the different channels of feature mapping to improve the detection accuracy of the model. The experimental results have shown that the proposed model has higher accuracy in target detection and lower model calculations than the existing models. Especially, the proposed model has the fastest reasoning speed among all models. The line of our future research is to improve the ability of the proposed model to detect objects under employee occlusion and extreme shooting angles.

## Data Availability

The code of the proposed algorithm and corresponding experimental data are provided. For interested readers, please visit <https://github.com/lstttt/projectnewone>.

## Conflicts of Interest

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

## Acknowledgments

This work was supported in part by the Fundamental Research Funds for the Central Universities under the grant no. N2117005, the Joint Funds of the Natural Science Foundation of Liaoning Province under grant no. 2021-KF-11-01, the Fundamental Research Funds for the Central Universities, the National Natural Science Foundation of China under the grant no. 62103150, and the project funded by China Postdoctoral Science Foundation under the grant no. 2021 M691012.

## References

- [1] T. Li, S. Xu, and Z. Yao, "Adaptive dim and weak target detection method based on DSP," *Computer Applications and Software*, vol. 35, no. 1, pp. 243–245, 317, 2018.
- [2] X. Xue and J. Zhang, "Matching large-scale biomedical ontologies with central concept based partitioning algorithm and adaptive compact evolutionary algorithm," *Applied Soft Computing*, vol. 106, article 107343, 2021.
- [3] L. Ma, X. Wang, M. Huang, Z. Lin, L. Tian, and H. Chen, "Two-level master-slave RFID networks planning via hybrid multiobjective artificial bee colony optimizer," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 5, pp. 861–880, 2019.
- [4] X. Xue, J. Lu, and J. Chen, "Using NSGA-III for optimising biomedical ontology alignment," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 3, pp. 135–141, 2019.
- [5] Q. He, X. Wang, Z. Lei, M. Huang, Y. Cai, and L. Ma, "TIFIM: a two-stage iterative framework for influence maximization in

- social networks,” *Applied Mathematics and Computation*, vol. 354, pp. 338–352, 2019.
- [6] X. Xue, J. Chen, and X. Yao, “Efficient user involvement in semiautomatic ontology matching,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 5, no. 2, pp. 214–224, 2018.
- [7] R. Usukhbayar and J. Choi, “Critical safety factors influencing on the safety performance of construction projects in Mongolia,” *Journal of Asian Architecture and Building Engineering*, vol. 19, no. 6, pp. 600–612, 2020.
- [8] G. Feng, Y. Chen, and N. Chen, “Research on automatic recognition technology of safety helmet based on machine vision,” *Mechanical Design and Manufacturing Engineering*, vol. 44, no. 10, pp. 39–42, 2015.
- [9] J. Liu, S. Hou, and K. Zhang, “Real time vehicle detection and tracking based on enhanced tiny-YOLOv3 algorithm,” *Transactions of the Chinese Society of Agricultural Engineering*, vol. 35, no. 8, pp. 118–125, 2019.
- [10] L. Ma, X. Wang, X. Wang, L. Wang, Y. Shi, and M. Huang, “TCDA: truthful combinatorial double auctions for mobile edge computing in industrial Internet of Things,” *IEEE Transactions on Mobile Computing*, vol. 3064314, p. 1, 2021.
- [11] M. Li, Q. Han, and T. Zhang, “Safety helmet detection method of improved SSD,” *Computer Engineering and Applications*, vol. 57, no. 8, pp. 192–197, 2021.
- [12] Q. Li, *Research and implementation of helmet video detection system based on human body recognition*, Chengdu University of Electronic Science and Technology of China, 2017.
- [13] H. Liu and X. Ye, “Skin color detection and Hu moments in helmet recognition research,” *Journal of East China University of Science and Technology*, vol. 40, no. 3, pp. 365–370, 2014.
- [14] G. Li, X. Zhang, and F. Qin, “Paper cut pattern recognition based on moment invariants and BP neural network,” *Computer Engineering and Application*, vol. 46, no. 29, pp. 158–160, 2010.
- [15] M. W. Park, N. Elsafty, and Z. Zhu, “Hardhat-wearing detection for enhancing on-site safety of construction workers,” *Journal of Construction Engineering and Management*, vol. 141, no. 9, p. 4015024, 2015.
- [16] A. Rubaiyat, T. Toma, M. Kalantari-Khandani et al., “Automatic detection of helmet uses for construction safety,” in *IEEE/WIC/IACM International Conference on Web Intelligence Workshops (WIW)*, pp. 135–142, Omaha, NE, USA, 2016.
- [17] L. Ma, S. Cheng, and Y. Shi, “Enhancing learning efficiency of brain storm optimization via orthogonal learning design,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 11, pp. 6723–6742, 2021.
- [18] Y. Huang and D. Pan, “Helmet recognition based on parallel two-way convolutional neural network,” *Enterprise Technology Development*, vol. 37, no. 3, pp. 24–27, 2018.
- [19] Q. Fang, H. Li, X. Luo et al., “Detecting non-hardhat-use by a deep learning method from far-field surveillance videos,” *Automation in Construction*, vol. 85, pp. 1–9, 2018.
- [20] B. Zhang, Y. Song, and R. Xiong, “Helmet wearing detection integrating human joint points,” *Chinese Journal of Safety Science*, vol. 30, no. 2, pp. 181–186, 2020.
- [21] Y. Zhang and X. Xu, “Helmet wearing detection method based on improved SSD,” *Electronic Measurement Technology*, vol. 43, no. 19, pp. 80–94, 2020.
- [22] M. Fang, T. Sun, and Z. Shao, “Fast helmet-wearing-condition detection based on improved YOLOv2,” *Optical Precision Engineering*, vol. 27, no. 5, pp. 1196–1205, 2019.
- [23] M. Zhang, Z. Cao, and X. Zhao, “Research on helmet wearing recognition of construction workers based on deep learning,” *Journal of Safety and Environment*, vol. 2, pp. 535–541, 2019.
- [24] Q. Wang, *Research on safety helmet wearing recognition of workers in construction site based on video stream*, Huazhong University of Science and Technology, 2018.
- [25] W. Liu, D. Anguelov, D. Erhan et al., “SSD: Single Shot Multi-box Detector,” *Proceedings of the European Conference on Computer Vision*, pp. , 201621–37, 2016.
- [26] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.
- [27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: unified, real-time object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, Nevada, USA, 2016.
- [28] J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271, Honolulu, Hawaii, USA, 2017.
- [29] J. Redmon and A. Farhadi, “YOLOV3: an incremental improvement,” in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–6, Salt Lake City, Utah, USA, 2018.
- [30] A. Bochkovskiy, C. Y. Wang, and H.-Y. M. Liao, “YOLOv4: optimal speed and accuracy of Object Detection,” 2020, <http://arxiv.org/abs/2004.10934>.
- [31] L. Ma, N. Li, Y. Guo et al., “Learning to optimize: reference vector reinforcement learning adaption to constrained many-objective optimization of industrial copper burdening system,” *Cybernetics*, pp. 1–14, 2021.
- [32] S. Wu, G. Li, L. Deng et al., “L1-norm batch normalization for efficient training of deep neural networks,” *The IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 7, pp. 2043–2051, 2019.
- [33] T. Lee and L. Luo, “Mosaic analysis with a repressible cell marker for studies of gene function in neuronal morphogenesis,” *Neuron*, vol. 22, no. 3, pp. 451–461, 1999.
- [34] L. Ma, M. Huang, S. Yang, R. Wang, and X. Wang, “An adaptive localized decision variable analysis approach to large-scale multiobjective and many-objective optimization,” *IEEE Transactions on Cybernetics*, vol. PP, pp. 1–13, 2021.
- [35] R. Qi, R. Jia, Q. Mao, H. M. Sun, and L. Q. Zuo, “Face detection method based on cascaded convolutional networks,” *IEEE Access*, vol. 7, pp. 110740–110748, 2019.
- [36] J. Bai, P. Hao, and S. Chen, “Traffic scene understanding using lightweight convolution neural network image semantic segmentation,” *Journal of Automobile Safety and Energy Saving*, vol. 9, no. 4, pp. 433–440, 2018.
- [37] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, “Shufflenet V2: practical guidelines for efficient CNN architecture design,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 116–131, Munich, Germany, 2018.
- [38] GitHub, “YOLOV5-Master,” 2021, <https://github.com/ultralytics/YOLOv5.git/>.
- [39] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

- [40] S. Woo, J. Park, J. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 3–19, Munich, Germany, 2018.
- [41] X. Xue and C. Jiang, "Matching sensor ontologies with multi-context similarity measure and parallel compact differential evolution algorithm," *IEEE Sensors Journal*, vol. 21, no. 21, pp. 24570–24578, 2021.
- [42] M. Kisantal, Z. Wojna, J. Murawski, J. Naruniec, and K. Cho, "Augmentation for small object detection," 2019, <http://arxiv.org/abs/1902.07296>.
- [43] GitHub, "Safety-Helmet-Wearing-Dataset," 2021, <https://github.com/njvisionpower/Safety-Helmet-Wearing-Dataset/>.
- [44] I. Melamed, R. Green, and J. Turian, "Precision and Recall of Machine Translation," *Companion Volume of the Proceedings of HLT-NAACL 2003-Short Papers*, pp. 61–63, 2003.



## Research Article

# A Robust Pupil Localization via a Novel Parameter Optimization Strategy

Wenjun Zhou , Xiaoyi Lu, and Yang Wang 

*School of Computer Science & Technology, Southwest Petroleum University, 61000 Chengdu, China*

Correspondence should be addressed to Yang Wang; wangyang@swpu.edu.cn

Received 20 December 2021; Revised 7 March 2022; Accepted 6 April 2022; Published 6 May 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Wenjun Zhou 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 iris biometrics, the pupil recognition method is widely used in the fields of identity recognition and so on. Most traditional iris recognition algorithms use an adaptive threshold to the eye image binarization processing; although this can at the maximum preserve the original characteristics of the image itself, the retained features in this method included a lot of noise points, including Gaussian noise and so on, in a series of image preprocessing. The subsequent iris recognition and pupil positioning are still prone to the loss of positioning accuracy, which cannot separate the pupil completely from the background area of the image, increasing the difficulty of connecting the pupil to domain localization, thus affecting the pupil positioning accuracy. Therefore, in order to meet the requirement of high precision pupil recognition accuracy in low-quality eye images, this paper takes the iris dataset provided by the China Research Institute of Automation as an example to improve the traditional pupil location method. A pupil localization method based on parameter optimization is proposed, which includes the setting of a three-step threshold, called as TST.

## 1. Introduction

With the innovation and iteration of iris biometrics, iris recognition technology has gradually replaced the traditional identification method. Traditional identification solves the problem of identity identification with regard to the possession of objects through the possession of these external objects to prove the person's identity. However, for keys, ID cards, bank cards, and other external objects, they can be easily stolen by others who can use these items to obtain benefits. Biometric identification technology [1] is based on automatic identification technology of personal unique physiological or behavioral characteristics. The biological characteristic is a more reliable and convenient identification method than traditional identification methods. It will not be lost or forgotten; relative to the straightforward replication of the characteristics of fingerprint recognition, the iris and pupil are not easily retained after being used. Therefore, the iris and pupil features of humans are not easily copied and used by others. At the same time, due

to the innate differences between human individuals [2] and the differences in living environment between different individuals, the physiological structures of different individuals have obvious differences over time. In addition, the human iris and pupil are highly stable in biological morphology [3], so they can be used for individual identification [4]. Reliable iris recognition depends on accurate pupil location, so pupil recognition is of great practical significance. However, there are also some difficulties in pupil localization. For example, in the process of obtaining the pupil image dataset, changing the lighting conditions, eyelashes and pupil images, glasses, and contact lenses will produce a large number of different forms of reflection, as well as interference when the pupil is off-center or on the edge of the image, and blurred pupil due to blinking. These disturbances make pupil localization under nonideal conditions a practical and challenging challenge.

Traditional pupil localization methods are mainly divided into threshold and Hough transform methods [5]. The basic idea of the pupil localization algorithm based on

the threshold is to calculate binarization parameters according to sliding window, then compare the gray threshold with each pixel value in the gray image and classify each pixel value into the appropriate category. The primary threshold selection methods include the adaptive threshold method [6] and the global threshold method, among which the adaptive threshold method adopts a sliding window. By sliding the sliding window in the image, the corresponding threshold of each region is obtained, and the image binarization is carried out. This method can retain the original texture features in the image. However, it also retains details other than the pupil in the image, which significantly increases the difficulty of extracting the contour. Therefore, the pupil localization accuracy is low in nonideal cases. The global threshold method mainly compares the pixel value in the whole image by setting a fixed threshold value. It sets the pixel value more significant than the fixed threshold value as 255 and the pixel value less than the fixed threshold value as 0. Standard representative algorithms include the OTSU (OTSU method) and iterative methods. However, selecting a global threshold in the global threshold method is challenging to some extent. When the threshold is large, the whole eye region in the eye image may be classified as the target region, but the pupil and background cannot be separated accurately. When the threshold value is small, the pupil area may be incomplete so that the pupil cannot be accurately located.

To overcome the above problems, this paper proposed a robust pupil localization via a novel parameter optimization strategy called TST, which has already been described in [7]. And in order to describe TST in more detail, this paper added more details and designed more experiments for verification.

The main work of this paper is as follows:

- (1) A new threshold adaptive pupil localization method is proposed, which shifts the focus of pupil localization to the acquisition of binarization parameters, thus simplifying the subsequent screening of pupil contour point sets and ultimately improving the accuracy of pupil localization
- (2) In order to solve the small gray value difference between the pupil and the iris, which prevent us from using methods such as the histogram to obtain more accurate pupil edge gray information, a three-step threshold method is proposed, by positioning the pupil edge gray-level information within and outside the edge of the gray-level information, with the method of weighting parameters, for the edge of the pupil parameter information. In this way, the interference of external background information is removed. In contrast, the complete pupil information is retained, which ultimately improves the robustness of pupil location
- (3) A method for pupil contour recognition is proposed, which introduces screening criteria such as pupil area ratio, length-to-width ratio of the whole image, and Euclidean distance between the pupil edge and

the center coordinate to simplify the pupil contour point set identification difficulty and increase the accuracy of pupil recognition

The rest of this paper is organized as follows: In Section 2, this paper will outline the algorithms of pupil recognition. Section 3 introduces the basic principles and steps of TST. Section 4 presents experiments on three datasets and uses the results to verify our algorithm. Section 5 discusses the performance of TST algorithm. Finally, Section 6 is the conclusion of this article.

## 2. Related Work

Pupil recognition is very similar to iris recognition. Daugman [4] first proposed a method for locating the inner edge of the iris based on differential operators. Then, Wildes [2] proposed to decompose the eye image into four regions of different degrees and calculate the similarity to achieve iris recognition. On the basis of the above two methods, a variety of similar pupil localization methods have been extended. Zhang et al. [8] improved the calculus operator proposed by Daugman and proposed to use polar coordinate transformation to represent the pupil area by using a matrix in the iris region in the original image while adding Laplacian noise to the iris and pupil region in the image. Then, a deidentification algorithm is used to identify the biometrics of the iris and pupil, achieving a similar pupil recognition effect. However, this method focuses on pupil privacy protection and biometric recognition of the iris and pupil and cannot accurately locate the pupil and iris region. By analyzing calculus operators, Minakova and Petrov [9] proposed an operator based on the Bresenham algorithm to integrate partial calculation methods to improve efficiency.

Meanwhile, the Bresenham algorithm was modified to calculate operators in a single arc. This method improved the speed and accuracy of pupil positioning. However, for the eye image with no apparent difference between the target and background, the segmentation parameters cannot be calculated accurately, resulting in a large error in pupil location. In addition, Chen et al. [10] and Setiawan et al. [11] improved the traditional Hough transform circle detection method by introducing the Canny edge detection operator and proposed a pupil localization method based on the circular Hough transform. This method can achieve relatively accurate localization of artificially blocked pupils in high-quality images. And Shang et al. [12] also built a pupil localization system based on this method. However, in low-quality eye images, due to the Hough transform circle detection method, there may be multiple round areas similar to pupils in the image, increasing pupil positioning error. In order to solve the problem that pupil localization is prone to failure in the case of off-axis occlusion, Dewi et al. [13] proposed an ellipse-fitting and a fine-adjustment algorithm for robust pupil localization in off-axis conditions. The accuracy of this method was 0.83 when the Z-value was 41.5. To improve the efficiency of pupil positioning, Li et al. [14] and Jan et al. [15] proposed to adopt the grayscale integral projection method for pupil localization.

However, these methods improve the speed and reduce the accuracy.

In addition, in the practical application of pupil localization, the acquisition of the eye image is not smooth. The obtained eye image also has a lot of noise, such as pupil and iris occlusion due to strong light. These low-quality eye images will directly lead to the decrease of the pupil localization accuracy of the above methods. For this reason, Fuhl et al. [16] proposed a histogram-based pupil localization algorithm, which improved the robustness of pupil localization by combining edge filtering with the angle integral projection function. Fusek and Dobeš [17] proposed to use a self-organizing migrating algorithm (SOMA) to determine the correct shape and position of the pupil model by considering the physiological characteristics of the eyes. This method improves the speed of pupil recognition while ensuring high accuracy. Jamaludin et al. [18] proposed a deblurring method based on the Wiener filter to improve the quality of iris pattern and achieve pupil positioning. Later, literature [19] proposed using the geodesic distance to locate the pupil. This method achieves pupil location by calculating the geodesic distance of the four corners of the image, which has good stability in complex images.

In recent years, with the integration of deep learning and image segmentation, relevant scholars have also tried to use a neural network to achieve pupil location. The precision of pupil and iris segmentation can be improved by the high-precision calculation of computers. Jalilian and Uhl [20] proposed a fully connected coding network for iris segmentation. Yang et al. [21] and Choi et al. [22] proposed a network model combining fully connected convolutional networks and cavity convolution to segment the iris and achieved good experimental results. Choi et al. [23] proposed adopting a heterogeneous CNN model for pupil localization, which also achieved high recognition accuracy in specified datasets. In order to improve the accuracy of iris recognition again, Lee et al. [24] proposed to use an adversarial network for data enhancement, thus achieving more accurate pupil recognition.

However, the pupil localization method based on a convolutional neural network needs to learn specific data features. It is not easy to achieve target localization for the target features that have never been learned. In this paper, the pupil location method based on the global threshold method is improved, and a robust localization via a novel parameter optimization strategy is proposed, which includes a three-step parameter optimization, referred to as TST. The algorithm is used to accurately extract the target area and achieve complete separation of the pupil from the background, as shown in Figure 1. In biometrics, the pupil and iris are clearly demarcated, as shown in Figure 2. It can be seen from Figure 2 that the pupil region represented by the red ellipse is significantly different from the iris region represented by the yellow ellipse. There is also a significant difference between the skin area outside the orange ellipse and the iris area.

Therefore, we design a binary parameter search algorithm with pupil edge gray value jump. The initial search coordinates are determined by setting the region of interest.

The gray parameters of the outer and inner edges of the pupil were obtained to ensure the integrity of pupil image information. Then, linear interpolation was used to narrow the difference between inner and outer edges, and the binarization parameters suitable for pupil separation were obtained.

### 3. Methodology

This section improves the traditional threshold pupil localization algorithm and proposes a method of pupil localization via a novel parameter optimization to locate the pupil in eye images. After preprocessing the image, we first proposed a binarization parameter acquisition method based on a three-step threshold to obtain accurate pupil boundary segmentation parameters. The method mainly includes an  $L$ -nearest neighbor domain search, binarization parameter optimization, and boundary value. Secondly, we designed a screening method suitable for the pupil contour point set to increase the accuracy of pupil location. Finally, we combine the binary parameter acquisition method based on a three-step threshold with the pupil contour point set screening method to achieve accurate pupil location.

**3.1. Image Preprocessing.** Since there may be RGB images in the images we process, we adopt a weighted average method to grayscale the images, and the formula is as follows:

$$\text{Gray}(x, y) = \alpha R(x, y) + \beta G(x, y) + \gamma B(x, y), \quad (1)$$

where Gray is the final pixel value;  $R$ ,  $G$ , and  $B$  are the pixel values of the corresponding channels;  $(x, y)$  is the coordinates of row  $x$  and column  $y$  in the image; and  $\alpha$ ,  $\beta$ , and  $\gamma$  are the weight parameters. After graying the image, there is still a lot of abrupt point noise in the dataset. In order to reduce the influence of such noise on pupil positioning, the algorithm adopts Gaussian filtering to denoise the image, and the formula is as follows:

$$G(x, y) = \frac{1}{\sqrt{2\pi}\delta} e^{-x^2/2\delta^2}, \quad (2)$$

where  $\delta$  is the standard deviation and  $x$  is the pixel value.

**3.2. ROI Region Definition.** In biological morphology, the ratio is between 4/1 and 3/1 for the iris to the pupil. Meanwhile, due to the acquisition of the dataset including some face regions such as eyelids and eyebrows, we define the initial ROI region as 1/4 of the image center, as shown in Algorithm 1.

where  $(X_{\text{center}}, Y_{\text{center}})$  is the center point of the original image; RangeH is the high range of ROI; RangeW is the wide range of ROI; and  $A, B, C, D$  are the coordinates of the four vertices of ROI as shown in Figure 3.

**3.3. L-Nearest Neighbor Domain Search.** Unlike the gray-level histogram method and iteration method, TST searches for a way to determine the final binarization parameters; in a more sophisticated image, simply using a histogram or iterative method to get the threshold, the pupil cannot be

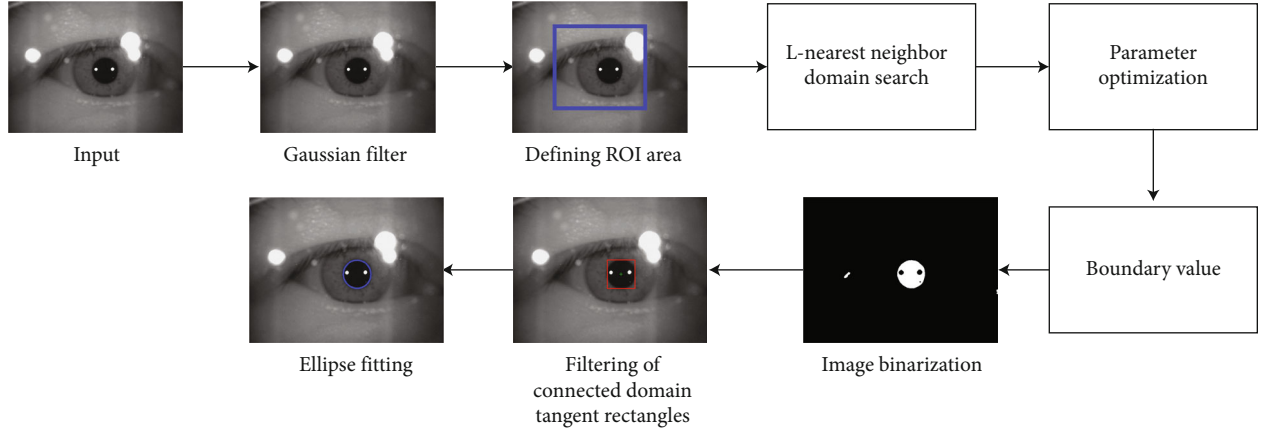


FIGURE 1: Basic idea of the robust pupil localization via a novel parameter optimization strategy.

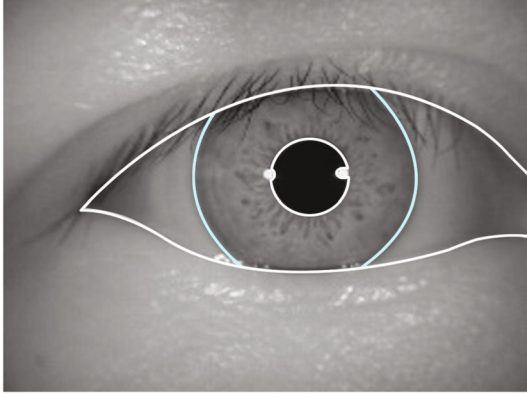


FIGURE 2: Pupillary boundary diagram.

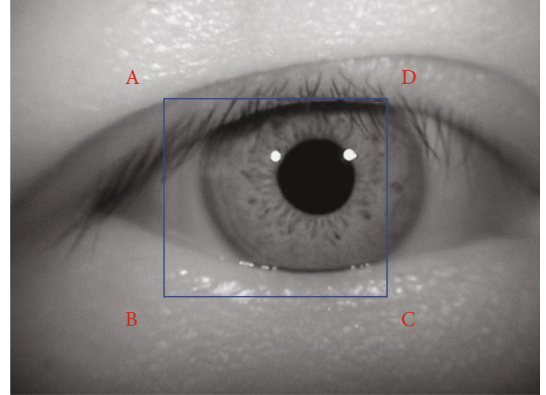


FIGURE 3: ROI region diagram.

**Input:** Length and width of the eye image;  
**Output:** Four coordinate points for the ROI region;  
 1.  $\text{RangeH} \leftarrow H/4$   
 2.  $\text{RangeW} \leftarrow W/4$   
 3.  $A \leftarrow (X_{\text{center}} - \text{RangeW}/2, Y_{\text{center}} - \text{RangeH}/2)$   
 4.  $B \leftarrow (X_{\text{center}} - \text{RangeW}/2, Y_{\text{center}} + \text{RangeH}/2)$   
 5.  $C \leftarrow (X_{\text{center}} + \text{RangeW}/2, Y_{\text{center}} + \text{RangeH}/2)$   
 6.  $D \leftarrow (X_{\text{center}} + \text{RangeW}/2, Y_{\text{center}} - \text{RangeH}/2)$

ALGORITHM 1: Define the ROI.

separated from the external area, which makes it hard for the pupil and eyelash, affecting the follow-up positioning accuracy of the pupils. The near- $L$  neighborhood search can accurately determine the gray parameters of the inner edge of the pupil to separate the pupil from the image background. This section will introduce the near- $L$  neighborhood search method in detail, as shown in Algorithm 2. The steps are as follows:

- (1) Take the minimum coordinate point of the gray value in the ROI area as the initial search coordinate,

set the search step, and start the search with the initial search coordinate, as shown in Figure 4.

In the figure,  $P_0$  is the initial search coordinate,  $L$  is the search step size, and the default value is 20.

- (2) In the rough positioning of the pupil edge, there are some sharp points in the image. When the search values in the eight directions of this point are equal, the rated step size will be automatically increased, and the calculation formula is as follows:

$$L_{\text{extra}} \begin{cases} 0k == 0 \\ L_{\text{extra}} + 1P_1 == P_2 \end{cases}, \quad (3)$$

where  $L_{\text{extra}}$  is the extra step size,  $K$  is the number of exceptions that occurred, and  $P_1$  and  $P_2$  are the search values returned.

- (3) In the search process, when all search directions jump simultaneously, the minimum value among



```

Input: 1. The initial search coordinates and the length and width of the image;
          2. Length and width of the eye image;
Output: The coordinate point and pixel value at which the
          jump occurred;
           $L \leftarrow 20$ .
           $L_{extra} \leftarrow 0$ 
          flag  $\leftarrow$  TRUE
          while flag do
            for  $i = 1$  to  $n$  do
              while  $P_{m_{later}} / P_{m_{last}} < T_{L_{nearest}}$  and  $P_i < H$  and  $P_i < W$ 
                Wdo
                   $P_{m_{later}} \leftarrow P_{m_{last}}$ 
                   $P_{m_{last}} \leftarrow P_{m_{last}} + L$ 
                end while
              if  $P_1$  to  $P_8$  is equal then
                 $L_{extra} \leftarrow L_{extra} + 1$ 
              else  $\{P_1$  to  $P_8$  is not equal $\}$ 
                flag  $\leftarrow$  FALSE
              end if
            end for
          end while

```

ALGORITHM 2: Search for gray parameters of inner pupil edge.

them is taken as the new starting point, and its calculation formula is as follows:

$$(X_{new\_point}, Y_{new\_point}) = \text{Point}[\min(P_{LT}, P_T, P_{RT}, P_R, P_{RD}, P_D, P_{LD}, P_L)], \quad (4)$$

where  $(X_{new\_point}, Y_{new\_point})$  is the new starting coordinate.  $P_{LT}, P_T, P_{RT}, P_R, P_{RD}, P_D, P_{LD}, P_L$ , and  $P_R$  are the dimensionless pixel values to the right of the starting search point.  $P_{LD}$  is the lower-left pixel value of the starting search point (dimensionless).  $P_D$  and  $P_{RD}$  are the gray values at the lower right (dimensionless). The Point is the generating function of the horizontal and vertical coordinates;

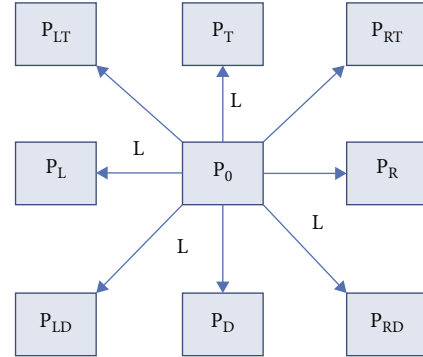
(4) Set the critical condition of the gray value. The formula is as follows:

$$P_m = \text{Max}(P_{m_{last}}, P_{m_{later}}), \frac{P_{m_{later}}}{P_{m_{last}}} > T_{L_{nearest}}. \quad (5)$$

After all the  $P_m$  were obtained, the minimum search value was taken as the internal pupil edge separation parameter, and the formula is as follows:

$$P_{L_{nearest}} = \min(P_m). \quad (6)$$

where  $P_m$  is the returned search value,  $P_{m_{later}}$  is the gray value of the next step,  $P_{m_{last}}$  is the current gray value of the search,  $T_{L_{nearest}}$  is the critical condition and the default value is 1.5, and  $P_{L_{nearest}}$  is the minimum binarization parameter

FIGURE 4: Diagram of near- $L$  neighborhood search.

value of pupil separation parameter and it is dimensionless. min is the minimum value function; max is the maximum value function.

**3.4. Optimization of Binarization Parameters.** The  $L$ -nearest neighbor domain search algorithm obtained the pupil edge's approximate inner boundary gray value. However, due to the high exposure or multiple light source points in the acquisition process of some eye images, there may be reflected light spots inside the pupil as the gray value of the pupil differs significantly from that of the reflected light spot. As a result, the binarization parameters obtained by near- $L$  neighborhood search do not contain the complete pupil range. Therefore, it is necessary to optimize the search of binarization parameters based on the  $L$ -nearest neighbor domain search to obtain the pupil outer edge parameters. The basic idea is to realize the extreme marginalization of binary parameters by reducing the search step size and



changing the judging boundary conditions. The idea is shown in the algorithm, and the main steps are as follows:

- (1) Through the last section of the nearly  $L$  neighborhood search method, we get the initial coordinate as starting points in eight directions of gray value jumps of the coordinates of the point, because the eight coordinates of the gray value of the step before leaping the gray value, so they represent the pupil in the edge location of the binary parameter values; as a parameter for image binarization, part of the image of the pupil will likely be lost. In order to ensure the integrity of the pupil, we reset these eight coordinates as the initial coordinate points
- (2) In the  $L$ -nearest neighbor domain search method, based on the initial coordinates of the starting point, eight of them point the direction to search, and in the process of binary parameter optimization, because the pupil within the gray value range has been confirmed, the search algorithm is no longer needed for eight directions at the same time; it is only needed according to the coordinate point in the nearly  $L$  neighborhood search process which is the returned direction search. Its basic formula is

$$P_m = \max(P_{m_{\text{last}}}, P_{m_{\text{later}}}), |P_{m_{\text{later}}} - P_{m_{\text{last}}}| > T_{L_{\text{optimization}}}, \quad (7)$$

where  $P_m$  is the gray value after the binarization parameters are refined on the basis of the inner pupil edge parameters,  $P_{m_{\text{later}}}$  is the gray value corresponding to the current coordinate in the binarization optimization search algorithm,  $P_{m_{\text{last}}}$  is the gray value corresponding to the last coordinate, and  $T_{L_{\text{optimization}}}$  is the new jump criterion with an initial value of 2. In the process of optimizing the search by the above binary parameters, the initial value of the search step is set as 1 in order to prevent the gray value of the pupil and iris in the eye image from increasing in sequence due to the dark light, thus affecting the judgment of the jump limit.

- (3) A relatively delicate pupil edge parameter is obtained after the binarization parameter optimization of the gray value of the inner pupil edge obtained by the near- $L$  neighborhood search method. The minimum value of the returned optimization parameters in eight directions is taken as the final value of binariza-

tion parameter optimization, and the formula is as follows:

$$P_{L_{\text{optimization}}} = \min(P_m), \quad (8)$$

where  $P_m$  is the binarization parameters of the eight directions after the parameters are defined by the method of binarization parameter optimization based on near- $L$  neighborhood search.  $P_{L_{\text{optimization}}}$  is the gray value of the outer pupil edge selected after binarization parameter optimization.

**3.5. Improved Linear Interpolation Method.** The inner margin and outer margin of the pupil and the method of acquisition obtained a rough location of the inner and outer pupil edges. However, when there is no obvious jump boundary between the gray values of the pupil and iris, the rough parameters of the outer edge of the pupil obtained by binarization parameter optimization will be pretty inaccurate. In this case, binarization of the image will lead to the integration of the pupil with the external background, which will make it challenging to locate the contour-point set of the connected domain of the pupil in the follow-up, thus making it unable to achieve accurate positioning of the pupil, as shown in Figure 5.

In order to eliminate the influence of no apparent difference between the outside edge of the pupil and the gray background value and solve the problem of the pupil and the outside background being mixed, we put forward the boundary value method to reduce the error caused by binarization parameter optimization. The main idea is linear interpolation. By reducing the weight of the parameters on the outer edge of the pupil and increasing the weight of the parameters on the inner edge of the pupil, the grayscale difference between the inner and outer edges can be well reduced to obtain the actual grayscale parameters of the pupil, as shown in Algorithm 4. Its calculation steps are as follows:

- (1) Set the upper limit range of binarization parameters and determine whether to use weight parameters to reduce parameter errors by judging whether the gray value parameters of the outer edge of the pupil optimized by binarization parameters are in the upper limit range
- (2) The binarization parameters of the image are determined by the set constraints

$$P_{\text{pupil}} = \begin{cases} \max(P_{L_{\text{optimization}}}, P_{L_{\text{nearest}}}), P_{L_{\text{optimization}}} > M_{\text{bound}} \wedge P_{L_{\text{nearest}}} > M_{\text{bound}}, \\ \frac{P_{L_{\text{optimization}}}}{P_{L_{\text{optimization}}} + P_{L_{\text{nearest}}}} \times P_{L_{\text{nearest}}} + \frac{P_{L_{\text{nearest}}}}{P_{L_{\text{optimization}}} + P_{L_{\text{nearest}}}} \times P_{L_{\text{optimization}}}, P_{L_{\text{optimization}}} < M_{\text{bound}} \vee P_{L_{\text{nearest}}} < M_{\text{bound}}, \end{cases} \quad (9)$$

```

Input: 1. The coordinate point and pixel value of the jump in Algorithm 2;
         2. Length and width of the eye image;
Output: Gray parameters of the outer edge of pupil;
 $L \leftarrow 1$ 
 $L_{\text{extra}} \leftarrow 0$ 
flag  $\leftarrow$  TRUE
while flag do
  for  $i = 1$  to  $n$  do
    while  $|P_{m_{\text{later}}} - P_{m_{\text{last}}}| < T_{L_{\text{optimization}}}$  and
       $P_i < H$  and  $P_i < W$  do
         $P_{m_{\text{later}}} \leftarrow P_{m_{\text{last}}}$ 
         $P_{m_{\text{last}}} \leftarrow P_{m_{\text{last}}} + L$ 
      end while
    if  $P_1$  to  $P_8$  is equal then
       $L_{\text{extra}} \leftarrow L_{\text{extra}} + 1$ 
    else  $\{P_1$  to  $P_8$  is not equal $\}$ 
      flag  $\leftarrow$  FALSE
    end if
  end for
end while

```

ALGORITHM 3: Optimization of binarization parameters.

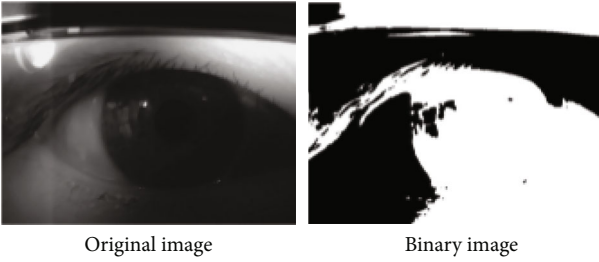


FIGURE 5: Image binarization rendering without boundary value.

where  $P_{\text{pupil}}$  is the parameter for binarization of the eye image determined by the three-step threshold method,  $P_{L_{\text{nearest}}}$  is the gray value parameter of the inner edge of pupil,  $P_{L_{\text{optimization}}}$  is the gray value parameter of the outer edge of pupil, and  $M_{\text{bound}}$  is a constraint on the maximum number of arguments in the bound value. The initial value is 80.

**3.6. Connected Domain Filtering.** After binarization of the image with the parameters obtained by the three-step threshold method, no matter the original multilight source points in the image, or the occlusions such as eyelashes and eye sockets, there are apparent segmentation boundaries with the pupil, as shown in Figure 6.

In order to accurately identify the pupil-connected domain, we first introduce the area proportion constraint of the outer tangent rectangle of the pupil-connected domain, whose calculation formula is as follows:

$$S_i = (X_{ri} - X_{li}) \times (Y_{di} - Y_{ti}), \quad (10)$$

where  $S_i$  is the area of the connected domain tangent rectangle,  $X_{ri}$  is the rightmost abscissa value of the tangent rectangle,  $X_{li}$  is the leftmost abscissa value of the tangent rectangle,  $Y_{di}$  is the bottommost ordinate value of the tangent

rectangle, and  $Y_{ti}$  is the uppermost ordinate value of the tangent rectangle. Secondly, the average Euclidean distance constraint between the contour point set and the center coordinate is introduced, and its calculation formula is

$$D_i = \frac{1}{N} \sum_1^N \sqrt{(x_i - X_{\text{center}})^2 + (y_i - y_{\text{center}})^2}, \quad (11)$$

where  $D_i$  is the average Euclidean distance of the  $i$ th contour point set,  $N$  is the number of pixels,  $(x_i, y_i)$  is the pixel coordinates of the contour points, and  $(X_{\text{center}}, Y_{\text{center}})$  is the center coordinates of the eye image. Finally, the empirical value of the number range of pupil contour points is added to calculate the pupil-connected domain satisfying the conditions, and the formula is as follows:

$$C_{\text{pupil}} = \max((D_i < M_D) \cup (S_i < M_S) \cup (100 < C_i < 300)), \quad (12)$$

where  $C_{\text{pupil}}$  is the contour point set of the pupil and  $C_i$  is the number of pixels in the contour point set.

**3.7. Least Square Ellipse Fitting.** In morphology, the morphological characteristics of the pupil are similar to those of the ellipse, and some pupils may be close to the circle. Therefore, we adopt the ellipse-fitting method of the least square method to fit the pupil and achieve the positioning of the pupil. The general formula is as follows:

$$x^2 + Axy + By^2 + Cx + Dy + E = 0. \quad (13)$$

According to the general equation of an ellipse, solving the equation requires at least five measuring points on the ellipse contour. By randomly selecting five-coordinate points from the contour point set in the pupil-connected domain

```

Input:  $L$ -nearest neighbor domain search parameter and
          binarization parameter optimization result;
Output: The final binarization parameter;
 $M_{\text{bound}} \leftarrow 80$ 
if  $P_{L_{\text{optimization}}} > M_{\text{bound}}$  and  $P_{L_{\text{nearest}}} > M_{\text{bound}}$ 
then
 $P_{\text{pupil}} \leftarrow \max(P_{L_{\text{optimization}}}, P_{L_{\text{nearest}}})$ 
else  $\{P_{L_{\text{optimization}}} < M_{\text{bound}}$  or  $P_{L_{\text{nearest}}} < M_{\text{bound}}\}$ 
 $P_{\text{pupil}} \leftarrow (P_{L_{\text{optimization}}} / (P_{L_{\text{optimization}}} + P_{L_{\text{nearest}}})) \times P_{L_{\text{nearest}}} + (P_{L_{\text{nearest}}} / (P_{L_{\text{optimization}}} + P_{L_{\text{nearest}}})) \times P_{L_{\text{optimization}}}$ 
end if

```

ALGORITHM 4: Set boundary value.

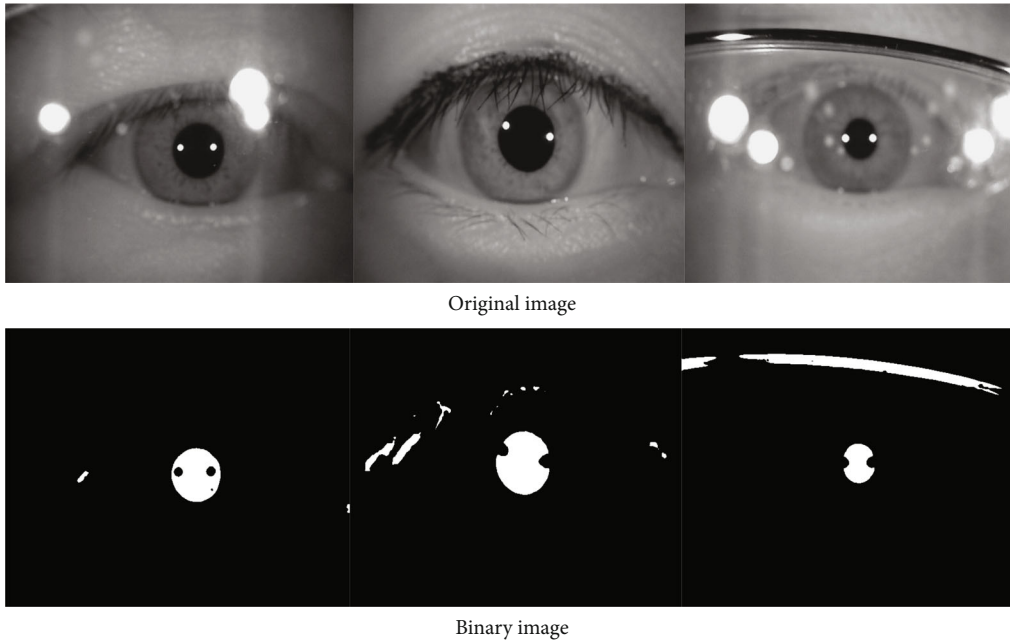


FIGURE 6: A binarization result graph with three-step threshold parameters is used.

and putting them into the equation for solving, the final ellipse-fitting equation can be obtained to achieve pupil location.

## 4. Experiments

### 4.1. Datasets and Metrics

**4.1.1. Datasets.** The CASIA-IrisV1 dataset is the earliest version of the dataset provided by the Chinese Institute of Automation [25]. The dataset contained 756 high-quality eye images, which were enhanced to distinguish the pupil from its background area. Meanwhile, in the process of data collection of the CASIA-IrisV1 version, the collected personnel are strictly required not to wear glasses and keep their eyes as wide as possible, so that the pupil area will not be blocked by eyelashes, hair, and other organs, which indirectly simplifies the pupil identification difficulty of the CASIA-IrisV1 dataset. In order to verify the high accuracy

of TST in high-quality eye images and prevent the algorithm from unilaterally adapting to pupil localization in complex images, but with the accuracy decreasing in high-quality images, this paper first adopts the CASIA-IrisV1 dataset for verification.

The CASIA-Irisv4 dataset is also a dataset provided by the Chinese Institute of Automation. The dataset contains iris images from more than 1800 real objects and 1000 virtual objects. All iris images are 8-bit gray files collected or synthesized under near-infrared illumination. At the same time, the CASIA-IrisV4 dataset has a diversity of eye images. For example, in Interval, a circular NIR LED array was designed to allow the iris camera to capture very clear iris images. In Lamp, elastic deformation of the iris texture was induced by light reaction, and the pupil expanded and contracted correspondingly under different light conditions. In addition, the CASIA-IrisV4 dataset also includes glasses-wearing, composite images, and iris images with specular reflection due to glasses-wearing. Therefore, this

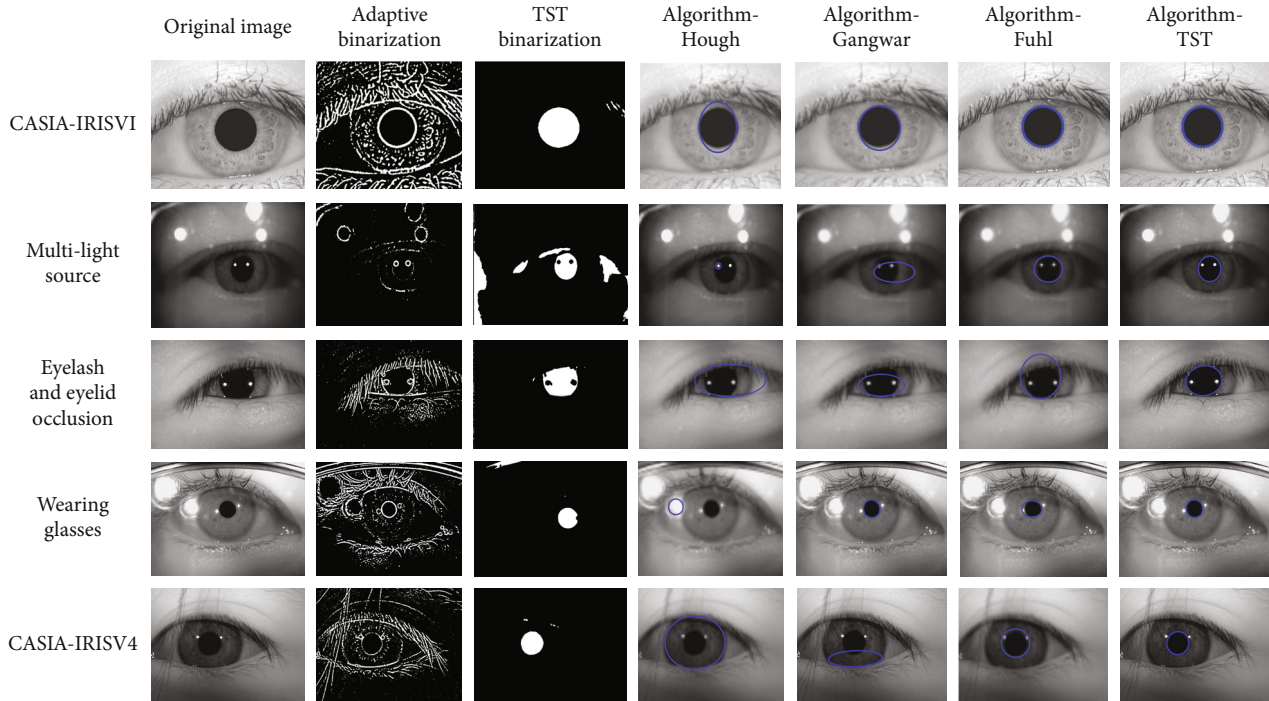


FIGURE 7: Test results for the datasets.

TABLE 1: The experimental results.

	CASIA-IrisV1	Accuracy	CASIA Iris Subject Ageing	Accuracy	CASIA-IrisV4	Accuracy
Gangwar [26]	743	98.3%	1893	93.1%	2467	77.5%
Hough circle [15]	731	96.7%	1876	92.3%	2245	70.5%
Fuhl [16]	751	99.3%	1921	94.5%	2657	83.4%
TST-adaptive	739	97.8%	1647	81.1%	1931	60.7%
TST	754	99.7%	1957	96.3%	2765	86.8%

dataset can well verify the accuracy and robustness of the algorithm in complex iris image localization. In order to verify the validity of TST, the CASIA-IrisV4 dataset was also used.

The CASIA-Iris-Aging dataset is a dataset provided by the Chinese Institute of Automation. In the CASIA-IrisV1 and CASIA-IrisV4 datasets, eye images with multiple light sources or high exposure rarely exist. In order to verify the practical effect of TST in such eye images, this dataset is introduced in this paper. In this dataset, images with pupils obscured by eyelashes or hair and images with multiple light points or high exposure are included.

**4.1.2. Metrics.** In order to obtain the specific pixel-level error, the improved mean error is used as the evaluation index of pupil positioning accuracy. Its calculation formula is as follows:

$$\text{err} = \frac{1}{4} \sum_0^3 |x_b - X_b|, \quad (14)$$

where  $x_b$  is the artificially marked boundary value of pupil outer tangent rectangle;  $X_b$  is the pupil tangent rectangular

boundary obtained by TST; (0, 1, 2, 3) represent the upper, lower, left and right boundaries, respectively; and  $\text{err}$  is the error value.

**4.2. Experimental Results.** In order to verify that TST has a better pupil localization effect, this paper made comparisons with the classical Hough [15] pupil localization method, Gangwar [19] pupil localization method, Fuhl [20] pupil localization method, and traditional threshold method. It can be seen from the results that TST has high pupil localization accuracy in high-quality datasets and maintains good robustness in complex eye images. However, Gangwar and Fuhl failed to separate the pupil from the image background, resulting in a decrease in the pupil localization accuracy of their algorithms in complex images, as shown in Figure 7.

As shown in Table 1, TST maintains high accuracy in both the high-quality CASIA-IrisV1 dataset and the complex image dataset of CASIA Iris Subject Ageing. Even in the specially processed CASIA-IrisV4 dataset, 86.8% accuracy was achieved.

The above results show that TST can better separate the pupil from the background area by fitting the gray

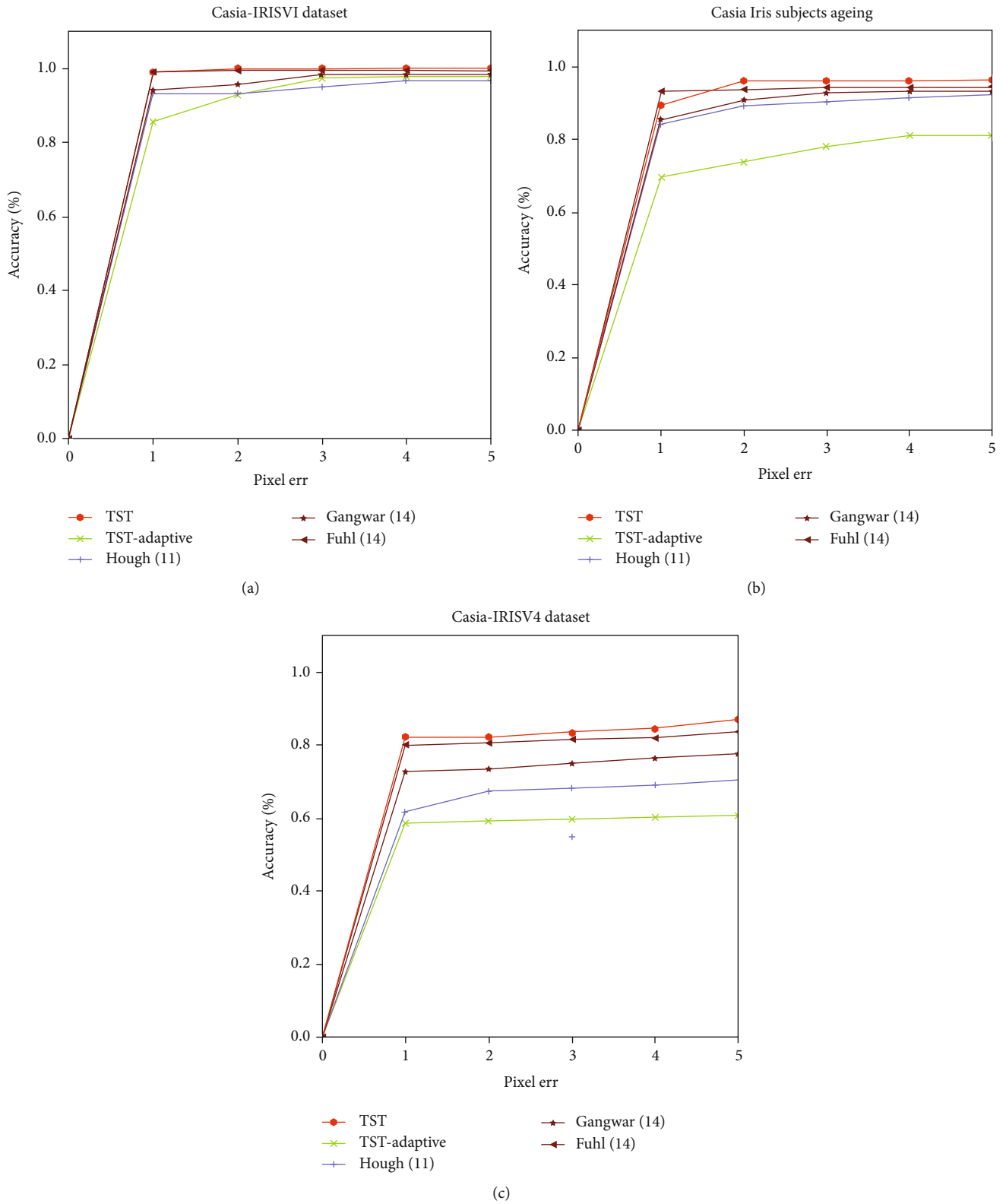


FIGURE 8: Accuracy results for the datasets.



parameters of the pupil edge, which simplifies the subsequent screening of the pupil-connected domain and improves the accuracy and robustness of pupil location.

## 5. Discussion

TST showed good pupil positioning accuracy and robustness in the above dataset images for both the high-quality CASIA-IrisV1 dataset and the challenging CASIA-IrisV4 dataset. The reason is that TST is different from other traditional pupil localization methods, which focus more on the selection of image binarization parameters. Through the three-step threshold method included in TST, the gray parameters of the inner edge of the pupil are obtained to eliminate the interference of most background targets. Then, by expanding the parameters of the inner edge of the pupil again, the integrity of the target pupil is guaranteed, and the information of the pupil edge will not be lost due to the selection of too small binarization parameters. Finally, through improved linear interpolation method, the pupil parameters and outer edge-to-edge values make the final binarization parameters suitable for the separation of the pupil to ensure the integrity of the pupil's case, as much as possible from the interference of background factors, and simplify the pupil-connected domain-filtering conditions. However, TST-adaptive adopts the adaptive threshold method to binarize the image, which retains most interference factors in the image and cannot achieve complete pupil separation, thus increasing the difficulty of pupil location and reducing the accuracy of pupil location.

In the pupil-connected domain in the process of screening, based on the idea of IOU losses of the convolutional neural network computation, we proposed the pupil of the connected domain circumscribed rectangular area ratio, aspect ratio, and contour point sets and the average Euclidean distance of the image center joining the pupil filter-connected domain, combining the pupil contour point number. Thus, the difficulty of filtering the pupil-connected domain is reduced. The accuracy and robustness of pupil location are increased. In Figure 8, TST in the 1-pixel error range reached the peak pupil location accuracy. The reason is that in the binary parameter selection problem, we adopt the three steps of the threshold value method to determine the threshold parameter, which hugely fits the pupil edge information. Therefore, in the process of pupil orientation, the positioning error is smaller.

## 6. Conclusion and Future

This paper proposed a method of pupil localization via a novel parameter optimization, called TST. TST includes the three-step threshold method. Compared with other pupil location algorithms, the three-step threshold method pays more attention to the early image processing, thus simplifying the difficulty of subsequent pupil location. In addition, in order to solve the problem that the pupil localization accuracy decreases when the pupil is close to the background gray value, this paper designed a unique filtering algorithm of the pupil-connected domain. The experimental results

show that TST has a better pupil localization effect and is suitable for pupil localization in complex images.

In the future, TST may help to improve the overall performance of eye movement tracking due to its good performance. We will improve the reliability of eye movement tracking by combining the reflected light spot of the pupil with the pupil center coordinate of TST positioning.

## Data Availability

To test our method, we used the CASIA-IrisV1 dataset, the CASIA-IrisV4 dataset, and 2032 eye images from the CASIA Iris Subject Ageing dataset collected by the Institute of Automation, Chinese Academy of Sciences, and the CASIA iris image database, in <http://biometrics.idealtest.org/activeuser.do?id=31855>, Chinese Academy of Sciences Institute of Automation.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] Y. Du, R. W. Ives, and D. M. Etter, "Iris recognition," in *Circuits, Signals, and Speech and Image Processing*, pp. 25-1-25-10, CRC Press, 2018.
- [2] R. P. Wildes, "Iris recognition: an emerging biometric technology," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1348-1363, 1997.
- [3] N. Ahmadi, M. Nilashi, S. Samad, T. A. Rashid, and H. Ahmadi, "An intelligent method for iris recognition using supervised machine learning techniques," *Optics & Laser Technology*, vol. 120, p. 105701, 2019.
- [4] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1148-1161, 1993.
- [5] N. Min-Allah, F. Jan, and S. Alrashed, "Pupil detection schemes in human eye: a review," *Multimedia Systems*, vol. 27, no. 4, pp. 753-777, 2021.
- [6] J. Liao, Y. Wang, D. Zhu, Y. Zou, S. Zhang, and H. Zhou, "Automatic segmentation of crop/background based on luminance partition correction and adaptive threshold," *IEEE Access*, vol. 8, pp. 202611-202622, 2020.
- [7] Y. Wang, X. Lu, and W. Zhou, "Global adaptive optimization parameters for robust pupil location," in *2021 17th International Conference on Computational Intelligence and Security (CIS)*, pp. 285-289, Chengdu, China, 2021.
- [8] H. Zhang, H. Zhou, W. Jiao et al., "Biological features de-identification in iris images," in *2018 15th international symposium on pervasive systems, algorithms and networks (I-SPAN)*, pp. 67-71, Yichang, China, 2018.
- [9] N. N. Minakova and I. V. Petrov, "Modification of Daugman's integrodifferential operator using Bresenham's algorithm for iris localization," in *2018 XIV international scientific-technical conference on actual problems of electronics instrument engineering (APEIE)*, pp. 183-187, Novosibirsk, Russia, 2018.
- [10] X. Chen, J. Wang, Y. Ruan, and S. Z. Gao, "An improved iris recognition method based on discrete cosine transform and Gabor wavelet transform algorithm," *Engineering Letters*, vol. 27, no. 4, 2019.

- [11] M. T. Setiawan, S. Wibirama, and N. A. Setiawan, "Robust pupil localization algorithm based on circular Hough transform for extreme pupil occlusion," in *4th International Conference on Science and Technology (ICST)*, pp. 1–5, Yogyakarta, Indonesia, 2018.
- [12] L. Shang, C. Zhang, and H. Wu, "Eye focus detection based on OpenCV," in *2019 6th international conference on systems and informatics (ICSAI)*, pp. 855–858, Shanghai, China, 2019.
- [13] D. A. S. Dewi, S. Wibirama, and I. Ardiyanto, "Robust pupil localization algorithm under off-axial pupil occlusion," in *2019 2nd international conference on bioinformatics, biotechnology and biomedical engineering (BioMIC')-bioinformatics and biomedical engineering*, pp. 1–6, Yogyakarta, Indonesia, 2019.
- [14] D. Li, F. Hu, L. Wang, and M. Zhang, "Iris center localization using integral projection and gradients," in *2014 International Conference on Audio, Language and Image Processing*, pp. 211–215, Shanghai, China, 2014.
- [15] F. Jan, I. Usman, S. A. Khan, and S. A. Malik, "Iris localization based on the Hough transform, a radial-gradient operator, and the gray-level intensity," *Optik*, vol. 124, no. 23, pp. 5976–5985, 2013.
- [16] W. Fuhl, T. Kübler, K. Sippel, W. Rosenstiel, and E. Kasneci, "Excuse: robust pupil detection in real-world scenarios," in *International Conference on Computer Analysis of Images and Patterns*, pp. 39–51, Nice, France, 2015.
- [17] R. Fusek and P. Dobeš, "Pupil localization using self-organizing migrating algorithm," in *International Conference on Advanced Engineering Theory and Applications*, pp. 207–216, Ostrava, Czech Republic, 2020.
- [18] S. Jamaludin, N. Zainal, and W. M. D. W. Zaki, "Deblurring of noisy iris images in iris recognition," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 1, pp. 156–159, 2021.
- [19] R. Fusek, "Pupil localization using geodesic distance," in *International Symposium on Visual Computing*, pp. 433–444, Springer, Cham, Switzerland, 2018.
- [20] E. Jalilian and A. Uhl, "Iris segmentation using fully convolutional encoder decoder networks," in *Deep Learning for Biometrics*, pp. 133–155, Springer, Cham, Switzerland, 2017.
- [21] Y. Yang, P. Shen, and C. Chen, "A robust iris segmentation using fully convolutional network with dilated convolutions," in *2018 IEEE International Symposium on Multimedia (ISM)*, pp. 9–16, Taichung, Taiwan, 2018.
- [22] J. H. Choi, K. I. Lee, and B. C. Song, "Eye pupil localization algorithm using convolutional neural networks," *Multimedia Tools and Applications*, vol. 79, no. 43–44, pp. 32563–32574, 2020.
- [23] J. H. Choi, K. I. Lee, Y. C. Kim, and B. C. Song, "Accurate eye pupil localization using heterogeneous CNN models," in *2019 IEEE international conference on image processing (ICIP)*, pp. 2179–2183, Taipei, Taiwan, 2019.
- [24] M. B. Lee, Y. H. Kim, and K. R. Park, "Conditional generative adversarial network-based data augmentation for enhancement of iris recognition accuracy," *IEEE Access*, vol. 7, pp. 122134–122152, 2019.
- [25] "China Research Institute of Automation iris image database-Chinese Academy of Sciences Institute of Automation <http://biometrics.idealtest.org/activeuser.do?id=31855>.
- [26] A. Gangwar, A. Joshi, A. Singh, F. Alonso-Fernandez, and J. Bigun, "IrisSeg: a fast and robust iris segmentation framework for non-ideal iris images," in *2016 international conference on biometrics (ICB)*, pp. 1–8, Halmstad, Sweden, 2016.

## Research Article

# Prediction of Air Leakage Rate of Sintering Furnace Based on BP Neural Network Optimized by PSO

Xiaokai Quan,<sup>1</sup> Nannan Zhang,<sup>2</sup> Guo Yu ,<sup>3</sup> Qunfeng Liu,<sup>4</sup> and Lianbo Ma<sup>1</sup>

<sup>1</sup>Department of Software Engineering, Northeastern University, Shenyang 110000, China

<sup>2</sup>Department of Physical Education, Northeastern University, Shenyang 110000, China

<sup>3</sup>Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, East China University of Science and Technology, Shanghai 200000, China

<sup>4</sup>School of Computer Science and Technology, Dongguan University of Technology, Guangdong 523000, China

Correspondence should be addressed to Guo Yu; [guoyu@ecust.edu.cn](mailto:guoyu@ecust.edu.cn)

Received 14 March 2022; Revised 1 April 2022; Accepted 13 April 2022; Published 29 April 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Xiaokai Quan 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.

Aiming at the difficulty of air leakage detection in the sintering process of the sintering furnace, especially the problems of high detection cost and poor timeliness of detection results when traditional methods are used for detection, we propose an air leakage rate prediction algorithm. Firstly, we use the particle swarm optimization algorithm to optimize the initial parameters of the neural network based on back propagation and get the best set of initial parameters through continuous search. Secondly, the optimized parameters are substituted into the neural network to train them with training data, and the trained parameters are obtained. Finally, the air leakage rate of the test set data is predicted by using the trained parameters. Compared with traditional calculation methods such as gas analysis and calorimetry, the proposed method can greatly simplify the detection process, shorten the detection time, and control the error within 5%, allowing the user to deal with the air leakage problem more timely and improve the overall sintering quality.

## 1. Introduction

With the development of the iron and steel industries, the demand for iron ore is increasing day by day. However, there is less and less rich ore that can be smelted directly into the furnace, so a large number of poor ore resources must be mined and used [1–4]. Direct smelting of lean ore into the furnace will worsen the production index of the blast furnace and reduce the economic benefit. Therefore, lean ore needs to be treated by beneficiation to select concentrate with high iron content, and fine ore produced in the mining and processing of concentrate and rich ore can only be used for blast furnace ironmaking after being lumped.

Iron ore powder agglomeration methods currently mainly include sintering and pelletizing. The production of pellets requires fine-grained concentrates and requires gas or liquid fuel for roasting. It has strong adaptability and can not only produce sintered ore with coarse-grained rich ore powder and concentrate, but also process industrial

iron-containing wastes. Due to the dominance of the production of iron ore powder agglomeration, sintered ore accounts for 70% to 90% of the total iron-containing raw materials entering the furnace [5–8]. Therefore, sinter has always been the main raw material for blast furnace ironmaking. It mainly determines the technical and economic indicators of blast furnace smelting production [9–13]. In blast furnace ironmaking, the use ratio of sinter basically accounts for about 75% of the blast furnace burden and more than 70% of the blast furnace ironmaking cost and energy consumption. Therefore, the technical and economic indicators and quality of sintering production play a decisive role in the cost and effect of the blast furnace.

It can be seen that in the modern steel production process, sintering plays an irreplaceable role as an important link in providing raw materials for ironmaking blast furnaces in modern iron and steel production [14–20]. Using the sintering method to produce sinter not only solves the problem of fine ore ironmaking but also improves the

metallurgical properties of iron-containing raw materials, so that the production index and economic benefits of the blast furnace are obviously improved [21–25]. At present, the most commonly used sintering equipment in sintering production at home and abroad is the belt sintering machine, which provides most of the high-quality man-made rich ore required for the production of ironmaking blast furnaces. From the perspective of energy saving and environmental protection, the biggest disadvantage of the sintering machine is that its air leakage rate is high, resulting in diffuse noise in the vast space around the equipment.

During the sintering process, the negative pressure of the system will inevitably lead to a certain degree of air leakage between the material surface gap and the sidewall of the trolley, so that the air enters the sintering system from the position with poor sealing performance of the equipment, and at the same time reduces the working negative pressure of the sintering system [26–30].

The main causes of air leakage in sintering machines are (1) the gaps between trolleys and trolleys, between baffles and baffles, and between trolley grate bars and baffle pins; (2) the air leakage of the air duct and compensator of the small bellows is due to the rapid wear due to high temperature, air pumping, material scouring, and other reasons; (3) the sintering machine bellows head and tail. Under the action of high negative pressure in the large flue, a large amount of air is pumped into the head and tail spring sealing cover plates and the slideways on both sides; (4) wear air leakage of each connecting flange, pipeline, and bellows of the main exhaust system. More details of the air leakage of the sintering machine are referred to in [31].

The effective air volume per unit area of the sintering trolley reduces the output of the sintering machine and the quality of the sintered ore. The lower the unit effective air volume of the sintering system, the less sintered ore output there will be. In the sintering process, the electricity consumed by the fan accounts for more than 70% of the total electricity. If the air leakage rate is too high, the effective power of the fan will be greatly reduced. A large amount of air leakage not only affects the electricity consumption, but also affects the effective utilization of energy in the sintering process, and reduces the production efficiency, finally increasing the energy consumption in sintering production and the production cost of sinter. Finally, the consumption of energy in sintering production and the production cost of sinter have increased. The practice of some sintering plants has proved that the output can be increased by 6% when the air leakage rate is reduced by 10%, the power consumption can be reduced by 2 kW/h per ton of sinter, the coke powder can be reduced by 1 kg/t, and the finished product rate can be increased by 1.5%~2.0% [32–34]. Therefore, one of the most direct and effective ways to increase production, reduce consumption, and increase economic benefits in the sintering production process is to control the air leakage rate.

At present, the air leakage rate of various types of sintering equipment is generally about 50%, and the air leakage rate of some advanced enterprises is 40–45% during normal production. The difference in air leakage indicators

of different enterprises is affected by various factors, but the main reason is that real-time online monitoring of air leakage cannot be performed, and then it is impossible to accurately determine the air leakage part of the sintering machine and carry out maintenance, which has become a common problem in the sintering industry. At present, the measurement of the air leakage rate of the sintering equipment is a general manual operation, which has a long measurement period and consumes more manpower and material resources, and the measurement results are greatly affected by production operation and working conditions [35]. If online monitoring can be realized, it will be of great help in the analysis of production status, operation, scheduling decision-making, and equipment maintenance. Both the oxygen content method and the colorimetric method can realize online detection and have been used in some factories [36]. The oxygen content method involves analyzing the air leakage rate of the sintering equipment system by detecting the oxygen content of the flue gas in the sintering flue. The oxygen content is detected by a zirconia analyzer, which is installed between the electrostatic precipitator and the bellows [37]. The principle of the calorimetric method is that the heat reserve change of exhaust gas leaving the system plus the heat loss of the system is equal to the heat of exhaust gas entering the system. To establish a heat balance formula, measure the temperature through thermocouples and calculate the air leakage rate according to the formula. Using deep learning to calculate the air leakage rate only requires a few easy-to-measure data points such as ignition temperature, airflow, and discharge temperature to calculate a more accurate air leakage rate [38]. The biggest advantage of neural networks is that they can continuously learn and improve to adapt to the prediction of air leakage rate under various conditions. A more rapid and accurate calculation of air leakage rate enables faster processing and maintenance, thereby improving production efficiency.

## 2. Proposed PSO-BP Combinatorial Model

In this section, we will firstly introduce the backpropagation (BP) based neural network model. Then, the particle swarm optimization (PSO) algorithm used in this study is described. Finally, the proposed PSO-BP combinatorial model is presented.

*2.1. BP Neural Network Model.* The BP neural network [39] is widely used in industrial control, and the basic idea of its model is to make real-time adjustments to the network weights by analyzing the network output error and gradient information, to further bring the network output closer and closer to the expected value. The BP neural network prediction model established in this paper uses the temperature and negative pressure data of 15 sintering processes to predict the air leakage rate, including ignition temperature, sintering machine speed, material layer thickness, and waste temperature. A complete neural network consists of three parts: an input layer, a hidden layer, and an output layer [40]. Figure 1 shows the structure of

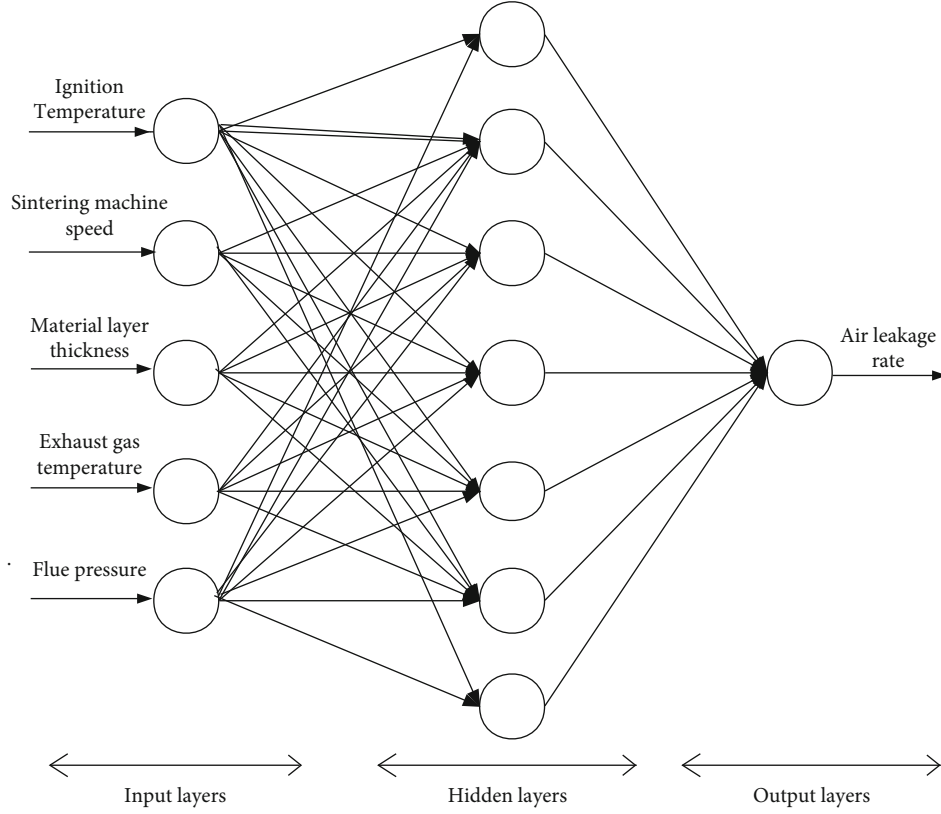


FIGURE 1: BP neural network structure.

the BP neural network used in this paper.  $x_1 \sim x_{15}$  in the figure represents the input layer of the neural network;  $w_1 \sim w_{15}$  represents the node weight of the input layer;  $y_1 \sim y_7$  represents the node weight of the hidden layer;  $o$  is the output layer. In the neural network constructed by the algorithm used in this paper, the input layer includes 15 nodes whose actual meanings are ignition temperature, sintering machine speed, material layer thickness, waste temperature, etc. The prediction result of the air leakage rate of sintering equipment is the output layer, the number of hidden layer nodes selected according to the empirical formula  $b = \sqrt{m+n} + a$ , where  $m$  is the number of input nodes,  $n$  is the number of output nodes,  $a$  is a constant, and here we set it to 3. Therefore, the final number of hidden layer nodes  $b$  is 7.

As shown in the three-layer BP network above, the output  $H_j$  of the hidden layer is given in Equation (1):

$$H_j = g\left(\sum_{i=1}^m w_{ij}x_i + a_j\right). \quad (1)$$

The output of the output layer is shown in Equation (2):

$$O_k = \sum_{j=1}^b H_j w_{jk} + b_k. \quad (2)$$

In this paper, the error formula is taken as Equation (3):

$$E = \frac{1}{2} \sum_{k=1}^n (Y_k - O_k)^2. \quad (3)$$

The update formula of the weight is given in Equation (4):

$$\begin{cases} w_{ij} = w_{ij} + \eta H_j (1 - H_j) x_i \sum_{k=1}^m w_{jk} e_k \\ w_{jk} = w_{jk} + \eta H_j e_k \end{cases}. \quad (4)$$

BP neural network training is roughly divided into two processes: forward propagation and back propagation. After the forward propagation is completed, the error between the target value and the actual value is calculated, and then the internal weights and thresholds are continuously updated and adjusted through the back propagation of the error until the error is less than the initial set accuracy value. The training is completed, and the parameters can be used to predict air leakage rates [41]. The BP neural network is a kind of single-hidden-layer feedforward neural network. The weight and threshold of the network are iterated continuously by the gradient descent method, and finally, the data regression and classification are completed. However, in the initial state, the weights and thresholds of the network are arbitrarily set, and the system batching and control parameters



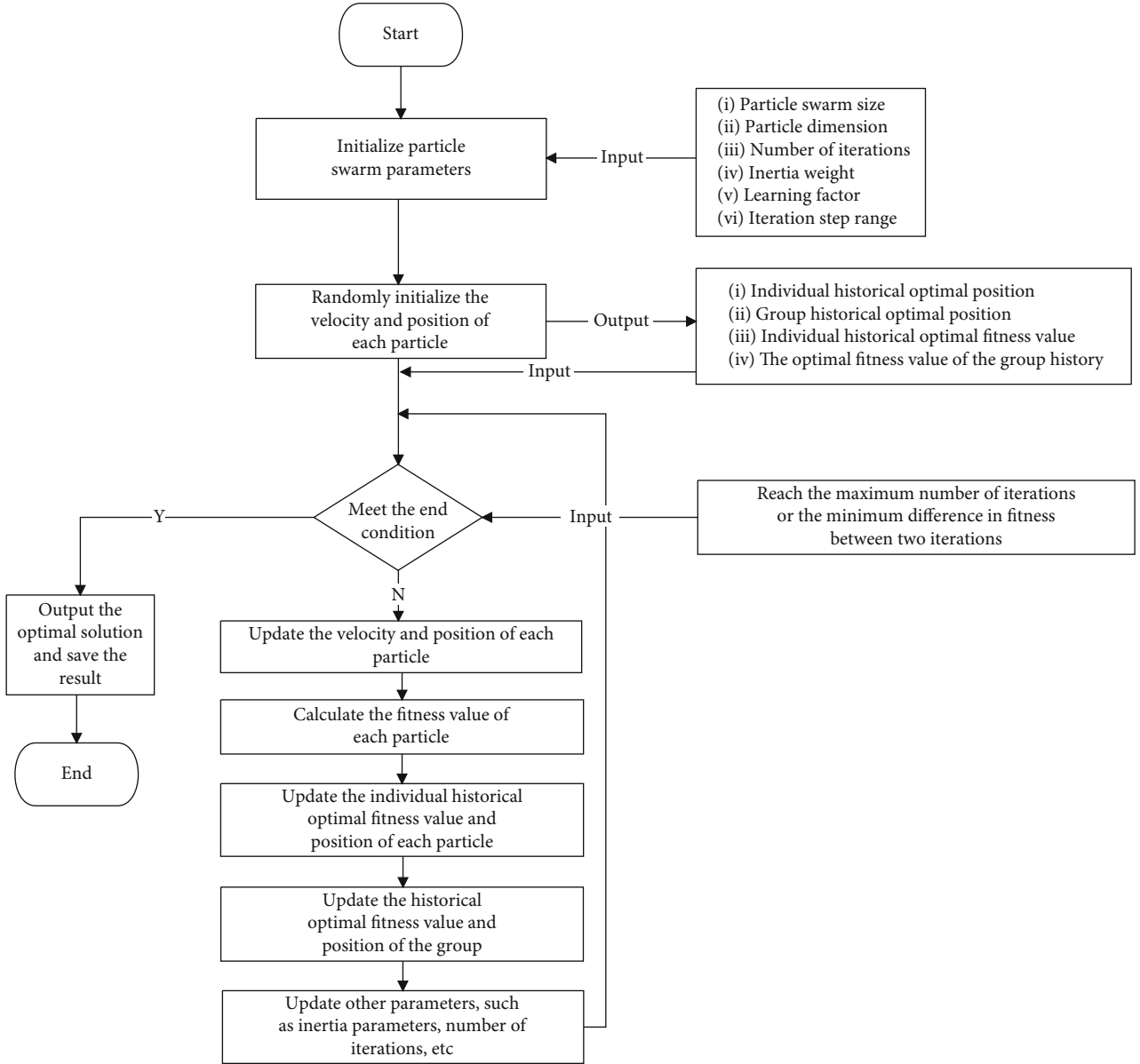


FIGURE 2: PSO model algorithm flow chart.

will be adjusted with the change of process requirements, which will lead to the difficulty of achieving optimal accuracy all the time. Therefore, this paper introduces the PSO algorithm to optimize the internal parameters to ensure that the prediction model is always in the optimal state.

Compared with other optimization algorithms, the PSO algorithm has the following advantages when optimizing neural networks: (1) for non-convex optimization problems such as neural network training, the stochastic optimization algorithm will have a stronger ability to explore the solution set space. (2) As a meta-heuristic algorithm, the implementation of PSO is relatively simple, and the calculation process is separated from the problem model. Many known modules of generic and meta-heuristic algorithms can be added to the PSO algorithm to improve its efficiency. (3) PSO is friendly to distributed computing and can effectively improve computing power. (4) Its speed updating mechanism, inertia,

and other factors can be used to optimize the parameters of benchmarking neural networks for continuous optimization problems.

**2.2. PSO Algorithm.** The PSO algorithm [42] is a kind of swarm intelligence optimization algorithm that simulates individual birds through particles, the flow chart of which is presented in Figure 2. Particles communicate and cooperate with each other, constantly feeding back on their position and speed and updating them to find the optimal target solution.

Assuming that there is a  $D$ -dimensional feasible space, the particle population number is  $N$ .

In the decision space, the current position of the  $i$ -th particle can be expressed as Equation (5):

$$x_i = (x_{i1}, x_{i2}, \dots, x_{iD}), \quad (5)$$

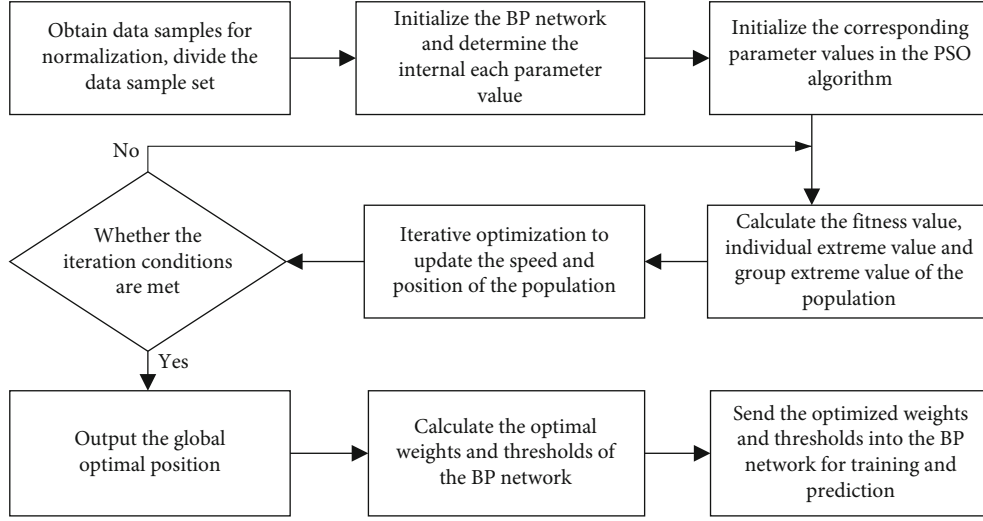


FIGURE 3: PSO-BP model algorithm flow chart.

where the particle  $x_i$  in this study represents one component ( $w_i$ ) of the initial parameter  $w$  of the neural network. Each  $x_{ij}$  of a particle represents the  $w_i$  after  $j$ -th perturbation, and  $D$  is the predefined time for perturbation.

The current velocity of the  $i$ -th particle is Equation (6):

$$v_i = (v_{i1}, v_{i2}, \dots, v_{iD}). \quad (6)$$

The best individual position of the  $i$ -th particle is Equation (7):

$$P_{best} = (P_{i1}, P_{i2}, \dots, P_{iD}). \quad (7)$$

The global optimal value in the whole search space is Equation (8):

$$g_{best} = (g_1, g_2, \dots, g_D), \quad (8)$$

$$v_{\text{BD}}^{k+1} = w_{\text{BD}}^k + c_1 r_1 (P_{iD}^k - x_{iD}^k) + c_2 r_2 (P_{gD}^k - x_{gD}^k), \quad (9)$$

$$x_{\text{BD}}^{k+1} = x_{\text{BD}}^k + v_{\text{BD}}^{k+1}. \quad (10)$$

At the same time, the velocity and position of the particles are updated according to the optimal value of each particle's feedback and the optimal value of the whole population [43]. The specific update formulas are shown in Equation (9) and Equation (10). In the formula:  $i$  represents the particle number;  $v_{\text{BD}}^{k+1}$  represents the speed of the particle at the next moment;  $w$  represents the inertia weight;  $v_{\text{BD}}^k$  represents the speed of the particle at the current moment;  $c_1$ ,  $c_2$  represent the learning factors;  $r_1$ ,  $r_2$  represent any numbers between  $[0, 1]$ ;  $P_{iD}^k$  represents the current optimal position of the particle;  $P_{gD}^k$  represents the global optimal position of the particle;  $x_{\text{BD}}^k$  represents the current position of the particle;  $x_{\text{BD}}^{k+1}$  represents the next moment position of the particle [44].

**2.3. Proposed PSO-BP Combinatorial Model.** When a single BP neural network model is trained, the parameters such as initial weight and threshold are random, and then they are gradually modified in the iterative process. The initial parameters may have some negative effects on the training speed and training results, so we use a particle swarm optimization algorithm to optimize the initial parameters to improve the neural network [45].

Firstly, we use a few particles to simulate our initial parameters, assign the initial values, and then send them to the constructed network for an iteration. The initial loss of this training can be obtained, recorded, and set as the individual optimal value of the particle. Then, the position of the particle is updated according to the formula of velocity and position defined in Equations (9) and (10). After that, the loss is calculated again, so that the historical minimum loss, which represents the historical optimal position of the particle, will be updated. Multiple particles are searched at the same time, and the position is constantly changed to find the optimal position until the iterative conditions are met and the search stops. The optimal position found by particle swarm optimization is the best initial parameter value of the network, and then it is substituted into the BP neural network for training, and the final result is obtained. The flow chart of the PSO-BP [46] model establishment is shown in Figure 3.

### 3. Experimental Results and Analysis

A total of 1121 groups of air leakage rate data were collected in the experiment, among which the first 1010 groups of data were used to train the network, and the last 111 groups of data were used to test the training results. The exact values in the experiment are accurately measured by gas composition analysis [47], which is used to compare with our predicted data. The maximum iteration time of the BP neural network is 1000, the error accuracy is 0.05, the learning rate is 0.001, and the activation function is selected as logsig. At the same time, in the PSO algorithm,

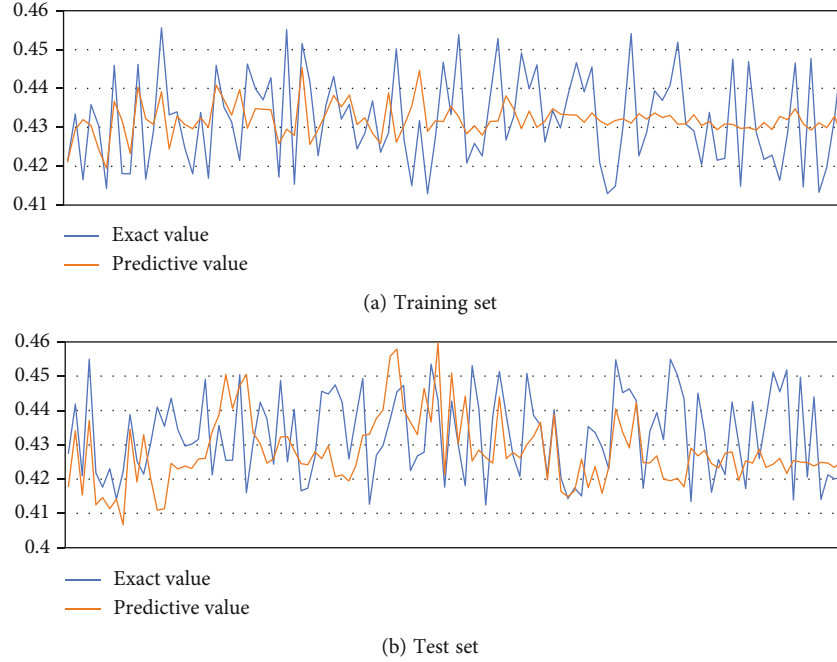


FIGURE 4: A single BP network model.

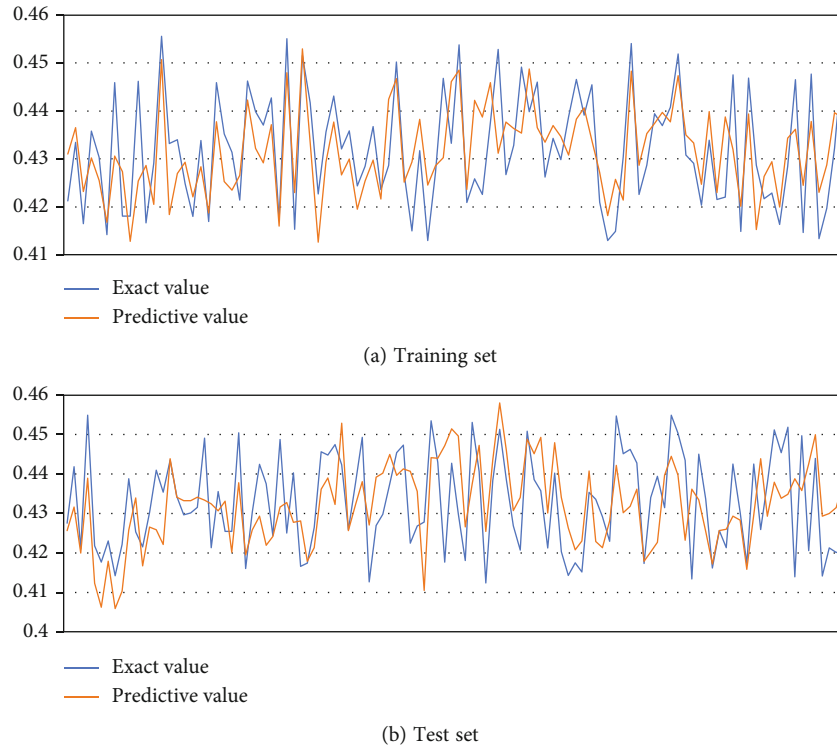


FIGURE 5: PSO-BP model.

the parameter  $c_1 = c_2 = 1.5$ , the velocity boundary is  $[-1, 1]$ , and the position boundary is  $[-5, 5]$ . The parameter  $D$  is set to 20, which indicates the maximum number of searches of the particle swarm algorithm. The definition of the regression coefficient is shown in Equation (11), and the value range is  $[0, 1]$ . The larger the value, the better the regression effect of the sample. In Equation (11):  $n$

represents the number of samples;  $x_i, y_i$  represents the sample data;  $\bar{x}, \bar{y}$  represents the sample mean.

$$R = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \times \sum_{i=1}^n (y_i - \bar{y})^2}}. \quad (11)$$

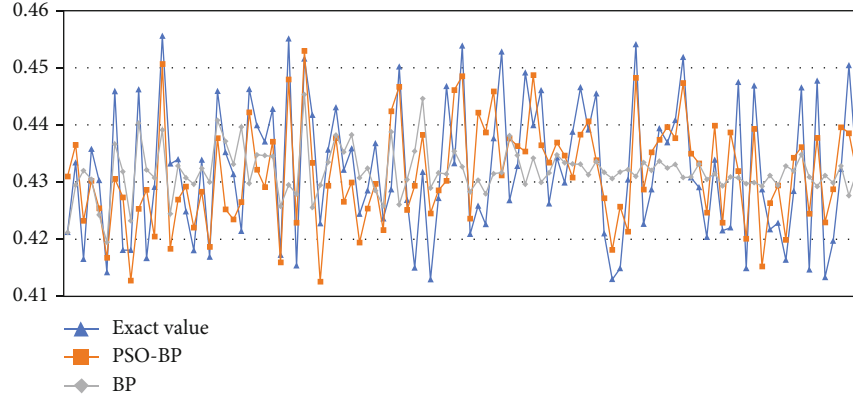


FIGURE 6: Comparison of the prediction model.

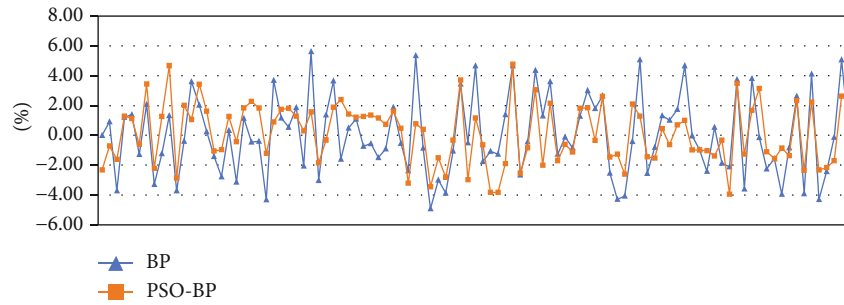


FIGURE 7: Comparison of prediction error.

Figure 4 shows the scatter plot of the predicted and actual values of air leakage rate trained by a single BP network model, and the regression coefficient at this time is  $R = 0.764908$ . From the training results, each test sample cannot effectively fit the actual value completely, and the accuracy still needs to be improved. Therefore, the combined model after the BP network is optimized by introducing the PSO algorithm is being studied.

Figure 5 shows the scatter plot of the actual value and the predicted value after the BP neural network is optimized by the PSO algorithm. The regression coefficient is  $R = 0.923661$ . It can be seen from the figure that the proposed model has good fitting ability.

To better understand the superiority of the proposed combined model prediction, the PSO-BP model is compared with the single BP neural network model, and the results are shown in Figures 6 and 7. According to Figures 6 and 7, it can be seen that the proposed BP neural network model optimized by the PSO algorithm has greatly improved the prediction accuracy. Because of the randomness of weights and thresholds in the BP network at the initial stage, the parameters are not optimal, so the optimal values of the particle population are searched for by the particle swarm optimization algorithm, and the optimal weights and thresholds in the network are calculated. The optimized parameters are then re-sent to the BP network for training and prediction, and the results are greatly improved. The latter parameters are sent back to the BP network for training and prediction, and the results have been greatly improved.

TABLE 1: Comparison of evaluation indexes of single BP and PSO-BP model.

Predictive model	RMSE	MAE	MAPE
BP	0.01131	0.00922	0.02128
PSO-BP	0.00864	0.00743	0.01718

To further verify the accuracy of the prediction, an error indicator can be introduced for further verification. The commonly used error indicators include root mean square error, average absolute error, and average absolute percentage error. Therefore, this paper uses these three evaluation indicators to evaluate the superiority of the model. The comparison of evaluation indicators is shown in Table 1. From the table, it can be clearly seen that the error index after the combination of the two models has been greatly improved based on the original single model, which provides an effective basis for verifying the accuracy of the combined model proposed in this paper.

#### 4. Conclusions

This paper aims to solve the problem of air leakage in the sintering process of the sintering furnace. We propose a neural network model based on PSO-BP and conduct a series of comparative experiments with the ordinary BP neural network model. The experimental results indicate that we have found promising weights and thresholds for the proposed neural network model through the designed optimization

method, where the regression coefficient of the training results is significantly improved. In addition, the indicator results referring to RMSE, MAE, and MAPE have demonstrated that the accuracy of the proposed PSO-BP neural network model has increased by 23.607%, 19.414%, and 19.267%, respectively, in comparison with the traditional BP neural network model. At present, this algorithm still has some shortcomings. The training of the BP neural network is often stagnant in the flat area of the error gradient surface, and the convergence speed is slow and may even fail to converge. We plan to improve it in the future by adapting the learning rate, increasing the learning rate where the error gradient is flat, and decreasing it otherwise, so that the algorithm can converge better.

### Data Availability

The code of the proposed algorithm and corresponding experimental data are provided. Interested readers please visit: <https://github.com/Darrenquan/Prediction-of-air-leakage-rate>.

### Conflicts of Interest

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

### Acknowledgments

This work was supported in part by the Fundamental Research Funds for the Central Universities No. N2117005, the Joint Funds of the Natural Science Foundation of Liaoning Province under grant No. 2021-KF-11-01, the Fundamental Research Funds for the Central Universities, the National Natural Science Foundation of China under grant No. 62103150, and the project funded by China Postdoctoral Science Foundation under grant No. 2021M691012.

### References

- [1] Z. Yuan, B. Wang, K. Liang, Q. Liu, and L. Zhang, "Application of deep belief network in the prediction of secondary chemical components of sinter," in *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pp. 2746–2751, May 2018.
- [2] S. G. Savel'ev, "Dependence of sinter strength on the sintering rate," *Steel Transl.*, vol. 41, no. 10, pp. 826–829, 2011.
- [3] X. Xue, J. Chen, and X. Yao, "Efficient user involvement in semiautomatic ontology matching," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 5, no. 2, pp. 214–224, 2021.
- [4] Y. Z. Wang, J. L. Zhang, Z. J. Liu, and C. B. Du, "Recent advances and research status in energy conservation of iron ore sintering in China," *JOM*, vol. 69, pp. 2404–2411, 2017.
- [5] L. Ma, S. Cheng, and Y. Shi, "Enhancing learning efficiency of brain storm optimization via orthogonal learning design," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 11, pp. 6723–6742, 2021.
- [6] D. H. Liu, H. Liu, J. L. Zhang et al., "Basic characteristics of Australian iron ore concentrate and its effects on sinter properties during the high-limonite sintering process," *International Journal of Minerals, Metallurgy, and Materials*, vol. 24, pp. 991–998, 2017.
- [7] L. Ma, M. Huang, S. Yang, R. Wang, and X. Wang, "An adaptive localized decision variable analysis approach to large-scale multiobjective and many-objective optimization," *IEEE Transactions on Cybernetics*, 2021.
- [8] C. Yang, T. Wei, X. Dong, Y. Li, S. Qu, and X. Li, "Sinter-hardening with concurrent improved plasticity in iron alloys induced by spark plasma sintering," *Journal of Materials Research*, vol. 29, pp. 981–988, 2014.
- [9] L. Ma, X. Wang, X. Wang, L. Wang, Y. Shi, and M. Huang, "TCDA: truthful combinatorial double auctions for mobile edge computing in industrial internet of things," *IEEE Transactions on Mobile Computing*, 2021.
- [10] Y. X. Xue, D. Q. Zhu, J. Pan et al., "Significant influence of self-possessed moisture of limonitic nickel laterite on sintering performance and its action mechanism," *Journal of Iron and Steel Research International*, pp. 1–13, 2022.
- [11] L. Lu and J. Manuel, "Sintering characteristics of iron ore blends containing high proportions of Goethitic ores," *JOM*, vol. 73, no. 1, pp. 306–315, 2021.
- [12] L. Ma, N. Li, Y. Guo et al., "Learning to optimize: reference vector reinforcement learning adaptation to constrained many-objective optimization of industrial copper burdening system," *IEEE Transactions on Cybernetics*, 2021.
- [13] A. Szewczyk-Nykiel and R. Bogucki, "Sinter-bonding of AISI 316L and 17-4 PH stainless steels," *J. of Materi Eng and Perform.*, vol. 27, no. 10, pp. 5271–5279, 2018.
- [14] L. Ma, X. Wang, M. Huang, Z. Lin, L. Tian, and H. Chen, "Two-level master-slave RFID networks planning via hybrid multi-objective artificial bee colony optimizer," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 5, pp. 861–880, 2019.
- [15] L. S. Pan, X. L. Wei, Y. Peng, X. B. Shi, and H. L. Liu, "Experimental study on convection heat transfer and air drag in sinter layer," *Journal of Central South University*, vol. 22, pp. 2841–2848, 2015.
- [16] J. S. Feng, H. Dong, J. Y. Gao, J. Y. Liu, and K. Liang, "Theoretical and experimental investigation on vertical tank technology for sinter waste heat recovery," *Journal of Central South University*, vol. 24, pp. 2281–2287, 2017.
- [17] X. Xue and J. Zhang, "Matching large-scale biomedical ontologies with central concept based partitioning algorithm and adaptive compact evolutionary algorithm," *Applied Soft Computing*, vol. 106, article 107343, 2021.
- [18] Y. Frolov, A. Nokesh, and L. Polotskii, "Study of flue-gas recirculation sintering," *Study of Flue-Gas Recirculation Sintering. Metallurgist*, vol. 61, no. 7-8, pp. 629–637, 2017.
- [19] C. Kuo, S. Qiu, and X. Yang, "A low-cost and highly efficient method of reducing coolant leakage for direct metal printed injection mold with cooling channels using optimum heat treatment process procedures," *International Journal of Advanced Manufacturing Technology*, vol. 115, no. 7-8, pp. 2553–2570, 2021.
- [20] W. Zhang, S. Wu, and Z. Hu, "Analysis of operational parameters affecting denitrification rate of sintering flue gas in cross-flow activated coke purification facility," *Journal of Iron and Steel Research, International*, vol. 27, no. 8, pp. 887–897, 2020.
- [21] C. Nahm, "Sintering effect on varistor properties and degradation behavior of ZVMB varistor ceramics," *Journal of*



- Materials Science: Materials in Electronics*, vol. 28, no. 22, pp. 17063–17069, 2017.
- [22] J. Park and C. Nahm, “Sintering effect on electrical properties and aging behavior of quaternary ZnO–V<sub>2</sub>O<sub>5</sub>–Mn<sub>3</sub>O<sub>4</sub>–Nb<sub>2</sub>O<sub>5</sub> ceramics,” *Journal of Materials Science: Materials in Electronics*, vol. 26, no. 1, pp. 168–175, 2015.
- [23] F. Y. Tian, L. F. Huang, L. W. Fan et al., “Pressure drop in a packed bed with sintered ore particles as applied to sinter coolers with a novel vertically arranged design for waste heat recovery,” *Journal of Zhejiang University-SCIENCE A*, vol. 17, pp. 89–100, 2016.
- [24] S. Sharma, H. Sharma, S. Kumar, S. Thakur, R. K. Kotnala, and N. S. Negi, “Analysis of sintering temperature effects on structural, dielectric, ferroelectric, and piezoelectric properties of BaZr<sub>0.2</sub>Ti<sub>0.8</sub>O<sub>3</sub> ceramics prepared by sol–gel method,” *Journal of Materials Science: Materials in Electronics*, vol. 31, pp. 19168–19179, 2020.
- [25] G. H. Chen, J. L. Li, X. Chen, X. L. Kang, and C. L. Yuan, “Sintering temperature dependence of varistor properties and impedance spectroscopy behavior in ZnO based varistor ceramics,” *Journal of Materials Science: Materials in Electronics*, vol. 26, pp. 2389–2396, 2015.
- [26] C. Nahm, “Sintering temperature dependence on microstructure and non-ohmic properties of ZVMND ceramic semiconductors,” *Journal of Materials Science: Materials in Electronics*, vol. 27, no. 9, pp. 9520–9525, 2016.
- [27] J. G. Fisher, M. G. Kim, D. Kim et al., “Reactive sintering of (K<sub>0.5</sub>Bi<sub>0.5</sub>) TiO<sub>3</sub>-BiFeO<sub>3</sub> lead-free piezoelectric ceramics,” *Journal of the Korean Physical Society*, vol. 66, pp. 1426–1438, 2015.
- [28] X. Xue and C. Jiang, “Matching sensor ontologies with multi-context similarity measure and parallel compact differential evolution algorithm,” *IEEE Sensors Journal*, vol. 21, no. 21, pp. 24570–24578, 2021.
- [29] Q. He, X. Wang, Z. Lei, M. Huang, Y. Cai, and L. Ma, “TIFIM: a two-stage iterative framework for influence maximization in social networks,” *Applied Mathematics and Computation*, vol. 354, pp. 338–352, 2019.
- [30] X. Y. Wu, J. R. Liu, Y. Chen, and M. H. Wang, “Effect of B<sub>2</sub>O<sub>3</sub> concentration and sintering temperature on microstructure and electrical properties in the ZnO–Bi<sub>2</sub>O<sub>3</sub>-based varistors,” *Journal of Electronic Materials*, vol. 48, pp. 7704–7709, 2019.
- [31] C. Nahm, “Microstructure and electrical properties of ZnO–Pr<sub>6</sub>O<sub>11</sub>–Bi<sub>2</sub>O<sub>3</sub>-based varistor ceramics with sintering changes,” *Journal of Materials Science: Materials in Electronics*, vol. 26, no. 11, pp. 8380–8385, 2015.
- [32] Z. Q. Tan, U. Engström, K. Li, and Y. Liu, “Effect of furnace atmosphere on sintering process of chromium-containing steel via powder metallurgy,” *Journal of Iron and Steel Research International*, vol. 28, pp. 889–900, 2021.
- [33] Z. Wang, Y. Huan, H. Wang et al., “The optimal sintering atmosphere and defect structure of CuO-doped NKN-based ceramic with p/n-type conduction mechanism,” *Journal of Materials Science: Materials in Electronics*, vol. 32, pp. 1928–1940, 2021.
- [34] L. Zhao, W. Sun, X. Li et al., “Assessment of particulate emissions from a sinter plant in steelmaking works in China,” *Environmental Monitoring and Assessment*, vol. 189, no. 8, pp. 1–16, 2017.
- [35] Y. Z. Wang, J. L. Zhang, Z. J. Liu, Y. P. Zhang, D. H. Liu, and Y. R. Liu, “Characteristics of combustion zone and evolution of mineral phases along bed height in ore sintering,” *International Journal of Minerals, Metallurgy, and Materials*, vol. 24, pp. 1087–1095, 2017.
- [36] W. D. Tang, S. T. Yang, L. H. Zhang, Z. Huang, H. Yang, and X. X. Xue, “Effects of basicity and temperature on mineralogy and reduction behaviors of high-chromium vanadium-titanium magnetite sinters,” *Journal of Central South University*, vol. 26, pp. 132–145, 2019.
- [37] J. Li, D. Bhattacharjee, X. Hu, D. Zhang, S. Sridhar, and Z. Li, “Effects of slag composition on H<sub>2</sub> generation and magnetic precipitation from molten steelmaking slag–steam reaction,” *Metallurgical and Materials Transactions B*, vol. 50, pp. 1023–1034, 2019.
- [38] M. Sharma and N. Dogan, “Dissolution behavior of aluminum titanate inclusions in steelmaking slags,” *Metallurgical and Materials Transactions B: Process Metallurgy and Materials Processing Science*, vol. 51, no. 2, pp. 570–580, 2020.
- [39] X. Xue, J. Lu, and J. Chen, “Using NSGA-III for optimising biomedical ontology alignment,” *CAAI Transactions on Intelligence Technology*, vol. 4, no. 3, pp. 135–141, 2019.
- [40] K. Zhang, F. Yuan, J. Guo, and G. Wang, “A novel neural network approach to transformer fault diagnosis based on momentum-embedded BP neural network optimized by genetic algorithm and fuzzy c-means,” *Arabian Journal for Science and Engineering*, vol. 41, pp. 3451–3461, 2016.
- [41] S. Xu, “Retraction note to: BP neural network–based detection of soil and water structure in mountainous areas and the mechanism of wearing fatigue in running sports,” *Arabian Journal of Geosciences*, vol. 14, no. 22, p. 2403, 2021.
- [42] G. Singh and K. Deep, “Effectiveness of new multiple-PSO based membrane optimization algorithms on CEC 2014 benchmarks and iris classification,” *Natural Computing*, vol. 16, no. 3, pp. 473–496, 2017.
- [43] D. K. Choubey, P. Kumar, S. Tripathi, and S. Kumar, “Performance evaluation of classification methods with PCA and PSO for diabetes,” *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 9, pp. 1–30, 2020.
- [44] B. Yang and L. Cheng, “Study of a new global optimization algorithm based on the standard PSO,” *Journal of Optimization Theory and Applications*, vol. 158, no. 3, pp. 935–944, 2013.
- [45] Y. Sun and Q. Zhang, “Optimization design and reality of the virtual cutting process for the boring bar based on PSO-BP neural networks,” *Neural Computing and Applications*, vol. 29, no. 5, pp. 1357–1367, 2018.
- [46] T. Bai, H. Meng, and J. Yao, “A forecasting method of forest pests based on the rough set and PSO-BP neural network,” *Neural Computing and Applications*, vol. 25, no. 7–8, pp. 1699–1707, 2014.
- [47] Y. Bernaldo de Quirós, O. González-Díaz, A. Møllerlækken et al., “Differentiation at autopsy between in vivo gas embolism and putrefaction using gas composition analysis,” *International Journal Of Legal Medicine*, vol. 127, pp. 437–445, 2013.

## Research Article

# Label Propagation Community Detection Algorithm Based on Density Peak Optimization

Ma Yan <sup>1</sup> and Chen Guoqiang <sup>2</sup>

<sup>1</sup>Department of Software Engineering, School of Computer and Information Engineering, Henan University, Henan Province, China

<sup>2</sup>Information Security Department, School of Computer and Information Engineering, Henan University, Henan Province, China

Correspondence should be addressed to Chen Guoqiang; [chengq08@163.com](mailto:chengq08@163.com)

Received 4 January 2022; Revised 2 March 2022; Accepted 10 March 2022; Published 28 April 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Ma Yan and Chen Guoqiang. 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.

Community structure detection in a complex network structure and function is used to understand network relations and find its evolution rule; monitoring and forecasting its evolution behavior have an important theoretical significance; in the epidemic monitoring, network public opinion analysis, recommendation, advertising push and combat terrorism, and safeguard national security, it has wide application prospect. A label propagation algorithm is one of the popular algorithms for community detection in recent years; the community detection algorithm based on tags that spread the biggest advantage is the simple algorithm logic, relative to the module of optimization algorithm, convergence speed is very fast, the clustering process without any optimization function, and the initialization before do not need to specify the number of complex network community. However, the algorithm has some problems such as unstable partitioning results and strong randomness. To solve this problem, this paper proposes an unsupervised label propagation community detection algorithm based on density peak. The proposed algorithm first introduces the density peak to find the clustering center, first determines the prototype of the community, and then fixes the number of communities and the clustering center of the complex network, and then uses the label propagation algorithm to detect the community, which improves the accuracy and robustness of community discovery, reduces the number of iterations, and accelerates the formation of the community. Finally, experiments on synthetic network and real network data sets are carried out with the proposed algorithm, and the results show that the proposed method has better performance.

## 1. Introduction

Community structure is a very important attribute in complex networks. Therefore, community structure plays a crucial role not only in the analysis of the social relations in human society [1] but also in the analysis of the functional relations between biological network organizations and organs [2], as well as the analysis of the citation relations between collaborative networks among scientists [3]. Therefore, the discovery of community structures from complex networks has been extensively studied in the past decade [4–8].

In 2002, Girvan and Newman achieved pioneering work pointing out that community structure is common in complex networks and proposed modularity  $Q$  to measure the stability of communities in networks [1]. Although the defi-

inition of community structure has not been unanimously determined by clear relevant studies, it is generally considered that a community is a group of nodes, which can also be called a community or a group of modules. These nodes are characterized by tight internal connections and sparse external connections [9].

As one of the hot spots of current research, community discovery algorithm based on label propagation has been widely used in community detection. This algorithm is a graph-based semisupervised learning method [10]. The advantage of semisupervised learning is that it can determine a lot of unlabeled samples by a small number of marked samples, thus improving the effectiveness of learning process [11, 12]. The basic idea of label propagation is to predict the label information of unlabeled nodes by using the topological relations between nodes from the label

information of labeled nodes and finally complete the division of the graph to form a clustering structure. Although this algorithm has the advantages of simple implementation, clear logic, no need to know the number of communities in advance, time complexity is close to linearity, etc., the unstable partition results and strong randomness are the defects of this algorithm. In each iteration of the label propagation algorithm, which community a node belongs to depends on the label with the largest cumulative weight of its neighbor nodes. Therefore, when more than one of the largest neighbor labels appears on a node, one of them will be randomly selected as its own label. This kind of randomness will bring avalanche effect, that is, a small clustering result error at the beginning will be continuously amplified. In addition, the updating order of node labels will also have a great impact. Obviously, the earlier the updating of the most important node will accelerate the process of convergence. In the label propagation algorithm, the closer the initial label is set to the core point, the more accurate the clustering effect is. However, in specific applications, it is often not feasible to know the number of communities in advance, and it is very inefficient to determine the number of communities ( $K$ ) by searching all possible candidate communities. Therefore, we are inspired by the density peak algorithm (DP) [13] and propose a label propagation algorithm based on density peak (DPLPA) for solving complex networks. The central idea of DP algorithm is that the core nodes are surrounded by other nodes in the same class, and there is no possibility for the core nodes to be closely connected. In other words, the core nodes have higher density, so this algorithm is feasible to calculate the core number. But unfortunately, DP algorithms cannot be directly used in a complex network, so DP algorithm is improved, and it can be applied to a complex network, can be reasonably come to the core number, applied to the label propagation algorithm, and according to the topology of the network that similarity matrix and priority to update nodes, reduce the randomness and the number of iterations.

## 2. Background

*2.1. Label Propagation Algorithm.* Raghavan et al. proposed the label propagation algorithm (LPA) [14], which used the label values of a few preset nodes to divide the community structure on a large-scale complex network. However, the accuracy of LPA is low because of the randomness of propagation, which leads to a large error in clustering results. The reason is that when the neighbor node label frequency appears with multiple highest values, the algorithm is fair to each label. We randomly select a label as the label of the update node. Therefore, the algorithm will appear small and fragmented communities or large communities which are not in line with the actual situation when the community is divided. Figure 1 is a situation where an error occurs in the label propagation process. The  $d$ -label finally appears in two communities, which is not in line with the actual situation.

In view of the problems of LPA, domestic and foreign scholars have proposed many improvement measures.

Tibély and Kertész [15] proved that the LPA will produce different community structures for the same network, and the algorithm still has a lot of randomness. Leung et al. [16] discovered the possibility of LPA application on tens of millions of networks and found the potential of large-scale data application of the algorithm. Barber and Clark [17] proposed the LPAm to solve the problem that the LPA cannot integrate different clustering results well by adding some restrictive conditions. Liu and Murata [18] solved the problem that LPAm was easy to fall into local optimal solution by optimizing the modularity. Zhuoxiang et al. [19] calculated the  $K$  value by calculating the potential influence of nodes. When the  $K$  value is less than the actual number of communities, the algorithm will not get the correct partition result. Xie and Szymanski [20] combined the label propagation algorithm with the Markov clustering algorithm (MCL) and proposed a new label propagation algorithm LabelRank. The biggest feature of the LabelRank algorithm is that a node can have multiple neighbor labels during the propagation process. Lin et al. [21] sorted the node weights and then updated the node labels in order. Zhang et al. [22] proposed a labeling algorithm based on edge clustering coefficient. Kipf and Welling [23] extended the graph-based label propagation algorithm and used graph convolution neural network for label propagation. The algorithm realized the propagation of label information through the aggregation of adjacent nodes. In addition, PageRank is used to quantify the importance of nodes, and LPaP algorithm [24] based on the importance of nodes is proposed. An improved community discovery algorithm based on feedback control [25], objective function [17], circle [26], and other methods for label propagation is proposed. The above algorithm is to optimize and improve the problem of node label in the propagation process, which can improve the stability and accuracy of LPA to a certain extent, but most of them bring more or less increased computational overhead, and do not achieve very ideal results.

However, Zhu et al. proposed another label propagation algorithm (LP) in reference [27]. They described the clustering problem as a form of propagation on the graph, in which the label of one node propagates to the neighboring nodes according to the similarity between them. In this process, LP fixes a small number of tags on the known label data. Then, the tagged data, like a signal source, pushes the label through the unlabeled data. Therefore, an accurate number of known tags will play an important role in the propagation process of LP algorithm, greatly improving the accuracy of clustering results.

The algorithm based on label propagation can be described as follows:

- (1) Propagation label:  $F = P \times F$
- (2) Reset the label of the core point in  $F:FL = YL$
- (3) Repeat steps (1) and (2) until  $F$  converges

Where step (1) multiplies the probability transition matrix  $P$  and the label matrix  $F$  to propagate the label of each node to other nodes with the probability of  $P$ . If the

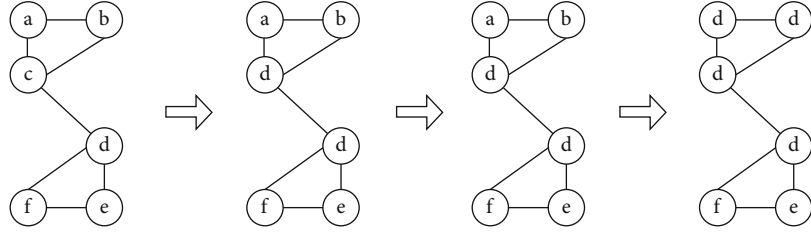


FIGURE 1: Example of the propagation of an error.

similarity between two nodes is very high, the easier it is for each other's label to be replaced by its own. Step (2) the most important thing is the known label, which cannot be changed, so every time it is propagated, it must return to the original label. As the label data point continues to propagate its label, the final class boundary passes through the high-density area and stays in the low-density interval. It is equivalent to the label node of each different category to divide the sphere of influence.

However, it is still an open question to determine the number of known labels. Traditional community detection algorithm can obtain the number of communities by optimizing the objective function or evaluation index. However, these methods are easily affected by many factors such as initial matrix and optimization objective function, so it is difficult to accurately determine the number of communities. In order to solve the above problem, we use an improved density peak clustering to obtain the kernel number as the input parameter of LP.

**2.2. Density Peak Algorithm.** In 2014, Rodriguez and Laio [13] proposed a density-based clustering method in Science, which can recognize clusters of various shapes, and the parameters are easy to determine. This method overcame the disadvantages of DBSCAN algorithm [28], which had large density differences among different classes and was difficult to determine the neighborhood range and had strong robustness. The core idea of the density peak algorithm (DP) is based on the assumption: for the center point of each cluster, the density of the cluster center point is greater than the density of surrounding neighbor points and the distance between the cluster center point and the higher density point is relatively large. Therefore, the DP algorithm has two quantities to calculate: the local density of the node and the distance from the high-density node. Usually,  $\rho_i$  is used to represent the local density of node  $i$ , and  $\delta_i$  is used to represent the distance between node  $i$  and the high-density node.

There are two ways to define local density  $\rho_i$ , one of which is

$$\rho_i = \sum_j \chi(d(i, j) - d_c), \quad (1)$$

where

$$\chi(x) = \begin{cases} 1, & x \leq 0, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

Here,  $d(i, j)$  represents the distance from node  $i$  to node  $j$ , and  $d_c$  is the cut-off distance, that is, the number of nodes whose distance to node  $i$  is less than  $d_c$ .

The second method is the Gaussian kernel function:

$$p_i = \sum_j \exp\left(-\frac{d(i, j)^2}{d_c^2}\right). \quad (3)$$

The minimum distance  $\delta_i$  between node  $i$  and other higher local density nodes is denoted by the formula defined as

$$\delta_i = \begin{cases} \max_j(d(i, j)), & \text{otherwise,} \\ \min_{j: \rho_j > \rho_i}(d(i, j)), & \text{if } \rho_j > \rho_i. \end{cases} \quad (4)$$

When all the nodes have calculated  $\rho_i$  and  $\delta_i$ , only the nodes with higher  $\rho_i$  and  $\delta_i$  can become the center points of the cluster, and the points with larger local density  $\delta_i$  but smaller local density  $\rho_i$  are abnormal points. The remaining nodes are assigned to the point with the highest local density among the neighbors.

Because the DP algorithm is a density-based clustering algorithm, it has the advantage of detecting clusters of arbitrary shape without the need to set the center point ( $K$  value) in advance. Moreover, when selecting the center point, the selection process of the center point can be visually seen through the decision graph. However, DP algorithm still has some defects. Firstly, the value of cut-off distance  $d_c$  needs to be set artificially, and improper setting will have a great impact on clustering results. Secondly, the central point needs to be selected artificially, so human subjective factors will affect the clustering results.

### 3. Methodology

In this section, the proposed label propagation based on density peak optimization clustering algorithm (DPLPA) is introduced. The core idea of DPLPA is to regard the high-density nodes surrounded by nodes of low-density neighbors as the community center points, and the distance between the community center points should be far away. In other words, a node with a higher density is more closely connected to its neighbors and is more likely to be the core point of the community. A community network is a complex network with connections between nodes, which usually reflects the network structure based on the connections between nodes. However, DP algorithm is a density-based clustering



algorithm that handles any shaped data set by calculating the distance between nodes to use high-density areas as a basis for judgment. But this way of calculating distance directly based on coordinates is not applicable to community networks. If the distance between nodes in community network is calculated, the similarity between nodes will become meaningless because the distance between nodes is more uniform or even the same. Therefore, DP algorithm cannot be directly used to detect community networks. In order to solve this problem, this paper uses the improved DP algorithm [29] to obtain the number of communities in a complex network as the input parameter of the label propagation algorithm.

**3.1. Predictive Fetch of Label Matrix.** Let  $G = (V, E)$  be a complex network with no direction and no weight. The node set  $V$  contains  $n$  nodes, the edge set  $E$  contains  $m$  edges, and the adjacency matrix of the graph  $G$  is  $A$ . If node  $i$  and node  $j$  have an edge connected, then  $a_{ij} = 1$  in the adjacency matrix  $A$ ; otherwise,  $a_{ij} = 0$ . Therefore, the node similarity formula of node  $i$  and node  $j$  is obtained, which is expressed by Salton index [30], also known as cosine similarity:

$$S(i, j) = \frac{|N(i) \cap N(j)|}{\sqrt{||N(i)|| \times ||N(j)||}}, \quad (5)$$

where  $N(i)$  and  $N(j)$  represent the neighbor nodes of node  $i$  and node  $j$ , respectively,  $|N(i)|$  represents the number of neighbor nodes of node  $i$ , so the molecular formula  $|N(i) \cap N(j)|$  represents the number of neighbors shared by node  $i$  and node  $j$ , while denominator formula  $\sqrt{||N(i)|| \times ||N(j)||}$  represents the number of neighbors expected to be shared by node  $i$  and node  $j$ . The value of  $S$  is between 0 and 1. When  $S$  is closer to 1, the similarity between the two nodes is very high. The formula for the distance between node  $i$  and node  $j$  is as follows:

$$d_{i,j} = \begin{cases} \frac{1}{S(i, j) + \sigma}, & i \neq j, \\ 0, & i = j. \end{cases} \quad (6)$$

Among them  $\sigma$  is a small positive number, in order to avoid the denominator being 0.

Next, we have two methods to calculate the local density of the node, one is to use the Gaussian kernel function, and the formula is as follows:

$$\rho_i = \sum_j \exp\left(-\frac{d_{i,j}^2}{d_c^2}\right), \quad (7)$$

where  $\rho_i$  represents the local density of node  $i$ ,  $d_{i,j}$  represents the distance between node  $i$  and node  $j$ ,  $d_c$  represents the cut-off distance, and the size of  $d_c$  is selected according to [13]. Then,  $\rho_i$  normalizes the value:

$$\rho^* = \frac{\rho_i}{\max_j \rho_j}. \quad (8)$$

Then, we start to define the distance formula between nodes:

$$\delta_i = \begin{cases} \max_j (d_{ij}), & \text{if } \max \rho_i, \\ \min_{j: \rho_j > \rho_i} (d_{ij}), & \text{otherwise.} \end{cases} \quad (9)$$

Among them, when the local density of node  $i$  is the largest, its distance is the maximum value of the distance between node  $i$  and other nodes. When the local density of node  $i$  is not the maximum, its distance is the distance between the node whose local density is slightly larger than that of node  $i$  and node  $i$ .

Then,  $\delta_i$  is standardized:

$$\delta_i^* = \exp\left(-\left(\frac{d_a^2}{\delta_i^2}\right)\right). \quad (10)$$

The threshold  $d_a$  is selected from the list of  $\delta$ , which is about 80% of the list of  $\delta$  from small to large [13].

Finally, take  $\rho^*$  as the X-axis and  $\delta^*$  as the Y-axis to generate a decision graph. Then, we calculate each node  $\gamma = \rho^* \times \delta^*$ , select a value greater than the sum of the average value of  $\gamma$  and the standard deviation of  $\gamma$  to enter the list, and then arrange them in order, and finally select the appropriate cut-off value as the core number (as the known label  $K$ ) and apply it to the label propagation algorithm.

**3.2. Label Propagation Algorithm Based on Density Peak.** LP is a graph-based clustering algorithm, so need to construct a graph first. The nodes of the graph are the data points. This paper uses the Gaussian kernel method to construct the weight between the two nodes:

$$w_{ij} = \exp\left(-\frac{d_{ij}^2}{\beta^2}\right). \quad (11)$$

Among them,  $d_{ij}$  is the distance between node  $i$  and node  $j$ , and  $\beta$  is the hyperparameter, and the similarity matrix composed of weight  $w$  is obtained.

Next, the known label is propagated through the edges between nodes. The greater the weight of the edge, the more similar the two nodes, and the easier the label is to spread. We define the probability transition matrix:

$$P_{ij} = P(i \rightarrow j) = \frac{w_{ij}}{\sum_{k=1}^n w_{ik}}, \quad (12)$$

where  $P_{ij}$  represents the probability of propagating the label of node  $i$  to node  $j$ . Since there are known  $K$  core points with known labels, a  $K \times K$  label matrix  $YL$  is defined. The  $i$  th row represents the label indication vector of node  $i$ , that is, if the label of the  $i$ th node is  $i$ , then the  $i$ th element is 1, and the rest are 0. It also defines an unlabeled matrix  $YU$



```

DPLPA
  Input:  $G = (V, E)$ 
  Output: Label matrix  $F$ 
1 Construct adjacency matrix  $A$  from complex network  $G = (V, E)$ .
2 Calculate node similarity  $S$  by Equation (5).
3 Calculate the distance matrix  $d$  between nodes by Equation (6).
4 Calculate the local density of the node  $\rho^*$  by Equations (7) and (8).
5 Calculate  $\delta^*$  by Equations (9) and (10).
6 Calculate  $\gamma = \rho^* \times \delta^*$  get  $K$  core points.
7 Get probability transition matrix  $P$  by Equations (11) and (12).
8 Build label matrix  $F$  by Equation (13).
9 while  $F$  convergence criteria not reached do
10:    $F = PF$ 
11:    $FL = YL$ 
12: end while
13:/*Iteratively update  $F$  until convergence, and the label change of the node has been very small. */
    
```

ALGORITHM 1: Gives the pseudocode of DPLPA.

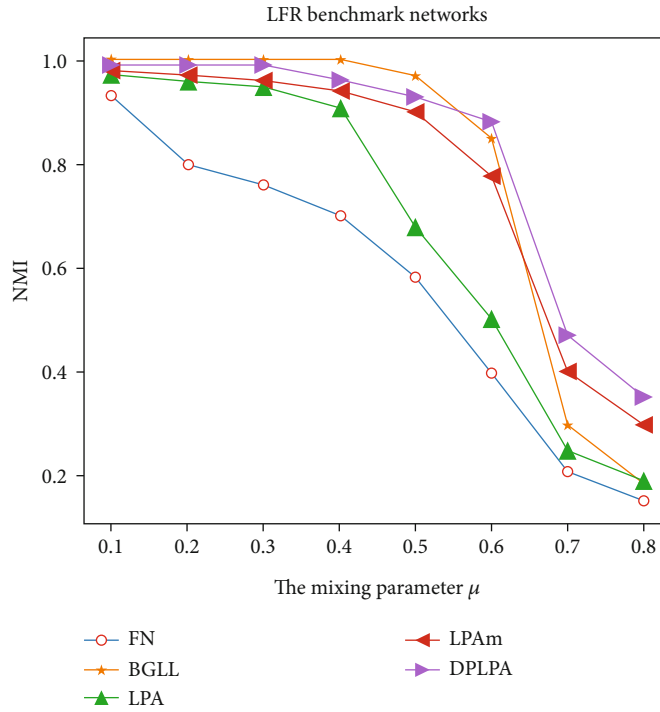


FIGURE 2: Experimental results on benchmark data set.

TABLE 1: Concrete description of real network.

Network	$n$	$m$	$k$	Descriptions
Karate	34	78	2	Zachary's karate club
Dolphins	62	159	2	Dolphin social network
Polbook	105	441	3	Books about US politics
Football	115	616	12	American college football

TABLE 2: Q value comparison of different algorithms in real network.

Networks	FN	BGLL	LPA	LPAm	DPLPA
Karate	0.3807	0.4188	0.3450	0.3496	0.3714
Dolphins	0.4955	0.5188	0.4788	0.4913	0.3789
Polbook	0.5020	0.4986	0.4953	0.4888	0.5063
Football	0.5497	0.6046	0.5445	0.5780	0.5539

of unlabeled nodes. We combine to get the label matrix of all nodes:

$$F = [YL, YU]. \tag{13}$$

Then, the label matrix  $F$  is propagated according to the similarity between nodes in the probability matrix  $P$ ; the formula is as follows:

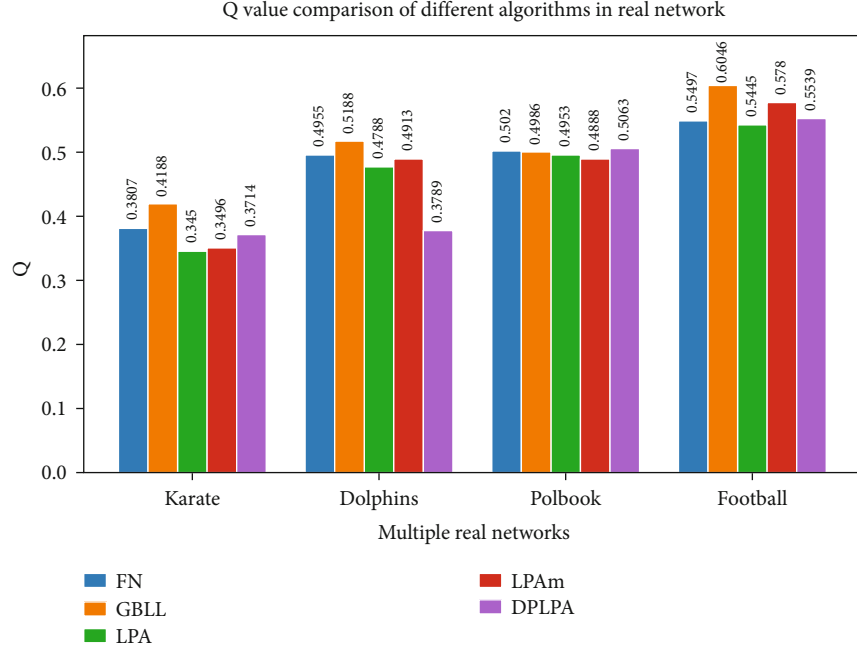


FIGURE 3: Modularity Q comparison of multiple algorithms on multiple data sets.

$$F = PF. \quad (14)$$

After the propagation, the  $YL$  in the known label matrix  $F$  changes during the propagation process, but  $YL$  is the label value we took for core nodes before, which is accurate and the label should not be changed. Therefore, need to reset the label matrix  $F$ , and the formula is as follows:

$$FL = YL. \quad (15)$$

Then, the label matrix  $F$  is propagated through the probability matrix  $P$  again, and the  $YL$  part in the propagated matrix  $F$  is reset. We iterate this process until the label change difference of  $YU$  in matrix  $F$  reaches a critical point. At this moment, DPLPA completes the label partition. Algorithm 1 shows the algorithm flow of DPLPA.

After obtaining the clustered label matrix  $F$ , the algorithm will gather the nodes with the value of 1 in the same dimension from  $F$  together to form a community. All nodes are divided according to the dimension. The clustering algorithm is finished, and the complex network is also divided.

## 4. Experimental Study

In order to assess our algorithm, we use a variety of real and synthetic data sets to test, and some classic methods to compare at the same time, including DPLPA in this paper, Newman fast greedy algorithm (FN) [31], Louvain algorithm (BGLL) [32], LPA [14], and improved label propagation algorithm (LPAm) [17]. The hardware environment of the experiment is as follows: Inter (R) Core (TM)i7-7700M CPU, 3.60 GHz, and 8 GB memory. The DPLPA is implemented in Python3.7 64-bit.

TABLE 3: Network actual grouping of football data sets.

Groups	Numbers
1	2 26 34 38 46 90 104 106 110
2	20 30 36 56 80 95 102
3	3 7 14 16 33 40 48 61 65 101 107
4	4 6 11 41 53 73 75 82 85 99 103 108
5	45 49 58 67 76 87 92 93 111 113
6	37 43 81 83 91
7	13 15 19 27 32 35 39 44 55 62 72 86 100
8	1 5 10 17 24 42 94 105
9	8 9 22 23 52 69 78 79 109 112
10	18 21 28 57 63 66 71 77 88 96 114
11	12 25 51 60 64 70 98
12	29 47 50 54 59 68 74 84 89 115

**4.1. Evaluation Metrics.** In this article, in order to verify the accuracy of the algorithm, we use the community discovery modularity function  $Q$  [31] proposed by Newman as the evaluation index of the experiment. Modularity is defined as

$$Q = \frac{1}{2E} \sum_{ij} \left[ A_{ij} - \frac{k_i k_j}{2E} \right] \theta(c_i, c_j), \quad (16)$$

where  $E$  represents the total number of edges of the community network,  $A$  represents the adjacency matrix,  $k_i$  represents the degree of node  $i$ , and  $c_i$  represents the community allocated by node  $i$ .  $\theta(c_i, c_j)$  is defined as follows:

$$\theta(c_i, c_j) = \begin{cases} 1, & (i, j) \in c, \\ 0, & \text{otherwise.} \end{cases} \quad (17)$$

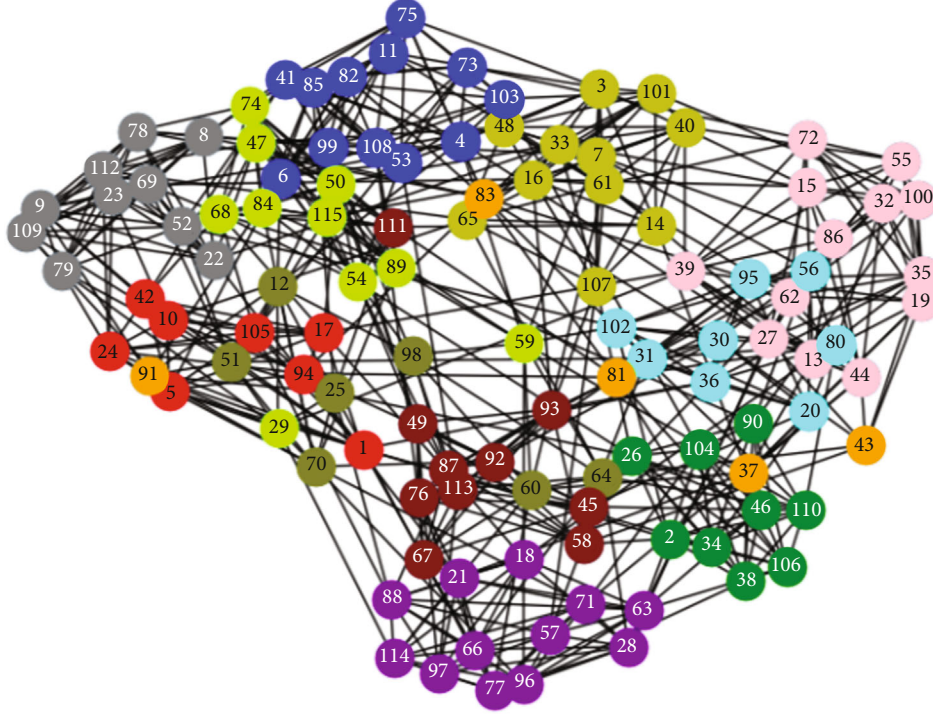


FIGURE 4: Partition of football data set by DPLPA.

Among them, when node  $i$  and node  $j$  are in the same community,  $\theta(c_i, c_j)$  is 1; otherwise, it is 0.

At the same time, we still use standardized mutual information (*NMI*) [33] to measure the similarity of two clustering results. It is an important measure of community discovery. It can basically objectively evaluate the comparison between a community division and a real division. For accuracy, the value range of *NMI* is  $[0, 1]$ , and the higher the value, the closer the divided community is to the real community result. *NMI* is defined as

$$NMI(A, B) = \frac{-2 \sum_{i=1}^{CA} \sum_{j=1}^{CB} C_{ij} \log(C_{ij}N / C_i C_j)}{\sum_{i=1}^{CA} C_i \log(C_i / N) + \sum_{j=1}^{CB} C_j \log(C_j / N)}. \quad (18)$$

Among them,  $A(B)$  represents the community discovery algorithm  $A(B)$ ,  $C$  is the confusion matrix,  $C_{ij}$  represents the number of nodes shared in the method  $A(B)$  partition,  $CA$  ( $CB$ ) represents the number of communities in the community discovery method  $A(B)$ , and  $C_i$  ( $C_j$ ) represents the  $i$ th row (column  $j$ ) in  $C$  and  $N$  represents the number of nodes. If the clustering results of methods  $A$  and  $B$  are the same, then  $NMI(A, B) = 1$ .

**4.2. Performance on Synthetic Networks.** The use of artificially synthesized networks to evaluate the effectiveness of the algorithm has become an effective means to test the pros and cons of the algorithm. Among them, the most used benchmark test network for community detection, LFR benchmark, was proposed by Lancichinetti et al. [34]. The

TABLE 4:  $K$  value comparison of different algorithms in real network.

Networks	FN	BGLL	LPA	DPLPA	True $K$
Karate	3	4	2	2	2
Dolphins	4	5	4	2	2
Polbook	4	3	4	3	3
Football	6	10	10	12	12

LFR reference network is an extension of the GN reference network [1] and has high practical value. The LFR benchmark network reflects the heterogeneity of community distribution and the power-law distribution of node degrees. Some of the important parameters are described as follows:  $n$  represents the number of nodes,  $k$  represents the average degree of nodes,  $\max k$  represents the maximum degree of nodes, and  $\min c$  represents the minimum community size,  $\max c$  represents the maximum community size,  $\tau_1$  and  $\tau_2$  represent the negative exponents of the power-law distribution of node degree and community size, respectively, and  $\mu$  is equal to the ratio of the number of connected edges between communities in the network to the total number of edges, to express the obvious degree of the community in the network; the smaller the  $\mu$  value, the more obvious the structure of the community. Figure 2 shows the comparison of the algorithm's *NMI* experiment results on the LFR benchmark data set.

The parameters set in this LFR experiment are  $n = 1000$ ,  $k = 15$ ,  $\max k = 40$ ,  $\min c = 20$ ,  $\max c = 50$ ,  $\tau_1 = 2$ ,  $\tau_2 = 1$ , and the range of  $\mu$  is from 0.1 to 0.8. It can be seen from Figure 2 that when  $\mu$  is small, that is, the community

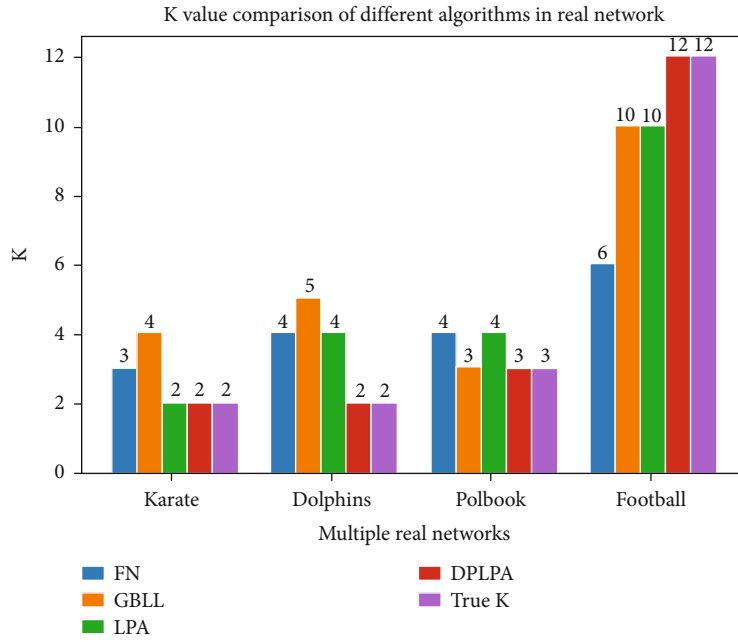


FIGURE 5: K value comparison of multiple algorithms on multiple data sets.

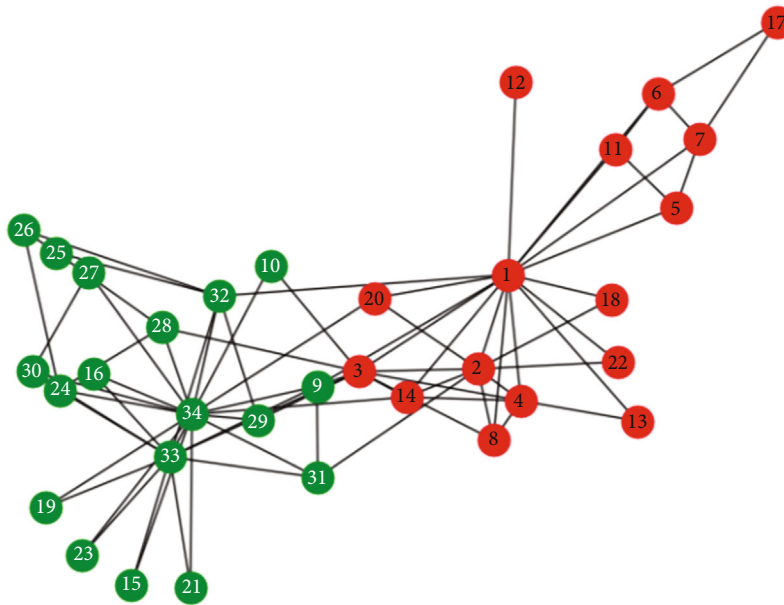


FIGURE 6: Partition of Karate data set by DPLPA.

structure of the complex network is obvious, the  $NMI$  values of the other algorithm results are high except for the FN algorithm. The  $NMI$  value of the FN algorithm and the LPA both began to decrease significantly. The remaining algorithms all began to decrease when the  $\mu$  value was 0.6, but the DPLAP decreased relatively slowly compared with the BGLL and LPAm, and finally, the  $NMI$  value is higher, so this can indicate that the DPLPA has a higher accuracy rate in community exploration and has better stability in high-complexity community exploration.

**4.3. Real-World Networks.** In order to further compare the pros and cons of the algorithms, this paper also tested the algorithm in a few real social networks. These networks are of different sizes but are representative and involve various fields. See Table 1 for details, where  $n$  represents the node,  $m$  represents the number of edges, and  $k$  represents the number of communities that have been identified.

Among them, Karate [35] is a data set of member relations of a university karate club in the United States, which is constructed based on the interactions between club

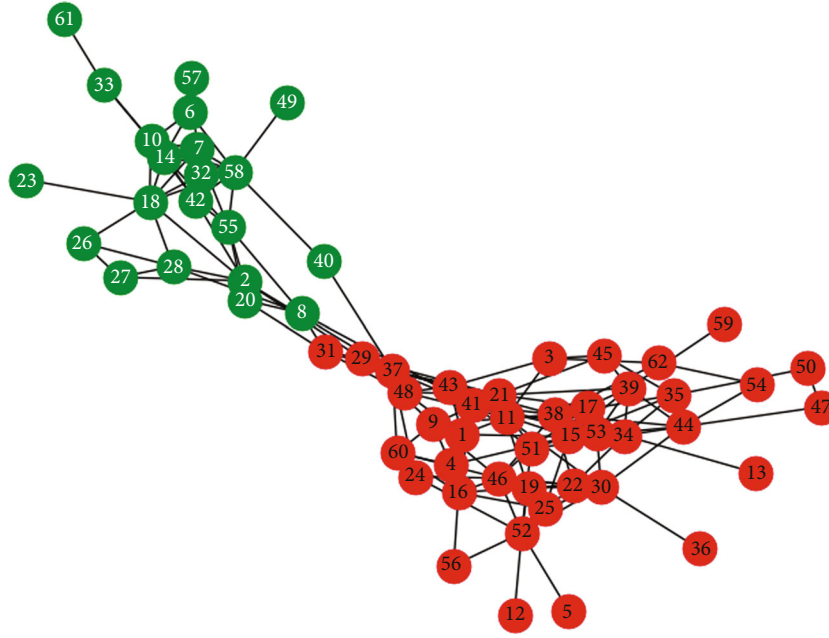


FIGURE 7: Partition of Dolphins data set by DPLPA.

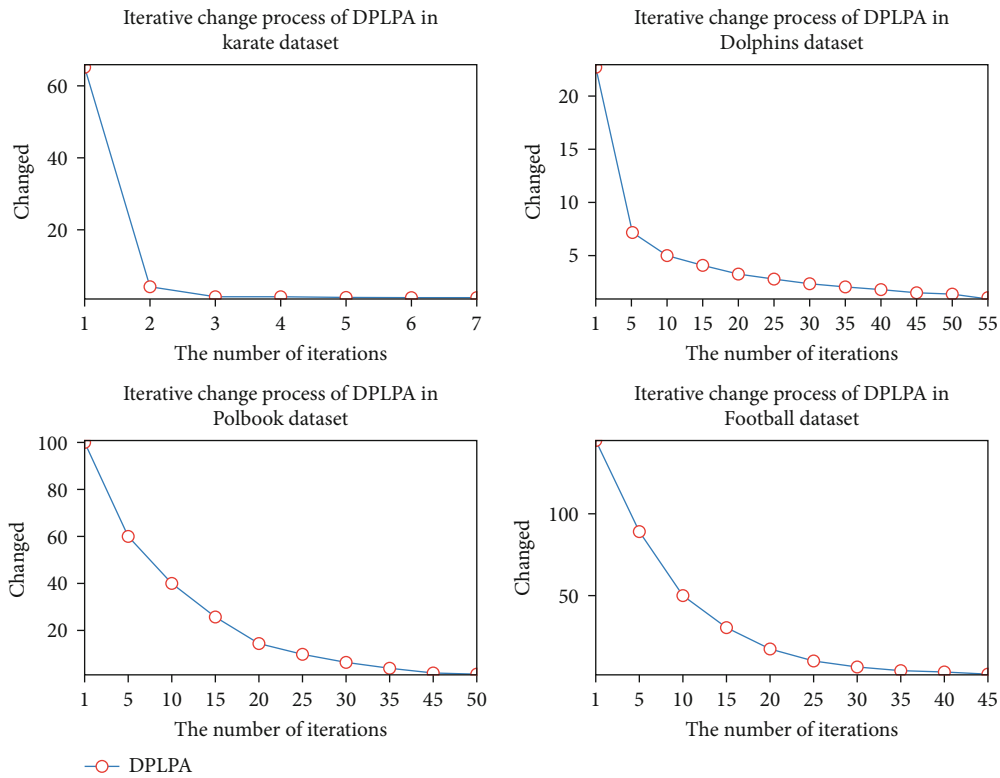


FIGURE 8: The changes of node labels in DPLPA algorithm during iteration.

members and is often used in the analysis of social networks. Dolphins [36] is a member network constructed from the living habits of 62 wide-mouthed dolphins, and the dolphins that are often together correspond to an edge between nodes. Polbook [37] is a community network constructed through political books sold on Amazon in the United States. Each node represents a book. If two books are purchased by the

same customer, there is an edge between them on the corresponding node. Football [1] is a network constructed by the American college football schedule. The nodes represent the participating teams. If there is a match between them, there will be an edge between the nodes. The calculation results of different algorithms on different networks are shown in Table 2 and Figure 3.



In order to better compare the clustering effect of DPLPA on the data set, this paper takes the Football data set to make a detailed explanation. The actual grouping of Football data set is shown in Table 3, and the clustering effect of DPLPA is shown in Figure 4.

It can be seen from Figure 3 that although the value of our method is not the best in some data sets, the division result of the DPLPA is the same as the actual community distribution, which can be seen from Table 3 and Figure 4. This is mainly because in the process of label dissemination, the probability transition matrix well suppresses the randomness of the dissemination process, so that each update of the node is updated to the label of the same community node as much as possible, making the result of community division more stable and closer to the real community situation. Comparison of  $K$  values of different algorithms on different networks is shown in Table 4 and Figure 5.

In addition, from Figure 5, we find that the DPLPA can detect the true number of communities, which is completely consistent with the actual  $K$  value. This is mainly because the DPLPA begins to calculate the local density and distance of nodes through the topology of the network at the very beginning and selects the number of  $K$  values through a decision graph. Therefore, we do not need to provide the  $K$  value, and the DPLPA has the advantage of detecting the  $K$  value.

In order to better show the experimental results, we use the Karate network and the Dolphins network as case studies to visualize the detected communities. Nodes in the same community are divided by the same color. Figure 6 is the visualization result of DPLPA division of the Karate network. Figure 7 is the visualization result of the DPLPA division of the Dolphins network.

It can be seen from Figure 6 that the local density of node 1 and node 34 is the highest, and it can be seen from Figure 7 that the local density of node 15 and node 18 is the highest, and these nodes have higher node distance, so it is very reasonable for the DPLPA to select these nodes as  $K$ , and the result of the division is completely consistent with the result of the actual community division. Therefore, we believe that the DPLPA is an algorithm that can perform high-quality community detection in real communities. In order to observe the convergence of DPLPA, this paper makes a comparison in multiple data sets, as shown in Figure 8.

Where the  $X$  axis is the number of iterations, and  $Y$  axis is the number of changes during node label iteration, as can be seen from Figure 8, in the process of Karate and Dolphins data clustering, the DPLPA has completed the division of most node labels after the first few iterations and then completed the division of a few nodes. In the process of Polbook and Football data clustering, the labels of most nodes have been partitioned until the 30th iteration. After that, the change curve of node labels becomes flat, indicating that all nodes have completed the label division and the algorithm has converged.

## 5. Concluding Remarks

In this paper, we propose a DPLPA for complex network community detection. It combines the characteristics of den-

sity peak algorithm and can predict the number of communities without a prior condition. It avoids the defects of random label algorithm, such as unstable division and strong randomness, and effectively improves the accuracy of community mining and the stability of the algorithm. In addition, the probability transition matrix is constructed to reduce the number of iterations of label propagation, so that the algorithm has efficient operation time, and finally can quickly find the network community structure. In the test results of the benchmark network and the classical real network, it is found that the proposed algorithm has better stability and accuracy than other advanced algorithms, and the number of communities found is always consistent with the actual number of communities in terms of the predicted  $K$  value. However, there is still room for improvement of the algorithm. In future research, we will face large-scale network data and further improve the time complexity of the algorithm. At the same time, dynamic network and overlapping network are also taken as research objects.

## Data Availability

The data used in “Label propagation community detection algorithm based on density peak optimization” is a commonly used data set to study complex networks, which can be queried on multiple data websites, for example: <https://snap.stanford.edu/data/>. <http://konect.cc/http://konect.cc/>. <https://networkrepository.com/index.php>. That is where I read the data I used in my experiments. New data availability url <http://www-personal.umich.edu/~mejn/netdata/>.

## Disclosure

Ma Yan and Chen Guoqiang current address is School of Computer and Information Engineering, Henan University, Henan Province, China. This work was outlined at the 2021 17th International Conference on Computational Intelligence and Security (CIS) [38].

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by the Key Science and Technology Program of Henan Province, China (Grant No. 162102210168). Group Name - on behalf of Key Science and Technology Program of Henan Province, China NO:162102210168 Affiliation - Belongs to Henan University, School of Computer and Information Engineering. Email Address - [hkjgg@163.com](mailto:hkjgg@163.com).

## References

- [1] M. Girvan and M. E. J. Newman, “Community structure in social and biological networks,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 12, pp. 7821–7826, 2002.

- [2] L. Hongchao, Z. Xiaopeng, L. Haifeng et al., "The interactome as a tree—an attempt to visualize the protein-protein interaction network in yeast," *Nucleic acids research*, vol. 32, no. 16, pp. 4804–4811, 2004.
- [3] M. E. J. Newman, "The structure of scientific collaboration networks," *Proceedings of the National Academy of Sciences*, vol. 98, no. 2, pp. 404–409, 2001.
- [4] D. Li, S. Zhang, and X. Ma, "Dynamic module detection in temporal attributed networks of cancers," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2021.
- [5] Z. Huang, Y. Wang, and X. Ma, "Clustering of cancer attributed networks by dynamically and jointly factorizing multi-layer graphs," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2021.
- [6] X. Ma, P. Sun, and M. Gong, "An integrative framework of heterogeneous genomic data for cancer dynamic modules based on matrix decomposition," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 19, no. 1, pp. 305–316, 2022.
- [7] W. Wenming, Z. Liu, and X. Ma, "jSRC: a flexible and accurate joint learning algorithm for clustering of single-cell RNA-sequencing data," *Briefings in Bioinformatics*, vol. 22, no. 5, pp. 1–15, 2021.
- [8] W. Wenming and X. Ma, "Joint learning dimension reduction and clustering of single-cell RNA-sequencing data," *Bioinformatics*, vol. 36, no. 12, pp. 3825–3832, 2020.
- [9] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3–5, p. 75, 2010.
- [10] W. Liu, J. Wang, and S. F. Chang, "Robust and scalable graph-based semi-supervised learning," *Proceedings of the IEEE*, vol. 100, no. 9, 2012.
- [11] Y. Chong, Y. Ding, Q. Yan, and S. Pan, "Graph-based semi-supervised learning: a review," *Neurocomputing*, vol. 408, pp. 216–230, 2020.
- [12] F. Nie, W. Zhu, and X. Li, "Structured graph optimization for unsupervised feature selection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 12, p. 1, 2019.
- [13] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.
- [14] R. U. Nandini, A. Réka, and K. Soundar, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 76, no. 3, 2007.
- [15] G. Tibély and J. Kertész, "On the equivalence of the label propagation method of community detection and a Potts model approach," *Physica A: Statistical Mechanics and its Applications*, vol. 387, no. 19–20, pp. 4982–4984, 2008.
- [16] X. Y. Leung Ian, H. Pan, L. Pietro, and C. Jon, "Towards real-time community detection in large networks," *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 79, no. 6, 2009.
- [17] J. Barber Michael and J. W. Clark, "Detecting network communities by propagating labels under constraints," *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 80, no. 2, 2009.
- [18] X. Liu and T. Murata, "Advanced modularity-specialized label propagation algorithm for detecting communities in networks," *Physica A: Statistical Mechanics and its Applications*, vol. 389, no. 7, pp. 1493–1500, 2010.
- [19] Z. Zhuoxiang, W. Yitong, T. Jiatang, and Z. Zexu, "A novel algorithm for community discovery in social networks based on label propagation," *Journal of Computer Research and Development*, vol. 48, 2011.
- [20] J. Xie and B. K. Szymanski, "LabelRank: a stabilized label propagation algorithm for community detection in networks," in *2013 IEEE 2nd Network Science Workshop (NSW)*, West Point, NY, USA, 2013.
- [21] Z. Lin, X. Zheng, N. Xin, and D. Chen, "CK-LPA: efficient community detection algorithm based on label propagation with community kernel," *Physica A: Statistical Mechanics and its Applications*, vol. 416, pp. 386–399, 2014.
- [22] X.-K. Zhang, X. Tian, Y.-N. Li, and C. Song, "Label propagation algorithm based on edge clustering coefficient for community detection in complex networks," *International Journal of Modern Physics B*, vol. 28, no. 30, p. 1450216, 2014.
- [23] F. T. N. Kip and M. Welling, *Semi-supervised classification with graph convolutional networks*, 2017, <https://arxiv.org/abs/1609.02907>.
- [24] L. Page, S. Brin, R. Motwani, and T. Winograd, *The page rank citation ranking: bringing order to the web*, Stanford Digital Libraries Working Paper, 1999.
- [25] Y. Li, H. Wang, J. Li, and H. Gao, "Efficient community detection with additive constraints on large networks," *Knowledge-Based Systems*, vol. 52, pp. 268–278, 2013.
- [26] M. Qianli and Z. Junhao, "A local strengthened multi-label propagation algorithm for community detection," *Computer Engineering*, vol. 40, no. 6, pp. 171–174, 2014.
- [27] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using Gaussian fields and harmonic functions," in *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003)*, Washington, DC, USA, 2003.
- [28] H. Bäcklund, A. Hedblom, and N. Neijman, *DBSCAN: a density-based spatial clustering of application with noise*, Linköpings Universitet-ITN, Data Mining TNM033, 2011.
- [29] H. Lu, Q. Zhao, X. Sang, and J. Lu, "Community detection in complex networks using nonnegative matrix factorization and density-based clustering algorithm," *Neural Processing Letters*, vol. 51, no. 2, 2020.
- [30] D. Martin, *Introduction to Modern Information Retrieval*, G. Salton and M. McGill, Eds., McGraw-Hill, New York, 1983.
- [31] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 69, no. 6, 2004.
- [32] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.
- [33] A. Estévez Pablo, T. Michel, A. Perez Claudio, and J. M. Zurada, "Normalized mutual information feature selection," *IEEE transactions on neural networks*, vol. 20, no. 2, pp. 189–201, 2009.
- [34] L. Andrea, F. Santo, and R. Filippo, "Benchmark graphs for testing community detection algorithms," *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics*, vol. 78, no. 4, article 046110, 2008.
- [35] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of anthropological research*, vol. 33, no. 4, pp. 452–473, 1977.

- [36] L. David, "The emergent properties of a dolphin social network," *Proceedings of the Royal Society B: Biological Sciences*, vol. 270, suppl\_2, 2003.
- [37] M. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 23, pp. 8577–8582, 2006.
- [38] Y. Ma and G. Chen, "Label propagation community detection algorithm based on density peak optimization," in *17th International Conference on Computational Intelligence and Security*, pp. 80–84, 2021.

## Research Article

# IOT Automation with Segmentation Techniques for Detection of Plant Seedlings in Agriculture

Shoaib Kamal <sup>1</sup>, K. R. Shobha,<sup>2</sup> Flory Francis,<sup>3</sup> Rashmita Khilar,<sup>4</sup> Vikas Tripathi,<sup>5</sup> M. Lakshminarayana <sup>6</sup>, B. Kannadasan <sup>7</sup>, and Kibebe Sahile <sup>8</sup>

<sup>1</sup>Department of Electronics and Communication Engineering, MVJ College of Engineering, Kadugodi, Bengaluru, Karnataka 560067, India

<sup>2</sup>M S Ramaiah Institute of Technology, MSR Nagar, Bengaluru, Karnataka 560054, India

<sup>3</sup>Department of Electronics & Communication Engineering, M.S., Ramaiah Institute of Technology, MSR Nagar, Bengaluru, Karnataka 560054, India

<sup>4</sup>Department of Information Technology, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, Tamil Nadu 600077, India

<sup>5</sup>Department of Computer Science & Engineering, Graphic Era Deemed to Be University, Dehradun, Uttarakhand 248002, India

<sup>6</sup>Department of Electronics & Communication Engineering, SJB Institute of Technology, Bengaluru, Karnataka 560060, India

<sup>7</sup>Department of Civil Engineering, B.S. Abdur Rahman Crescent, Institute of Science and Technology, Vandalur, Chennai, Tamil Nadu 600048, India

<sup>8</sup>Department of Chemical Engineering, College of Biological and Chemical Engineering, Addis Ababa Science and Technology University, Ethiopia

Correspondence should be addressed to Shoaib Kamal; [shoaibkamal87@gmail.com](mailto:shoaibkamal87@gmail.com) and Kibebe Sahile; [kibebe.sahile@aastu.edu.et](mailto:kibebe.sahile@aastu.edu.et)

Received 22 January 2022; Revised 6 March 2022; Accepted 10 March 2022; Published 25 April 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Shoaib Kamal 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 present work proposes to evaluate, compare, and determine software alternatives that present good detection performance and low computational cost for the plant segmentation operation in computer vision systems. In practical aspects, it aims to enable low-cost and accessible hardware to be used efficiently in real-time embedded systems for detecting seedlings in the agricultural environment. The analyses carried out in the study show that the process of separating and classifying plant seedlings is complex and depends on the capture scene, which becomes a real challenge when exposed to unstable conditions of the external environment without the use of light control or more specific hardware. These restrictions are driven by functionality and market perspective, aimed at low-cost and access to technology, resulting in limitations in processing, hardware, operating practices, and consequently possible solutions. Despite the difficulties and precautions, the experiments showed the most promising solutions for separation, even in situations such as noise and lack of visibility.

## 1. Introduction

The most widely used approach in image separation is to integrate its components based on their color similarities, i. e., color detection. In the agricultural environment, its application is widely used to identify pests and seedlings as opposed to soil and to identify fruits due to their characteristic colors. Access to cameras is easier by obtaining a color

image by providing more information about the environment than monochrome cameras, and the comparative simplicity of the color approach is advantageous compared to approaches that are separated by shape and structure application of these techniques [1].

However, this color separation technique still has many challenges, especially when it is used outdoors, due to the difficulties in working in an uncontrolled environment. [2]

In particular, the effects of lighting and the presence of shadows are important sources of variation in the visible properties of the object, making detection and identification very unstable and difficult, especially in terms of color. To overcome these challenges, image processing techniques and methods can be developed and improved, removing noise associated with ambient conditions and extracting only useful information from the image [1, 3, 4].

Approaches to removing the effects of a light source on the colors of an object are called color consistency methods. Proposals for solving problems described on the basis of assumptions about visual, optical, capture device, or their combinations [5] are very different. However, despite the many proposed mechanisms for color stability, no single solution has been identified, the possibilities of applications and environmental conditions are so vast and diverse that they prevent the global solution from being achieved, and thus approaches are highly dependent applications and specific issues [6, 7]. Recent advances have shown promising results through convolutional neural networks and in-depth learning [8–10], but its operation requires training data and relatively high computational costs, restricting its use.

Because the agricultural environment is a highly controlled universe of conditions and applications, there are certain approaches to treating light variations. Many spaces and color codes were created and used for color pictures for the plant segment, based on specific conditions, such as the natural difference between plants' green and soil in general. However, again, there is no evidence for a single approach to problem solving because the conditions of the experiments are so specific that the methods are difficult to compare, and the accuracy of the documents is unclear [11].

Despite the different approaches to color coding, there is a lack of use of broad and traditional methods for light stability in the agricultural environment. Another small explored feature is the estimation of processing times and cost of methods, while the hardware used is less discussed and processing is rarely carried out on embedded units, making it difficult to analyze performance and reliability all-in-one view settings.

Considering what has been presented here, it can be observed that the path to technological development in agriculture is very promising because; in addition to the challenges of mitigating environmental impacts, most productive plantations of global production are below its potential [12]. Similarly, the applications of color stabilization algorithms can be shown to be an interesting alternative to the use of vision systems in outdoor and agricultural environments. Although there are different types of algorithms for color consistency, the solutions are very specific to the applications and operating conditions of the vision system. Developments to explore these nodes should ensure a more stable environment and an equal global food supply [13].

## 2. Implementation

The determination of the project hardware was the main motivation for accessibility, cost, and strength in aligning with the proposed objectives of the project. A wide range

of comprehensive and basic hardware solutions with acceptable performance and affordable price were sought in the market. In addition, more than one solution was considered in the study, which aims to expand compatibility and ensure greater consistency to its results. In this excuse, two different capture settings were defined: a Logitech USB webcam model C615 and a Raspberry Pi model Raspicam rev1.3 embedded system module. Both devices have RGB sensors, and their specifications are summarized in Table 1.

For some analyzes, two different processing systems with different operating systems and operating systems were evaluated, and their configurations can be found in Table 2. The two systems proposed, although very accessible, and have very different characteristics. The first desktop personal computer (PC), running on the Windows operating system, was fully compatible with the webcam capture device but could not communicate directly with the RaspiCam module [14, 15]. The second smaller and cheaper module, the Raspberry Pi 3 Model B (Rpi), is worth approximately 35 USD as of the date of the study and is offered as one of the most popular alternatives for embedded applications: credit card size, 45 g, 5 volt and 4 W power, audio output, HDMI video, and a built-in Wi-Fi internet adapter [16]. In addition to the hardware features, the module runs an open and free operating system based on Debian, and the system is compatible with both its dedicated camera, RaspiCam and webcam.

As for the software, again, two different approaches were used in [17]. Advanced programming software Wolfram Mathematica (WM) was selected for the development of algorithmic logics and fast performance evaluations. It has a wide range of imaging tools that greatly speed up algorithm development time. In addition, the software was used as a second measurement system to analyze the performance of research approaches, i.e., to provide information on the reproduction of algorithms, which were evaluated in parallel to the main one.

The main software was implemented using the Python programming language, a relatively high-level language with excellent transparency, autonomy, and computational power, in [18] addition to being widely spread and recognized that the recognized language is not arbitrary, and its use aims to ensure the proposed access through research, ensuring its compatibility and ensuring the portability and usability of algorithms in different systems. In this way, test codes can use the same codes, regardless of the hardware and operating system used [19–21].

The development of algorithms in Python relies on the use of the OpenCV (Open Source Computer Vision) programming functions library, which supports computer vision applications for a free and open source, educational, and business use. Its development, based on computational skills and application in real-time applications, builds a set of all the basic tools and structures needed for image capture, manipulation, and processing throughout the study.

## 3. Experimental Results

The test campaign was basically divided into two stages, the initial stage for analyzing the test parameters, the second



TABLE 1: Capture device specifications.

Description	Logitech C615	Raspicam rev1.3
Maximum resolution	1920 × 1080	2592 × 1944
Field of view	74° (diagonal)	53.5°H/41.4°V
Maximum acquisition rate	30 fps at 640 × 480	60 fps at 640 × 480
Focal length	Not informed	3.6 mm
Light correction	Yes	Yes
Mass	138 g	3 g
Price	70 USD	25 USD

TABLE 2: Proportional increase in processing through the use of morphological operations.

Resolution	PC	Rpi	WM
1280 × 960	13.32%	87.45%	36.65%
640 × 480	7.05%	82.50%	47.55%
320 × 240	8.60%	82.60%	113.10%
Average	12.05%	86.20%	42.18%

tests for validating the measurements, and evaluating the measurement method and comparing the division strategies. These two moments of the project share the same capture and acquisition method but use different infrastructures.

Several components need to be defined and adjusted to ensure the representation of the experiments with the application and consequently the validity of the study results. Infrastructure features cover everything from the implementation to the elements that make up the scenery for creating experimental images. Its character and reason for study are given below. Information about the configurations of these elements in the experiments will be described in Section 3.6.

Following the section of the test campaign discussed, two different test sites were used, the first being an indoor environment with strict control of lighting conditions. The second approaches the application stage, under experiments in the external environment and under natural light conditions.

The interior environment consists of a room with white walls without windows, with only light coming from artificial lighting. The room was illuminated by two lamps with a color temperature of 6800 K, i.e., colored equal to daylight, and a lamp with a temperature of 3000 K was used to simulate light at the end of the day, all with a high reproducibility ratio of fluorescent and color (IRC). In this configuration, with approximate brightness of 161.0 lux and constant distortion of 5.4 lux was the object of interest throughout the experiments, this variation is within the expected range for devices.

The external environment, on the other hand, is created by an open environment with exclusive exposure to natural light and, therefore, subject to its variations. Seizures at this stage occurred during the first week of November 2018, and all of this was carried out between 19:30 and 19:15 to the day level (evening) between 14:30 and 15:30 hours and between 11:45 and 19:50 in the main light (at dusk).

The material of interest in the pictures created by the plant seedlings characterizes an important aspect of the divi-

sion process and, consequently, the experiments. In preliminary studies, three types of seedlings were selected with deliberately different characteristics of structure, size, leaf shape, and mainly color. Selected seedlings can be found in coriander leaves, mint, and curry leaves (Figure 1).

For comparative testing, it was decided to use seedlings from cultures that were more explicit in the national context to bring more relevance and applicability to the results; soy and corn seedlings were used. These, in addition to their high agricultural importance, have distinctive characteristics of color and texture, with corn seedlings being soft and slightly lighter green shades and soybean seedlings being coarse and dark (Figure 2).

In the background of the scene, the soil is mostly made of clay soil, also known as terra roxa, as it is in line with the proposals in the study area (State of Tamil Nadu) and is one of the most common soils seedlings for testing.

In some experimental stages, small portions of the organic substrate, naturally black, were added to create visual disturbances, as well as coconut fiber, pine bark, spruce, dried leaves, sticks, and small stones. These components were added to simulate unpredictable and complex conditions in the final application, commonly referred to as debris or visual noise testing. To reduce the variance between test conditions, soil and other debris were placed in a plastic reservoir, and for each capture stage, the seedling was transplanted to the center of this situation.

Furthermore, the position of the camera relative to the base is guaranteed by the support attached to the height-gain system and the adjustable angle, which ensures control over the capture pose. Finally, a reference object was developed to classify lighting conditions in an open environment and to implement color consistency through the reference method. The instrument consists of an image of 4 squares, each in pure colors, white, red, green and blue, and a 40 mm dark gray sphere, which ensures that light is captured from different angles by minimizing the concentration points in your grip. This last component was inspired by the work of [13–21]. Figure 3 shows the note in our application.

#### 4. Metrics

This topic explores the criteria used to evaluate and differentiate sectional approaches, i.e., we currently define test response variables based on planned experiments.



FIGURE 1: Coriander leaves, mint, and curry leaves seedlings, respectively.



FIGURE 2: Soybean and corn seedlings, respectively, in soil with debris.



FIGURE 3: Use of the reference in the open and cloudy weather conditions, respectively.

Evaluation focuses on two different, one aimed at evaluating unit performance, measuring aspects such as sensitivity, errors, and accuracy. The other looks at the delay of the calculated cost and approaches. Both responses were calculated in standard scenes of the scene, not in the videos. However, it is important to emphasize that this approach is appropriate for the application. The codes generated using the OpenCV library process the images from the cameras as well as the sequence of photos.

**4.1. Computational Performance Metrics.** For performance evaluation, techniques related to classification measurements were used. If the partition is considered a binary classification function, in other words, as a result of the partition, each pixel of the image is immediately marked as belonging to the plant group or background group.

Several classification measurements were used to evaluate specific features of the results, but due to the extensive

data size of the test campaign, it was decided to use a single measurement. In the first analysis, a total error was used, but after some evaluation of the measurement method, the  $F$ -score metric was chosen to compare the results. The latter proved to be a good alternative, with a high representation of segment performance, summarizing the critical accuracy and sensitivity information of approaches at the same value.

Also known as the  $F$ -score, the  $F$ -rating is defined as the corresponding average between accuracy and sensitivity and other measurements discussed in its formation and sequence:

$$\text{Total error} = \frac{FP + FN}{(TP + TN + FP + FN)} = \frac{FP + FN}{\text{Total elements}}, \quad (1)$$

$$\text{Sensitivity} = \frac{TP}{(TP + FN)}, \quad (2)$$

$$\text{Precision} = \frac{TP}{(TP + FP)}, \quad (3)$$

$$F\text{-score} = \frac{2 * \text{precision} * \text{sensitivity}}{(\text{precision} + \text{sensitivity})} = \frac{2 * TP}{(2 * TP + FP + FN)}. \quad (4)$$

TP (true positive) is the number of true positive classifications, TN (true negative) is the number of correct negative classifications, and FP (false positive) is the number of false classifications such as false positive (type I error) and FN (false negative) and number of lost classifications of the object of interest (Type II error).

In a practical way, total error refers to the ratio of misalignment relative to total, which, although intuitive in use, depends on the size of the object of interest in the scene, which creates an unwanted relationship. Results and discussion. Sensitivity or recall, on the other hand, provides the ratio of the exact classifications relative to the total number of pixels owned by the plant; so, the higher the sensitivity of the approach, the more likely the plant is to be properly classified. However, false positive errors are not considered. Accuracy observes the ratio of positive and correct classifications relative to the sum of the positive classifications, i.e., the higher the accuracy, the more certain that a classification as a plant actually belongs to this class. Its calculation evaluates the relevance of the exam but does not look at the absolute ratio of the correct answers.

The  $F$ -score metric combines the properties of both measurements and mitigates their main implications; so, only high-sensitivity and precision approaches can yield good results. This measurement covers the area from 0 to 1, where 1 is the exact classification and 0 is the absence of true positives. This measurement proved to be very strong to produce uniform performance results and show variation regardless of the distance of the object in the scene.

To calculate the classification criteria, it is necessary to use references to determine whether the image pixel classification is accurate or not, so that for each image evaluated in the study, a real image will be created manually and carefully

by experts and acted upon as a standard.. The best part of the plant is on display.

Considering that one of the possible purposes of separating plant seedlings is to guarantee the recognition and monitoring of this material, a second performance indicator was used, which estimates the distance between the centers of the material separated by a given method. This distance is determined by the difference in pixels between the centers in the  $x$  and  $y$  directions of the adjusted images as a function of the diagonal size of the image. Creating a percentage error result that does not affect the image size is specified as follows:

$$\text{Distance error}(\%) = 100 \sqrt{\frac{\Delta x_p^2 + \Delta y_p^2}{\text{width}^2 + \text{height}^2}}. \quad (5)$$

With  $\Delta x_p^2$  and  $\Delta y_p^2$ , the difference in pixels from the center of the segmented object in relation to the center of the reference on the  $x$  and  $y$  axes of the image.

**4.2. Computational Performance Metrics.** The calculation was performed by measuring the operating times of the algorithms based on the performance rating. For this purpose, high-precision time counters were implemented in specific functions associated with each approach, so that the common processing for all approaches, such as capture, output image recording, and interface commands, was not calculated.

It only considers aspects related to timing, including color sampling functions, required data manipulations and transitions, scale adjustment, color rearrangement functions, entry functions, classification, and other required processing. Postprocessing steps such as color stabilization approaches and postprocessing were monitored separately from the section process.

To guarantee greater accuracy and reliability in processing time estimation, the processing value calculated by an average of 100 consecutive processes for each approach was calculated during the acquisition process by the processing unit and all data in the same context.

## 5. Result Analysis

Due to the causal structure of the experiment, most researchers used common tools and techniques for this type of approach. Results are often evaluated graphically and occasionally analytically using statistical tools.

In graphic analysis, all the data of the test are arranged in a graphic to facilitate a practical analysis of the results: the  $Y$  axis contains the answers, and the  $X$  shows the test conditions by the titles of the appropriate groups of the test. This method of compiling data classifies a map into a variance chart or a multivariable chart.

For quantitative analyzes, the effects of factors and the correlations between factors were calculated. The response to change at the factor level has a numerical effect indicating the change of the variable, i.e., explaining the cause-and-effect relationship between the factor and the experimental response. Its calculation is simple and is based on the sum

of the values of all the test stages, in which the factor or correlation level is one (+) with the conditions at level two (-). Its calculation can be described as follows:

$$\text{Effect} = \frac{\Sigma \text{ values at level one} - \Sigma \text{ values at level two}}{\text{number of observations}/2}. \quad (6)$$

After calculating all the outcomes of the experiment, an occasional Barreto chart facilitates the comparative visualization of the results, thus highlighting the largest contributors to the change in the response variable of the experiment.

## 6. Acquisition System

Despite the different test conditions, the computer and data used in the study are consistent and configure the basic functions of the test campaign. As described in the Infrastructure section, the plants are mounted on the bases and the device attached to the capture system is attached, which ensures the display and pose of the images.

The capture system is characterized by the Raspberry Pi embedded module, a capture device (webcam or rospecam) and battery, which ensures portability on the system and enables its use in the open environment. Capture commands are carried out by remote access to the computer via a VNC (Virtual Network Computing) connection over a wireless local network using a portable router, which allows the camera to display the image and control the capture time (Figure 4). Images are then stored in the mobile device's memory and stored for analysis.

Capture software was developed in Python and followed the acquisition process used in the application, aimed at accessing customized images for higher capture speed and processing. Therefore, the captured image samples are more relevant to those obtained in practice. Additionally, the program allows you to manually configure the parameters of both cameras, allowing you to calibrate and adjust the gain of their channels, as well as enable or disable the automatic white and brightness adjustment functions of the cameras.

The captured images were classified as test samples. These were submitted to preprocessing or section approaches, eventually forming binary images, which were exported with their average processing time. Two versions of this program were created, one on Python, the other on Wolfram Mathematics, and the Python version on both processors.

These images were then evaluated with their actual image, and the data were finally compiled into spreadsheets for analysis, generating final answers to the functions of a program section in mathematics and the computational performance measurements described in the previous sections.

In other words, this first inquiry aims to understand the problem of comparing approaches and to assess the conditions and configurations that may affect this process. Table 3 below provides an overview of the rated components and capture system with internal environment, 400 mm height, and Rpi + Raspicam.

In the results, we looked at the impact of plant characteristics, the impact of capture resolution on measurements, the

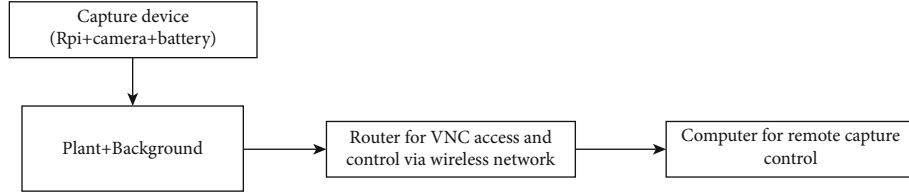


FIGURE 4: Scheme of the capture system used in the experiments.

TABLE 3: Experimental conditions.

Variables	Condition (1)	Condition (2)	Condition (3)	Condition (4)
Object	Mint	Curry leaves	Coriander leaves	
Resolution	320 × 240	640 × 480	1280×960	
Processor	Rpi + OpenCV	PC + OpenCV	PC + WM	
Segmentation	Otsu	Hue segmentation	Modified hue interval	Excess green

relationship between different processors and software effects, and the representation of total error (%) and bot section performance measurements based on the calculation. Check the performance generated based on the processing time (MS) to measure the performance and also check the gain generated by the simulation process in subsequent processing.

This first test was carried out indoors under control lights, disabling the standard pose and white camera's automatic pose and exposure, which minimizes test variations as a result of lighting. Its morphological postprocessing and other experimental approaches have completion functions following the opening by a  $3 \times 3$  square structural element. Furthermore, three sample repetitions were performed for each condition, i.e., three consecutive recordings of the scene. Eventually, 30 different images of the scene were obtained. These were used in 4 section approaches, with two image states on three different processors, a total of 650 processes, each with processing time data and total section error.

## 7. Results

The experiment used the experimental framework of variance component analysis, with three corrections, to evaluate the measurement method and graphic analysis of the experiment (Figure 5).

Groups A1 to A4 correspond to the conditional 1 to 4 section approaches in Table 4, respectively.

With data covering values from less than 1 ms to more than 1 s, a large variation in processing times was initially observed. Many elements produced changes in the results of time, demonstrating the sensitivity of the measurement system and showed repeated values at each test stage, and their variance was significantly smaller than the variances found on the system. The estimated standard deviation for the duration of this test is 0.110 ms, which demonstrates the high accuracy of the submitted test. However, the occurrence of some erroneous points proves that the processing is not uniform enough, and that the increase in the number of

repetitions will benefit the metric and, consequently, the results.

In turn, it should be noted that the main contributors to the difference in computational times are processors, with OpenCV placing a clear emphasis on system performance and, for worst results, Raspberry Pi executing on the embedded module. By checking the algorithms, it also verifies that the material in the same process does not have a significant impact over time, which shows signs of visual freedom at the calculated cost. These graphical observations can be found in Table 5.

Furthermore, it is noteworthy that other factors also contribute to the variability in results. To better illustrate the contribution of the sources of variation, the diagram in the time results normalized by the mean value of each processor.

For these maps, the mean values (green horizontal lines) correspond to the approaches (from A1 to A4), which explains the apparent difference they make in the process; so, the higher green approach will take longer, followed by the HSV transition, with Otsu is directly expecting only HSV and short-term function. In addition, by adjusting the scale, the regeneration of time between processes is greater, and assumptions and evaluations of approaches and results can be made under any process so that the answers are sufficiently similar, thus making use reasonable. Very powerful processors for prototype or preliminary evaluations of the performance of approaches in embedded modules.

However, this assumption is not exhaustive, there are processing differences, and it is worth noting that real-time can only be obtained on the target processor, which becomes clear when observing the effects of image operations between processes. Its use constantly increases the calculation cost for all conditions, but its contribution to the algorithm time for each processor is different as seen in Table 2.

This difference may be due to architectural differences, different levels of memory, and the mathematical processing of its variables. As an important observation, this type of function presents a significant computational challenge for the embedded module, although relatively small (14%



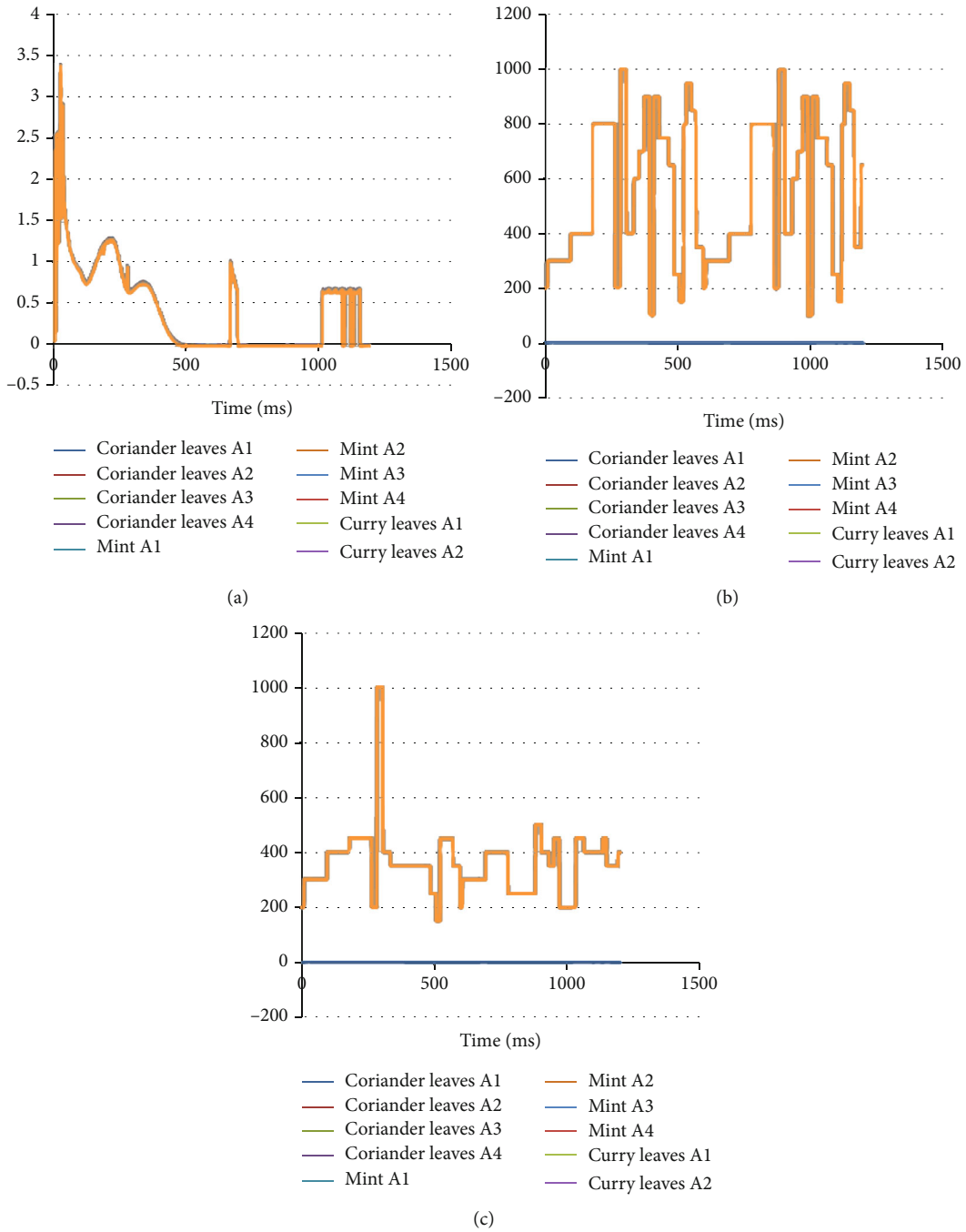


FIGURE 5: Variability chart of processing times in milliseconds.

TABLE 4: Addition of time due to morphological operation (ms).

Resolution	PC	Rpi	WM
1280 × 960	1.63	235.92	33.20
640 × 480	0.17	60.90	11.15
320 × 240	0.06	14.68	6.07
Average	0.62	103.80	16.76

TABLE 5: Comparison of processing times (ms).

Processor	PC	Rpi	WM	Average
Mint	5.63	174.57	48.03	76.10
Coriander leaves	5.50	176.84	48.02	76.82
Curry leaves	5.52	165.91	48.34	73.26



increase) for the personal computer, thus increasing the computational cost of the system operation to 80%. A paradoxical behavior is observed for small images in mathematics, with an excessively proportional increase in the cost of calculation, perhaps due to some overlap with activity.

Since it is classified as a postprocessing step, the morphological function does not depend on the submitted approach or view, and its function depends only on the size of the structural elements and the number of pixels in the image, i.e., for each resolution. Although these are found in Table 4, its duration increase is approximate.

In addition to the other observations made so far, this map shows the apparent impact of image size on the cost of computing the large number of pixels. The resolution levels are 4 times higher than the ratio of the number of pixels between them, and this ratio is noticeable in average operations, especially in the Raspberry Pi bag, but differences in processing indicate that they handle different amounts of data. The differences in these relationships can be seen in Table 6.

This table indicates the relationship of the calculated times between the processors, which will act as a measure of the time for simulation and algorithmic testing between different sites. This speed ratio depends on the resolution of the image, but on average, the calculated times on the embedded module are approximately 30 times longer than the advanced system and 3.5 times longer than the advanced language Wolfram Mathematica Computer.

It is important to highlight that despite the great influence of matter on sectoral responses, and consistency was maintained in sectoral approaches, i.e., best and worst methods were maintained despite the analyzed plant. This information can be used to recreate results for different plants and expand the potential of the results. The graphic results discussed can be found in Table 7.

In terms of morphology, its application demonstrates total error reduction in all applications, making consistent improvements to the results. However, the improvement was not as significant as expected, especially in situations where the error was already reduced by the use of approaches aimed at color separation. The mean reduction of the total error with the morphological application was 0.706%, a difference of 0.059%, which was even smaller when considering color approaches alone, indicating a proportional improvement of approximately 3%.

The resolution, in turn, reflects the most significant impact, taking into account most variations in the same approach. In color-based approaches, there is a tendency to reduce the error as the image size increases, from averaging to low resolution: 1.17%, 1.73%, and 2.19%, respectively. This difference can be explained by two effects, some changes and incorrect classifications in current noise and material definitions; so, the higher the resolution, the smaller the ratio of pixels to the change and, consequently, the lower the error. Another possibility is that the quality of the reference image deteriorates, reducing clarity increases classification errors, making the manual process for determining plant pixels more difficult, and having a greater impact on measurements due to any errors in the actual image for a small number of pixels.

## 8. Discussion

Visual systems are already a technology in precision farming, and in its various applications, the image processing problem for the plant segment is the initial and important step in obtaining valuable information from the environment, and the importance of this process is comparable to its difficulty. The study shows that the process of separating and classifying plant seedlings is complex and depends on the captured scene, which becomes a real challenge when exposed to unstable conditions of the external environment without the use of light control or more specific hardware. These restrictions are driven by functionality and market perspective, aimed at low-cost and access to technology, resulting in limitations in processing, hardware, operating practices, and consequently possible solutions.

Under these restrictions, literary analyses suggest color-based separation processes as one of the most effective in identifying plants. Therefore, common devices such as RGB cameras and modular processing units and open and widespread software were considered. On this basis, several techniques based on color separation and color stabilization methods were combined, compared, and tested under different and general lighting conditions. Despite the difficulties and precautions, the experiments showed the most promising solutions for separation, even in situations such as noise and lack of visibility.

Preliminary tests have identified key features of the measurements to support other comparative efforts. The processing time measurement was accurate and very effective in differentiating approaches, but the process showed signs of instability. It was found that display features did not affect processing times, lighting conditions, objects in the scene, or the distance of the capture system. On the other hand, the resolution directly triggers the processing, which is proportional to the number of pixels. The different processing units analyzed showed the limitations of low-cost modular hardware, but the reproduction of responses found in both processing time and segment performance allowed for the adjustment of factors and encouraged the use of high-level software for prototype and testing approaches.

In terms of division performance, the first tests helped to modify and improve the classification measurements, indicating that the *F*-score was a more accurate and accurate measure of the division than the total error used in the first attack used in conjunction with the proportional error: measurement between centers for comparative studies. Both measurements show sensitivity to related test methods and factors, and with the right increase in the number of test reviews, they set the correct response variables for the measurement method.

The rated components in the first experiments show that high-resolution images have made little progress in the category for most strategies, but their influence has not been precursor and high reproducibility has been observed between responses. The differences were said to be due to the actual loss of actual images, calculations, and ultimately the transition between object and background. Capture devices showed significant differences in performance

TABLE 6: Comparison of resolutions and processing in computational time (ms).

Resolution	Total average time			Ratio of velocity to Rpi			Time ratio as a function of the smallest image		
	PC	Rpi	WM	PC	Rpi	WM	PC	Rpi	WM
1280 × 960	13.29	387.89	107.26	29.44	1.08	3.7	15.62	15.51	12.9
640 × 480	2.73	104.38	29	39.44	1.08	3.69	3.2	4.23	3.54
320 × 240	0.93	25.22	8.44	29.66	1.08	3.09	1.08	1.08	1.08
Average	5.65	172.50	48.23	32.85	1.08	3.49			

TABLE 7: Comparison of segmentation error (%).

		Approach		Excess green	Average
		HSV	HSV modified		
Object	Mint	1.02	0.95	1.96	1.31
	Coriander leaves	0.44	0.43	1.13	0.67
	Curry leaves	2.26	2.20	4.39	2.95
Average		1.24	1.19	2.49	1.64

associated with transient approaches, where the intrinsic oscillations of the capture were related to the less separable nature of the system; otherwise, the difference was not observed. Morphological operations such as postprocessing and noise removal in binary images showed consistent improvements in results, but an increase in their computational time, especially in the embedded volume, did not compensate for the minor improvements made, and this function is only recommended.

## 9. Conclusion

In conclusion, we obtained the following results from this experiment by combining computational performance and segment answers:

- (i) The measurements were able to detect differences between the other test variables, showing great potential in the proposed measurement system; although, it has been predicted that the potential variation in the responses of the approaches to the continuous images could provide better use of the best estimates of the mean value. In addition to providing important information about the stability of the proposed solutions.
- (ii) Processors have a great influence on the calculated cost, but one can observe the regeneration by adjusting the size. Features of the scene, subject, etc. do not seem to change the timing of the answers
- (iii) Since it is proportional to the number of pixels, the resolution directly and significantly affects the processing time. However, the increase in processing, with significant improvements in segment performance, establishes a cost-benefit ratio

- (iv) The morphological function produced improvements in all responses, but its high computational cost, especially in the embedded module, encourages its use due to the small segment gains
- (v) The characteristics of seedlings affect the response of the segment and should be considered in performance analyzes
- (vi) Approaches are expected to have a major impact on extraction, with HSV location-based approaches showing the best results in this experiment; in addition, the responses of the methods show reproduction between different seedlings
- (vii) Because both the computer and the embedded module (Rpi) are closely related to the required alerts, advanced tools such as math and software can be used for prototyping and testing approaches

## Data Availability

The datasets used and/or analyzed during the current study are available from the corresponding authors on reasonable request.

## Conflicts of Interest

There is no conflict of interest.

## References

- [1] L. F. Tian and D. C. Slaughter, "Environmentally adaptive segmentation algorithm for outdoor image segmentation," *Agriculture*, vol. 21, no. 3, pp. 153–168, 1998.
- [2] M. El-Faki, N. Zhang, and D. Peterson, "Weed detection using color machine VISION," *American Society of Agricultural and Biological Engineers*, vol. 43, no. 6, pp. 1969–1978, 2000.

- [3] E. Hamuda, M. Glavin, and E. Jones, "A survey of image processing techniques for plant extraction and segmentation in the field," *Computers and Electronics in Agriculture*, vol. 125, pp. 184–199, 2016.
- [4] J. L. Hernández-Hernández, G. García-Mateos, J. M. González-Esquiva, D. Escarabajal-Henarejos, A. Ruiz-Canales, and J. M. Molina-Martínez, "Optimal color space selection method for plant/soil segmentation in agriculture," *Computers and Electronics in Agriculture*, vol. 122, pp. 124–132, 2016.
- [5] S. Ratnasingam and T. M. McGinnity, "Chromaticity space for illuminant invariant recognition," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3612–3623, 2012.
- [6] V. Agarwal, B. R. Abidi, A. Koschan, and M. A. Abidi, "An overview of color constancy algorithms," *Journal of Pattern Recognition Research*, vol. 1, no. 1, pp. 42–54, 2006.
- [7] A. Gijzenij, T. Gevers, and J. Van De Weijer, "Computational color constancy: survey and experiments," *IEEE Transactions on Image Processing*, vol. 20, no. 9, pp. 2475–2489, 2011.
- [8] S. Bianco, C. Cusano, and R. Schettini, "Color constancy using CNNs," 2015, <https://arxiv.org/abs/1504.04548>.
- [9] Z. Lou, T. Gevers, N. Hu, and M. P. Lucassen, "Color constancy by deep learning," in *Proceedings of the British Machine Vision Conference*, pp. 76.1–76.12, U.K., 2015.
- [10] S. W. Oh and S. J. Kim, "Approaching the computational color constancy as a classification problem through deep learning," *Pattern Recognition*, vol. 61, pp. 405–416, 2017.
- [11] G. E. Meyer and J. C. Neto, "Verification of color vegetation indices for automated crop imaging applications," *Computers and Electronics in Agriculture*, vol. 63, no. 2, pp. 282–293, 2008.
- [12] D. Tilman, C. Balzer, J. Hill, and B. L. Befort, "Global food demand and the sustainable intensification of agriculture," *Proceedings of the National Academy of Sciences, National Acad Sciences*, vol. 108, no. 50, pp. 20260–20264, 2011.
- [13] H. H. Choi, H. S. Kang, and B. J. Yun, "CNN-based illumination estimation with semantic information," *Applied Sciences*, vol. 10, no. 14, pp. 4806–4817, 2020.
- [14] H. Zhan, S. Shi, and Y. Huo, "Computational colour constancy based on convolutional neural networks with a cross-level architecture," *IET Image Processing*, vol. 13, no. 8, pp. 1304–1313, 2019.
- [15] M. A. Hussain, A. S. Akbari, and E. Abbott-Halpin, "Color constancy for uniform and non-uniform illuminant using image texture," *IEEE Access*, vol. 7, pp. 7294–72978, 2019.
- [16] A. Akbarinia and C. A. Parraga, "Colour constancy beyond the classical receptive field," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 9, pp. 2081–2094, 2018.
- [17] S. Bianco, C. Cusano, and R. Schettini, "Single and multiple illuminant estimation using convolutional neural networks," *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4347–4362, 2017.
- [18] Y. Hu, B. Wang, and S. Lin, "FC4: fully convolutional color constancy with confidence-weighted pooling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4085–4094, The Netherlands., 2017.
- [19] W. Shi, C. C. Loy, and X. Tang, "Deep specialized network for illuminant estimation," in *European conference on computer vision*, pp. 371–387, The Netherlands., 2016.
- [20] D. Fourure, R. Emonet, E. Fromont, D. Muselet, A. Trémeau, and C. Wolf, "Mixed pooling neural networks for color constancy," in *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3997–4001, Phoenix, AZ, USA, 2016.
- [21] X.-S. Zhang, S.-B. Gao, R.-X. Li, X.-Y. Du, C.-Y. Li, and Y.-J. Li, "A retinal mechanism inspired color constancy model," *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1219–1232, 2016.

## Research Article

# Neural Network-Based Ultra-High-Definition Video Live Streaming Optimization Algorithm

Yunning Feng,<sup>1</sup> Nan Hu ,<sup>1</sup> and Xiaosheng Yu<sup>2</sup>

<sup>1</sup>School of Electrical and Control Engineering, Shenyang Jianzhu University, Shenyang 110168, China

<sup>2</sup>School of Information Science and Engineering, Northeastern University, Shenyang 110819, China

Correspondence should be addressed to Nan Hu; [hunan@sjzu.edu.cn](mailto:hunan@sjzu.edu.cn)

Received 11 February 2022; Revised 20 March 2022; Accepted 29 March 2022; Published 23 April 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Yunning Feng 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.

Online live streaming has been widely used in distant teaching, online live shopping, and so on. Particularly, online teaching live streaming breaks the time and space boundary of teaching and has better interactivity, which is a new distant education mode. As a new online sales model, online live shopping promotes the rapid development of Internet economy. However, the quality of live video affects the user experience. This paper studies the optimization algorithm of ultra-high-definition live streaming, focusing on superresolution technology. Convolutional neural network (CNN) is a multilayer artificial neural network designed to process two-dimensional input data. It takes advantage of CNN in image processing. This paper proposes an image superresolution algorithm based on hybrid dilated convolution and Laplacian pyramid. By mixing the dilated convolution module, the receptive field of the network can be improved more effectively to obtain more context information so that the high-frequency features of the image can be extracted more effectively. Experiment was running on Set5, Set14, Urban100, and BSD100 datasets, and the results reveal that the proposed algorithm outperforms baselines with respect to peak signal to noise ratio (PSNR), structural similarity index measurement (SSIM), and image quality.

## 1. Introduction

At the end of 2019, the COVID-19 pandemic spread across the world. However, with the continuous spread of the pandemic, the application of online live streaming is becoming more and more extensive, such as online live streaming for shopping and online live streaming for teaching [1, 2]. However, the quality of live video needs to be further improved. With the continuous development of multimedia technology, online live streaming, short video, and other video applications have gradually become mainstream media for people's study, life, and entertainment due to their strong social and interactive characteristics [3]. According to relevant survey [4], video media generated 60% of Internet traffic in 2016, and it will develop further. By 2020, that figure has reached 78%. However, the video system is often limited by various objective conditions, especially in the video sending end, the video collection equipment with insufficient accuracy, limited network bandwidth, and terminal with

insufficient processing capacity make it difficult for the video system to provide adequate ultra-high-definition video sources [5–7].

In order to solve the above problems, superresolution is used in the video system so that the video application with limited objective conditions can also provide high-quality video presentation [8]. For example, wearable video devices with relatively weak CPU processing capacity are often unable to support high-resolution video and can only use low-resolution video formats. But users of these devices still want high-definition video content. Therefore, using superresolution technology to restore the video quality at the video display end can greatly improve the visual experience of users [9, 10]. In addition, video superresolution technology is also used in many fields such as medical image research, security monitoring processing, and video coding and decoding, which has very high research value. Although many video superresolution methods have been proposed, due to the characteristics of video frames and the diversity

of video scenes, their superresolution results are not completely satisfactory. Further research is needed to improve the performance of video superresolution.

Superresolution is the process of generating high-resolution images from a set of low-resolution images [11]. This process supplements the spatial pixels of the image, increases the texture details of the image, restores the high-frequency information lost in the imaging process, and makes the image exquisite in detail, natural in picture, and has a better visual effect. This process can also be seen as the reverse process from high-resolution images to multiple low-resolution images. The simplest method of video superresolution is to perform single-frame superresolution on each frame of low-resolution video directly and finally restore all the obtained high-resolution images to high-resolution video according to the sequence of video stream [12]. However, such methods do not take into account the correlation between frames in the video, and the resulting high-resolution video may have problems such as poor interframe transition and interframe flicker. When superresolution technology is used in video, not only the current low-resolution video frame to be restored but also the transition relationship between the current frame and adjacent frames should be considered [13, 14]. Based on the single-frame image superresolution technology, combined with the imaging characteristics of video sequence, video superresolution technology often uses the redundant information between adjacent frames to further improve the superresolution performance.

Convolutional neural networks (CNN) imitating biological vision mechanisms are widely used in the field of computer vision. In superresolution reconstruction, CNN takes advantage of its learning function to establish the mapping relationship between low-resolution images and high-resolution images through training [15]. Due to its unique properties, CNN not only provides good performance in feature perception but also can detect features close to human visual system observation, so it has been widely applied in the field of superresolution technology. Accordingly, the main contribution of this paper is that a hybrid dilated convolution and Laplacian pyramid-based image superresolution algorithm is proposed.

The remainder of this paper is organized as follows. Section 2 reviews related work. In Section 3, the improved image superresolution algorithm is presented. Experimental results are presented in Section 4. Section 5 concludes this paper.

## 2. Related Work

Video is a sequence of images that are projected onto the screen at a rate that gives it image continuity. In order to improve the image resolution and display effect, many image resolution enhancement algorithms have been proposed by relevant researchers. Traditional superresolution algorithms include image interpolation, image superresolution based on sparse representation, and image superresolution based on manifold learning. In [16], the authors proposed a new single image superresolution method, which obtained the

initial high-resolution image by feature constrained polynomial interpolation. In [17], a new random forest method for image superresolution feature enhancement was proposed, using the traditional gradient-based feature to enhance the image superresolution feature, and formulated different feature formulas in different stages of image superresolution processing. In [18], a new superresolution algorithm for vertically guided neonatal image was proposed. In [19], a single image recognition method combining comprehensive sparse coding and analytical sparse coding was proposed. In [20], the author proposed a sparse Bayesian estimation-based single image superresolution method to reconstruct the superresolution image. In [21], a manifold learning-based improved texture image superresolution algorithm was proposed.

Many superresolution methods based on neural network have been proposed. In [22], the authors developed a capsule attention and reconstruction neural network (CARNN) framework to incorporate the capsule into the image superresolution CNN. In [23], the authors constructed a full deconvolution neural network (FDNN) and used FDNN to solve the problem of single image superresolution. In order to improve the resolution of remote sensing images, in [24], a superresolution neural network called progressive residual depth neural network (PRDNN) was proposed. In [25], a new perceptual image superresolution method was proposed to gradually generate visually high-quality results by constructing a stage network. In [26], the authors used CNN to generate superresolution underwater images. In [27], the authors used the enhanced attention network to realize the superresolution of compressed images. In [28], an arbitrary scale superresolution method of medical image was proposed, which combined meta learning with generative adversarial networks. In [29], the authors used gradual strategy to train CNN and proposed an efficient superresolution model. In [30], a deformable residual convolution network for image superresolution was proposed to enhance the transformation modeling ability of CNN. The emergence of deep learning has greatly promoted the development of image superresolution reconstruction, motivating this paper.

## 3. Proposed Method

Inspired by the Laplacian pyramid structure, in this section, a hybrid dilated convolution and Laplacian pyramid-based image superresolution algorithm is proposed.

As shown in Figure 1, the network can be divided into  $\log_2 RT$  levels, and RT is the reconstruction times. Each level can be divided into feature extraction and image reconstruction. The feature extraction of each level is composed of a hybrid dilated convolution layer and a deconvolution layer. The reconstruction consists of a convolution layer and an element-wise summation layer [31]. The feature extraction of the first level is slightly different from that of other levels. Compared with other levels, the first level has an extra convolution layer for extracting shallow features and channel transformation because of the number of channels in the input image is different from that of each level [32, 33]. Therefore, a convolution layer with input channel 1 and output channel 64 is added at the beginning of the first level to



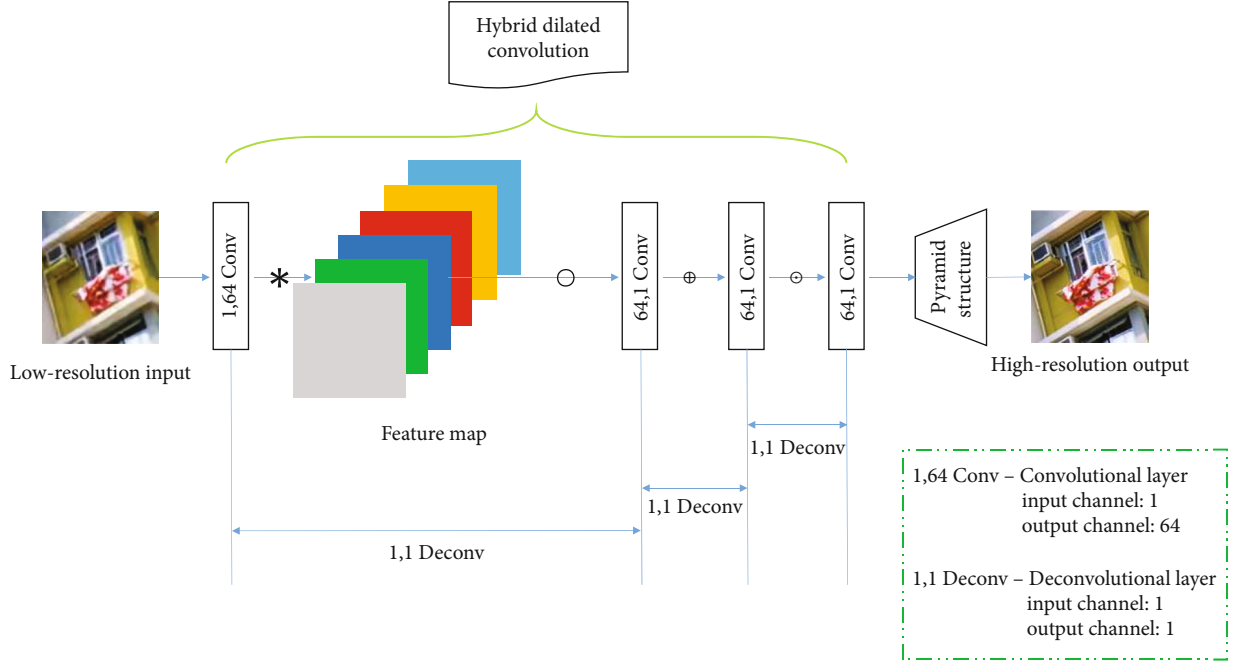


FIGURE 1: Overall network architecture of the proposed method.

extract shallow features from low-resolution images with input channel 1 and output 64 feature images. In feature extraction, high-dimensional features are extracted, and then, the extracted features are amplified by two times with a deconvolution layer, which serves as the input of feature extraction at the next level and reconstruction at this level.

The input of the image reconstruction is the output image of the reconstruction of the upper level and the output of the feature extraction of this level. The image goes through a deconvolution layer with input channel 1 and output channel 1, and the image is amplified by two times. The feature image is fused by a convolution layer with input channel 64 and output channel 1, and the original feature image with channel number 64 is transformed into a residual image with channel number 1. Finally, the amplification image and residual image are added together by element-wise summation operation to obtain the output of the reconstruction of this level and then serve as the input of the reconstruction of the next level. In this way, feature extraction and reconstruction are performed level by level. Moreover, after  $\log_2 RT$  level feature extraction, high-resolution images with the required amplification of  $RT$  can be obtained.

Different from upsampling and downsampling, this section adopts the Laplacian pyramid structure step by step for image reconstruction to better balance the accuracy and efficiency of reconstruction. Let input low-resolution image from the network be  $I^{LR}$ , and  $C_{RT}$  is used to represent the convolution layer of the first layer for shallow feature extraction.

$$C_{RT}(I^{LR}) = \max(0.3 \times (\log w_c * I^{LR} + b_c), w_c * I^{LR} + b_c), \quad (1)$$

where  $w_c$  and  $b_c$  are the weight and bias of the first-level convolution kernel,  $*$  is the convolution operation, the convolution kernel size of the first-level convolution is  $3 \times 3$ , the input channel is 1, and the output channel is 64. Among them,  $\max()$  is the activation function ReLU, and then,  $C_{RT}(I^{LR})$  is passed into the network as the input of the first-level feature extraction.

Let  $C_{HD}^k$  represent the hybrid dilated convolution module of the  $k$ th level feature extraction,  $C_k$  represents its output feature map,  $C_D^k$  represents the deconvolution layer of the  $k$ th level feature extraction, and  $C_{k,D}$  is its output feature map; then we have

$$C_1 = C_{HD}^1(C_{RT}(I^{LR})), \quad (2)$$

$$C_{k,D} = C_D^k(C_k), \quad k = 1, 2, \dots, \log_2 RT, \quad (3)$$

$$C_k = C_{HD}^k(C_{k-1,D}), \quad k = 1, 2, \dots, \log_2 RT. \quad (4)$$

As can be seen from equations (2) to (4), each level of Laplacian pyramid structure network feature extraction is made by hybrid dilated convolution module that is followed by deconvolution layers. The output of the hybrid dilated convolution module for deconvolution layer at the corresponding level of input and the output of the deconvolution layer to the next level is regarded as the input feature extraction at the next lower level hybrid dilated convolution of the input [34]. The hybrid dilated convolution module  $C_{HD}^k$  is composed of multiple hybrid dilated convolution blocks. The deconvolution layer can be defined as follows.

$$C_D^k(C_k) = \text{ReLU}(\log w_d \circ C_k + b_d), \quad (5)$$

where  $w_d$  and  $b_d$  are the weight and bias of the deconvolution layer and  $\circ$  is the deconvolution operation. The size of the convolution kernel of the deconvolution layer of the specific feature extraction is  $4 \times 4$ , and the step size is 2 (amplified twice each time). The input channel is 64, and the output channel is 64.

At the first level in reconstruction branch, the low-resolution image  $I^{LR}$  is input to the deconvolution layer of the reconstruction,  $C_{O,D}^k$  is used to represent the deconvolution layer of the reconstruction branch, and  $O_k$  represents its output; then, we have

$$O_1 = C_{O,D}^1(I^{LR}). \quad (6)$$

$C_R^k$  represents the convolution layer of the reconstruction branch, and  $O_R$  represents its output; then, we have

$$\begin{aligned} O_R &= C_R^k(C_{k,R}), \\ C_R^k(C_{k,R}) &= w_R^k * C_{k,R} + b_R^k, \end{aligned} \quad (7)$$

where  $C_{k,R}$  is the output of the feature extraction branch of this level and  $w_R^k$  and  $b_R^k$  are the weight and bias of the convolutional layer of the reconstruction branch of the  $k$ th level, respectively. While the size of the convolutional kernel of the convolutional layer is  $3 \times 3$ , the input channel is 64, and the output channel is 1. The output of the reconstructed branch at this level is defined as follows.

$$O_k^R = \text{ReLU}(O_k \oplus O_R), \quad (8)$$

where  $\oplus$  represents the element-wise summation layer.

The output expression of other level is defined as follows.

$$\begin{aligned} O_k &= C_{O,D}^k(O_{k-1}), \\ C_{O,D}^k(O_{k-1}) &= w_{O,R}^k \odot O_{k-1} + b_{O,R}^k, \end{aligned} \quad (9)$$

where  $w_{O,R}^k$  and  $b_{O,R}^k$  are the weight and bias of the deconvolution layer of the reconstruction branch of the  $k$ th level, respectively. While the size of the convolutional kernel of the convolutional layer is  $4 \times 4$ , the step size is 2, the input channel is 1, and the output channel is 1.

$$I^{SR} \left( O_{\log_2 RT}^k \right) = C_{RT} \left( I^{LR} \right) \circ C_D^k(C_k) + C_R^k(C_{k,R}) \oplus C_{O,D}^k(O_{k-1}). \quad (10)$$

Given the above, it can be concluded that the input of the reconstruction network at this level is the output of the reconstruction network at the upper level (except the first level, which is the input low-resolution image  $I^{LR}$ ) and the output of the feature extraction network at this level. The whole network is extracted and reconstructed step by step. Finally, after  $\log_2 RT$  times of extraction and reconstruction, high-resolution image  $I^{SR}$  of target multiple can be obtained, namely,  $O_{\log_2 RT}^k$ . Although the network is divided into two

parts, the whole network is trained and optimized together. Compared with the upsampling method, the proposed method can greatly reduce the time and space consumption of the algorithm.

## 4. Experiments

*4.1. Evaluation Metrics.* The evaluation index of image superresolution reconstruction algorithm can be divided into subjective evaluation index and objective evaluation index. The subjective evaluation index is mainly scored by the assessor after comprehensive consideration of all aspects of the image based on experience. This method is intuitive and can well reflect the visual quality of the reconstructed image. However, this method is affected by many factors. In order to avoid the situation of different evaluation scores for the same reconstructed quality image, we need objective evaluation indexes that can reflect subjective evaluation and not be transferred by subjective will. However, there is no objective evaluation index that can completely describe subjective evaluation, which can only reflect to a certain extent. In this paper, the two most commonly used metrics are selected for detailed introduction, respectively, peak signal to noise ratio (PSNR) and structural similarity index measurement (SSIM). Both of these two evaluation metrics use mathematical formulas to describe the similarity of two images, which are not interfered by subjective factors and are more scientific than subjective impression scoring.

PSNR evaluates the image quality by calculating the ratio of the error of the corresponding pixel point between image X to be evaluated and the standard reference image Y to the maximum pixel of the image. PSNR is intuitive, simple, and easy to understand with a small amount of calculation, but it is a pixel-by-pixel evaluation index. It is error sensitive and often differs from human visual perception because it does not take into account the visual characteristics of human eyes, such as brightness structure and other information [35]. Unlike PSNR, which only considers pixel difference between images, SSIM describes the similarity between image X to be evaluated and the standard reference image Y from three aspects: brightness, contrast, and structural information. The value range of SSIM is (0, 1]. The larger the value is, the better the quality of reconstructed image is. SSIM takes into account various factors to evaluate the quality of an image, including brightness, contrast, and structural information of the image, so as to measure the quality of reconstructed high-resolution images in a more comprehensive way and the evaluation results are more consistent with human vision.

*4.2. Datasets.* The training set used in this paper consists of 292 pictures with different resolutions, scenes, and types. However, the data amount of 292 images is not enough to support CNN learning, so data enhancement is needed for the training set. The process of enhancement is as follows. (i) The original image is image-resized using bicubic method (to save the texture, the image is only downsampling; upsampling will destroy the texture information of the image), and the scaling factor is (1, 0.9, 0.7). (ii) The scaled

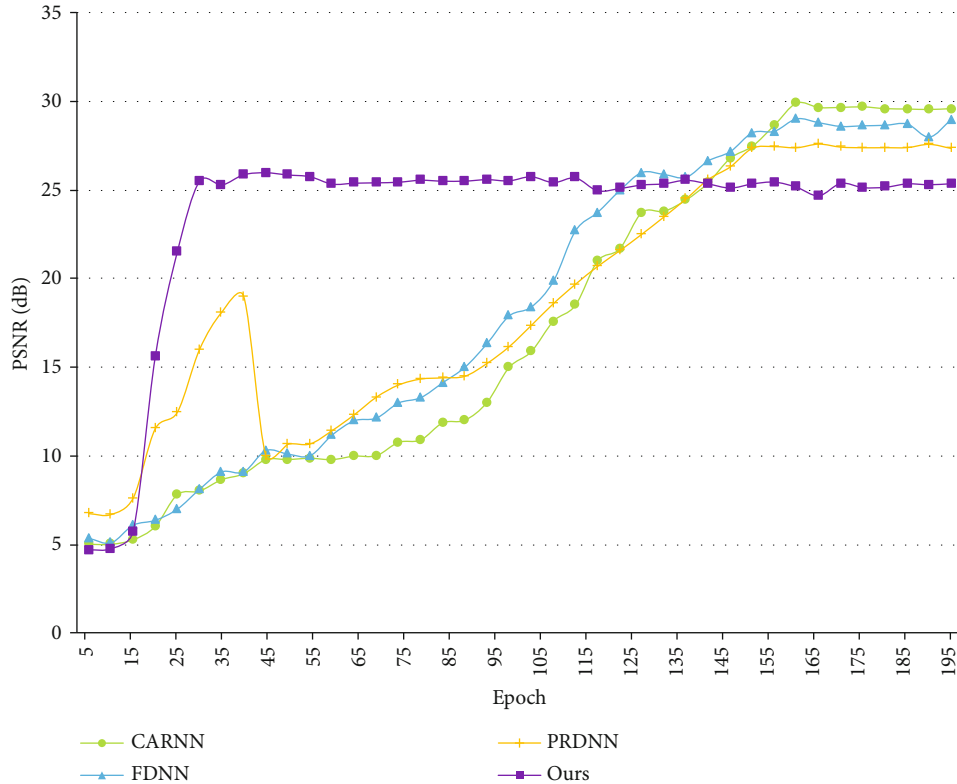


FIGURE 2: Performance comparison of PSNR.

image is conducted image-rotate with  $(0, 90, 180, 270)$  degree. (iii) Image-flip is performed on the rotated image in both horizontal and vertical directions. After image enhancement, the amount of image data (number of images) can be 48 times as much as before  $(4 \times 4 \times 3)$ . The test sets used in this paper are Set5, Set14, Urban100, and BSD100.

**4.3. Experiment Settings.** The experiment was running on an Intel i7-12700KF 3.6 GHz CPU, 32 GB of RAM (3333 MHz), and NVIDIA RTX 3080 Ti, 12 GB GDDR6X GPU.  $w_c = 0.35$ ,  $b_c = 0.92$ ,  $w_d = 0.37$ ,  $b_d = 0.91$ ,  $w_R^k = 0.41$ ,  $b_R^k = 0.89$ ,  $w_{O,R}^k = 0.42$ , and  $b_{O,R}^k = 0.90$ , where  $w$  is set randomly and  $b$  is set according to literature [36]. The batch size is set as 64, and initial learning rate of weight is  $1e-4$ . Every 200 epochs, the learning rate of weights decreases by 10 times. To verify the effectiveness of superresolution, CARNN [22], FDNN [23], and PRDNN [24] were selected for performance comparison and image reconstruction.

**4.4. Model Analysis.** Figure 2 shows that the hybrid dilated convolution and Laplacian pyramid-based image superresolution algorithm proposed in this paper was relatively high within 150 epochs at the beginning. However, after 150 epochs, the network based on Laplacian pyramid structure had a fast convergence speed and a high PSNR value, and the learning was stable and forward. The results show that the image superresolution reconstruction based on Laplacian pyramid structure network can better learn the low-resolution image and high-resolution image mapping. It is also indicated that the step-by-step upsampling method adopted in

this paper can better extract features in different resolution spaces, alleviate the learning limitations of one-layer deconvolution when the amplification is too large, better learn the mapping from low-resolution image to high-resolution image, and obtain more high-frequency information. Moreover, a high-resolution image with sharp edges and rich texture details is obtained. As can be seen from Figure 3, the SSIM of the algorithm proposed in this paper is always at a stable level and always higher than 0.9. In contrast, the other three baselines fluctuated frequently between 0.75 and 0.92, which was unfavorable to the reconstruction process of superresolution images. This also confirms the validity of the algorithm in this paper.

To more intuitively feel the quality of image reconstruction of the model, we also show the high-resolution images reconstructed by the above compared algorithm. Four images from Set5, Set14, Urban100, and BSD100 datasets were selected, and the final experimental results are shown in Figures 4–7. Set5 dataset and Set14 dataset are low-complexity single-image superresolution datasets based on nonnegative neighborhood embedding, and the training set is used for single-image superresolution reconstruction, that is, to reconstruct high-resolution images from low-resolution images to obtain more details. Urban100 dataset has rich texture images and is generally used as a dataset for network testing, while BSD100 is a classical image dataset having 100 test images.

From Figures 4–7, it can be seen that the high-resolution image reconstructed by the algorithm proposed in this paper is more similar to the real high-resolution image, and the

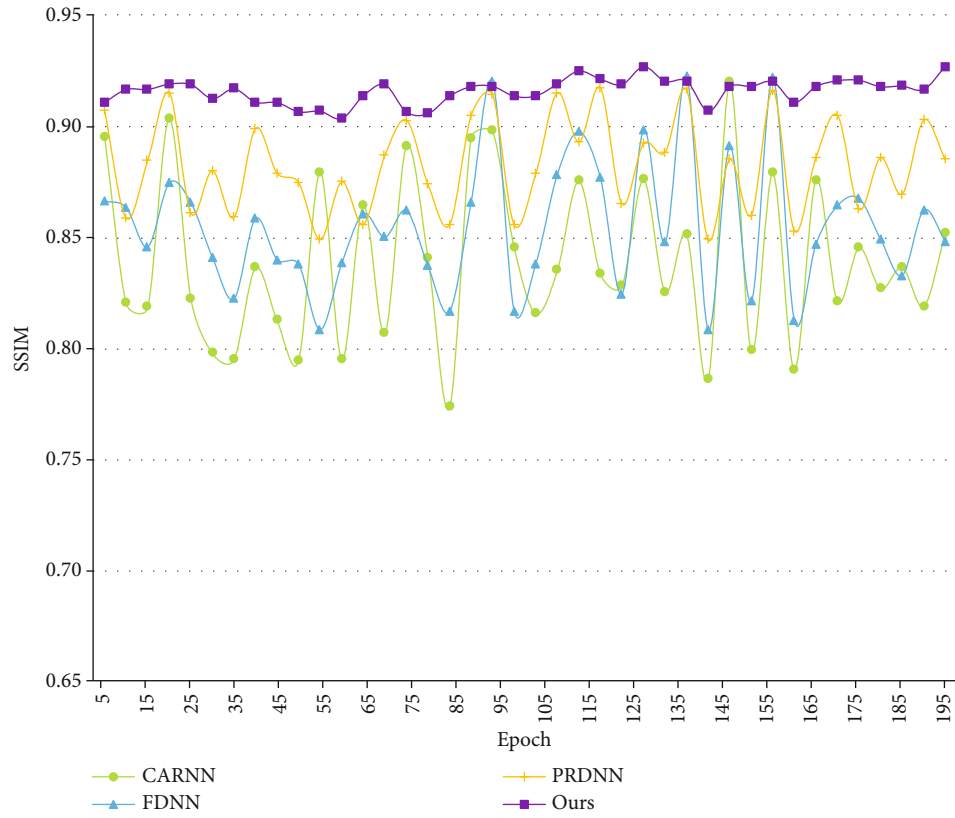


FIGURE 3: Performance comparison of SSIM.



FIGURE 4: Visual comparison of different models of Set5.



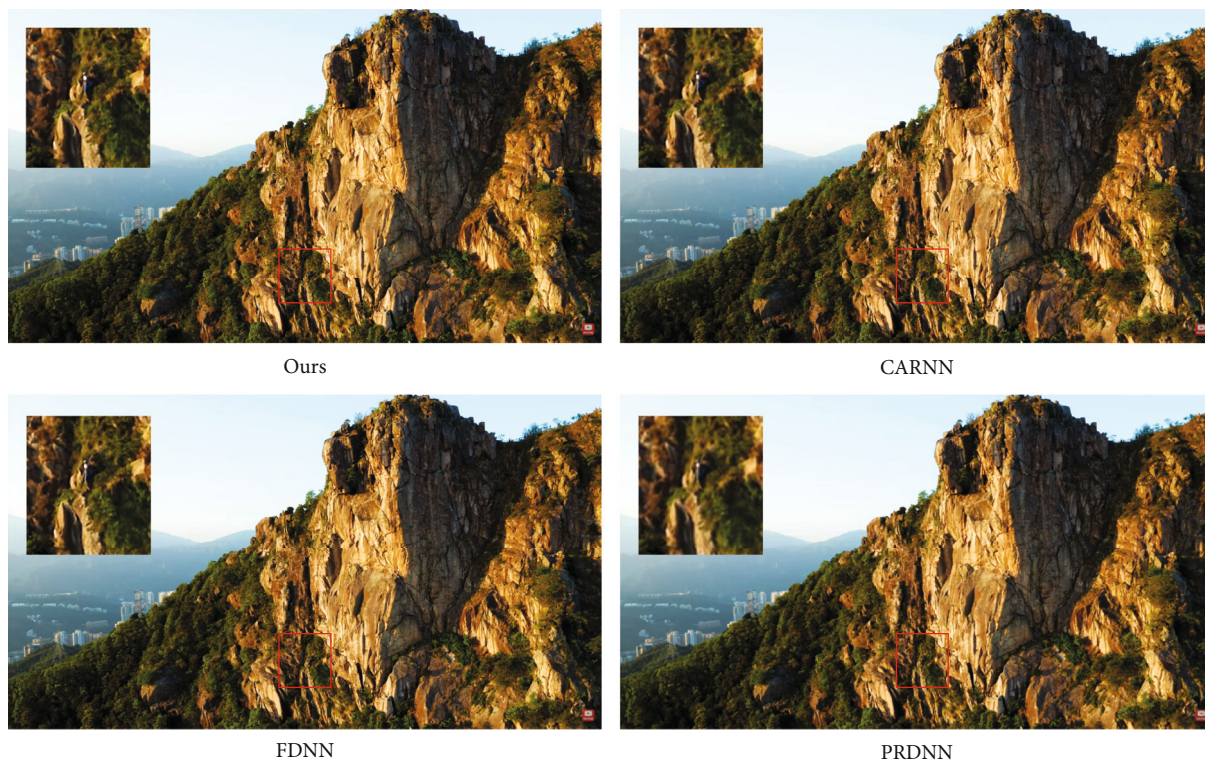


FIGURE 5: Visual comparison of different models of Set14.

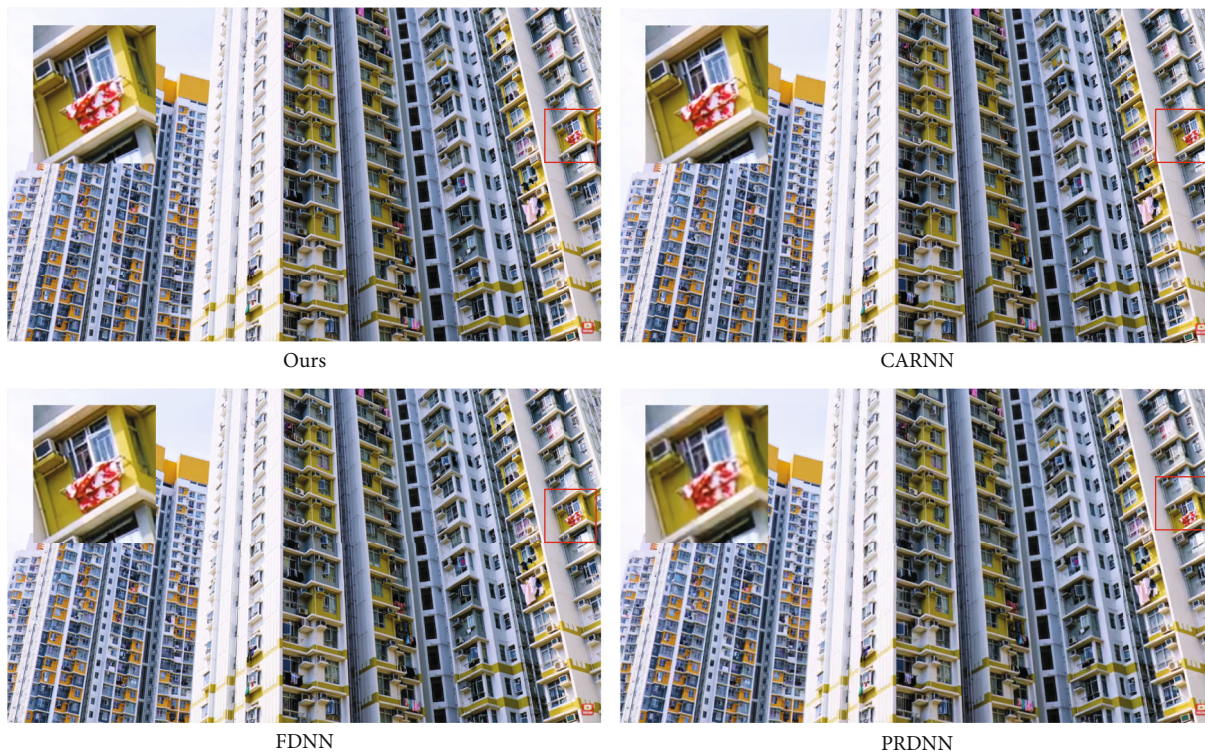


FIGURE 6: Visual comparison of different models of Urban100.



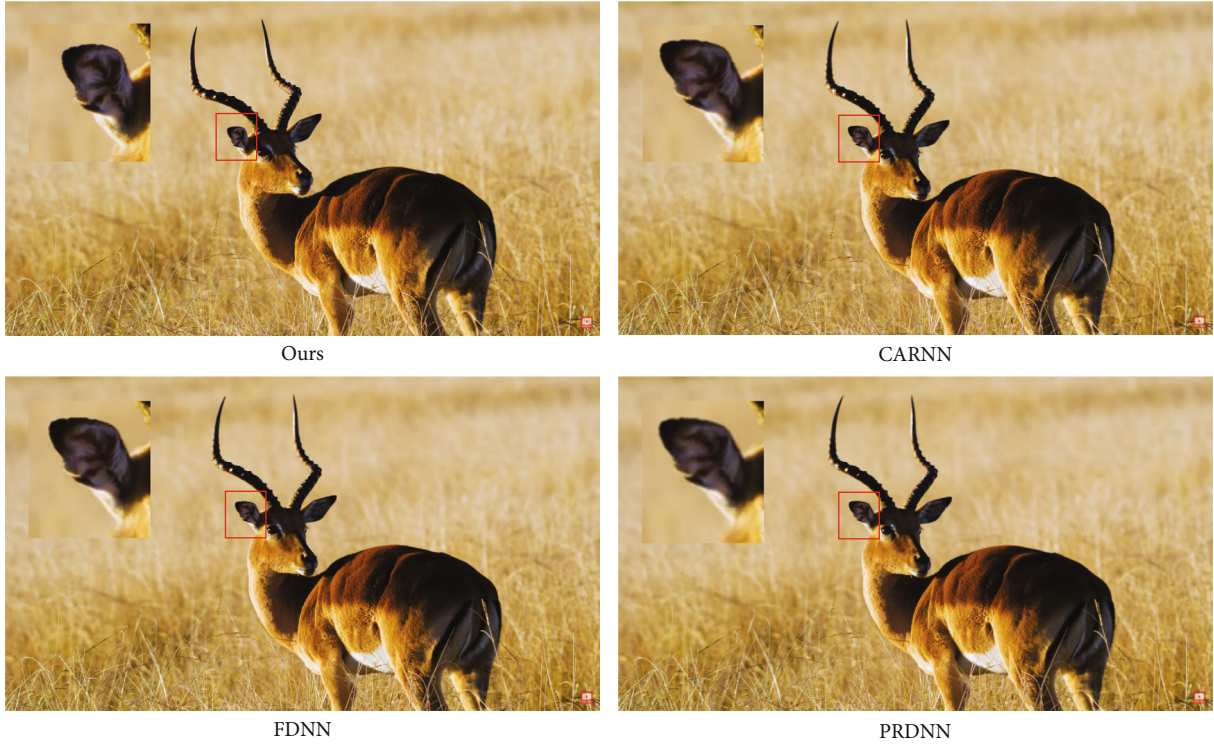


FIGURE 7: Visual comparison of different models of BSD100.

real image texture is reconstructed. Although there are flaws in performance, compared with other baselines, the performance of the algorithm proposed in this paper is the best. It can also be seen from Figure 4 that, compared with other baselines, the algorithm proposed in this paper has reconstructed the edge contour of the font better and is closer to the real high-resolution image. The experimental results show that the image superresolution algorithm based on hybrid dilated convolution and Laplacian pyramid proposed in this chapter can extract the medium- and high-frequency features of images more effectively, reduce the attenuation phenomenon when features are transmitted in the network, and reconstruct high-resolution images that are more consistent with human visual perception.

## 5. Conclusion

In this paper, an image superresolution algorithm based on hybrid dilated convolution and Laplacian pyramid structure is proposed. The hybrid dilated convolution module is used to extract image features, which can better expand the receptive field of the model without introducing additional parameters. In this way, the model can obtain more context information and improve the feature extraction ability of the model and will not cause grid effect. To alleviate the problems of insufficient learning ability of the upsampling layer when the reconstruction multiple is large and the large amount of calculation and long reconstruction time of the front upsampling model, the step-by-step upsampling method based on Laplacian pyramid is used to gradually

amplify the image, which balances the relationship between reconstruction time and reconstruction quality and allows the model to learn more high-frequency information. Experiments show that the proposed algorithm effectively improves the image quality.

This paper proposes a solution to the shortcomings of the existing CNN-based image superresolution reconstruction algorithm and verifies the effectiveness of the proposed method through experiments. However, there are still some problems to be solved in this field. The follow-up work will be carried out from the following aspects in the future.

- (i) For standard low-resolution and high-resolution image datasets, the existing image superresolution algorithm acquires low-resolution images by down-sampling high-resolution images from public datasets to obtain corresponding low-resolution images. However, low-resolution images acquired in this way cannot completely simulate the image degradation process, and acquired low-resolution images have similar styles. Therefore, in the future, efforts will be made to establish a complete set of low-resolution and high-resolution image data to make up for this shortcoming in the field of image superresolution reconstruction
- (ii) The existing image based on CNN superresolution algorithm is to rebuild a multiple single algorithm. When the reconstruction multiple is different, a new model has to be retrained, which is extremely inflexible. Moreover, the existing algorithms can only

reconstruct high-resolution images with integer multiple and cannot achieve arbitrary amplification. The future will be how to realize the reconstruction of the flexible network

- (iii) Natural images are rich in prior information, while the existing CNN-based image superresolution ignores the prior information. In the future, we will focus on the exploration of prior information and make full use of the prior information of images to reconstruct high-resolution images with richer high-frequency information

## Data Availability

All data used to support the findings of the study is included within this paper.

## Conflicts of Interest

The authors declare no conflicts of interest in this paper.

## Acknowledgments

This work was supported by Youth Program Research Projects of Liaoning Higher Education Institutions (Grant No. lnqn202014).

## References

- [1] T. Chen, L. Peng, J. Yang, G. Cong, and G. Li, "Evolutionary game of multi-subjects in live streaming and governance strategies based on social preference theory during the COVID-19 pandemic," *Mathematics*, vol. 9, no. 21, p. 2743, 2021.
- [2] Y. Zhao and F. Bacao, "How does gender moderate customer intention of shopping via live-streaming apps during the COVID-19 pandemic lockdown period?," *International Journal of Environmental Research and Public Health*, vol. 18, no. 24, 2021.
- [3] M. Zhang, F. Qin, G. Wang, and C. Luo, "The impact of live video streaming on online purchase intention," *Service Industries Journal*, vol. 40, no. 9-10, pp. 656-681, 2020.
- [4] M. Martini, "Online distant witnessing and live-streaming activism: emerging differences in the activation of networked publics," *New Media & Society*, vol. 20, no. 11, pp. 4035-4055, 2018.
- [5] V. M. Baskaran, Y. C. Chang, J. Loo, and K. Wong, "Design and implementation of parallel video combiner architecture for multi-user video conferencing at ultra-high definition resolution," *Multimedia Tools and Applications*, vol. 74, no. 17, pp. 6589-6622, 2015.
- [6] H. Kim, S. Ahn, W. Kim, and S. Lee, "Visual preference assessment on ultra-high-definition images," *IEEE Transactions on Broadcasting*, vol. 62, no. 4, pp. 757-769, 2016.
- [7] F. Zhu, Y. Ning, X. Chen, Y. Zhao, and Y. Gang, "On removing potential redundant constraints for SVOR learning," *Applied Soft Computing*, vol. 102, 2021.
- [8] C. Zhang, Y. Niu, T. Wu, and X. Li, "Color image super-resolution and enhancement with inter-channel details at trivial cost," *Journal of Computer Science and Technology*, vol. 35, no. 4, pp. 889-899, 2020.
- [9] S. Pang, Z. Chen, and F. Yin, "Lightweight multi-scale aggregated residual attention networks for image super-resolution," *Multimedia Tools and Applications*, vol. 81, 2022.
- [10] P. Liu, Y. Hong, and Y. Liu, "Deep differential convolutional network for single image super-resolution," *IEEE Access*, vol. 7, pp. 37555-37564, 2019.
- [11] X. Jiang, N. Wang, J. Xin, X. Yang, Y. Yu, and X. Gao, "Image super-resolution via multi-view information fusion networks," *Neurocomputing*, vol. 402, pp. 29-37, 2020.
- [12] C. Liu, X. Sun, C. Chen et al., "Multi-scale residual hierarchical dense networks for single image super-resolution," *IEEE Access*, vol. 7, pp. 60572-60583, 2019.
- [13] F. Hao, T. Zhang, L. Zhao, and Y. Tang, "Efficient residual attention network for single image super-resolution," *Applied Intelligence*, vol. 52, no. 1, pp. 652-661, 2022.
- [14] F. Zhu, J. Gao, J. Yang, and N. Ye, "Neighborhood linear discriminant analysis," *Pattern Recognition*, vol. 123, 2022.
- [15] P. V. Arun, K. M. Buddhiraju, A. Porwal, and J. Chanussot, "CNN-based super-resolution of hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 9, pp. 6106-6121, 2020.
- [16] J. Liu, Y. Liu, H. Wu, J. Wang, X. Li, and C. Zhang, "Single image super-resolution using feature adaptive learning and global structure sparsity," *Signal Processing*, vol. 188, 2021.
- [17] H. Li, K. M. Lam, and M. Wang, "Image super-resolution via feature-augmented random forest," *Signal Processing: Image Communication*, vol. 72, pp. 25-34, 2019.
- [18] Y. Zhang, F. Shi, J. Cheng, L. Wang, P. Yap, and D. Shen, "Longitudinally guided super-resolution of neonatal brain magnetic resonance images," *IEEE Transactions on Cybernetics*, vol. 49, no. 2, pp. 662-674, 2019.
- [19] X. Li, G. Cao, Y. Zhang, A. Shafique, and P. Fu, "Combining synthesis sparse with analysis sparse for single image super-resolution," *Signal Processing-Image Communication*, vol. 83, p. 115805, 2020.
- [20] Y. Yang, "Research on the single image super-resolution method based on sparse Bayesian estimation," *Cluster Computing-the Journal of Networks Software Tools and Applications*, vol. 22, no. S1, pp. 1505-1513, 2019.
- [21] D. Mishra, B. Majhi, S. Bakshi, A. K. Sangaiah, and P. K. Sa, "Single image super resolution for texture images through neighbor embedding," *Multimedia Tools and Applications*, vol. 79, no. 13-14, pp. 8337-8366, 2020.
- [22] J. T. Hsu, C. H. Kuo, and D. W. Chen, "Image super-resolution using capsule neural networks," *IEEE Access*, vol. 8, pp. 9751-9759, 2020.
- [23] F. Cao, K. Yao, and J. Liang, "Deconvolutional neural network for image super-resolution," *Neural Networks*, vol. 132, pp. 394-404, 2020.
- [24] J. Zhang, S. Liu, Y. Peng, and J. Li, "Satellite image super-resolution based on progressive residual deep neural network," *Journal of Applied Remote Sensing*, vol. 14, no. 3, 2020.
- [25] Z. Hui, J. Li, X. Gao, and X. Wang, "Progressive perception-oriented network for single image super-resolution," *Information Sciences*, vol. 546, pp. 769-786, 2021.
- [26] T. Yang, S. Jia, and H. Ma, "Research on the application of super resolution reconstruction algorithm for underwater image," *CMC-Computers Materials & Continua*, vol. 63, no. 3, pp. 1249-1258, 2020.

- [27] X. Wang, Z. Wang, X. He, C. Ren, and P. Karn, "Super-resolution of compressed images using enhanced attention network," *Journal of Electronic Imaging*, vol. 30, no. 3, 2021.
- [28] J. Zhu, C. Tan, J. Yang, G. Yang, and P. Lio, "Arbitrary scale super-resolution for medical images," *International Journal of Neural Systems*, vol. 31, no. 10, 2021.
- [29] M. Zareapoor, P. Shamsolmoali, and J. Yang, "Learning depth super-resolution by using multi-scale convolutional neural network," *Journal of Intelligent & Fuzzy Systems*, vol. 36, no. 2, pp. 1773–1783, 2019.
- [30] Y. Zhang, Y. Sun, and S. Liu, "Deformable and residual convolutional network for image super-resolution," *Applied Intelligence*, vol. 52, no. 1, pp. 295–304, 2022.
- [31] Z. Zhang, X. Wang, and C. Jung, "DCSR: dilated convolutions for single image super-resolution," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1625–1635, 2019.
- [32] J. Lv, X. Wang, K. Ren, M. Huang, and K. Li, "ACO-inspired information-centric networking routing mechanism," *Computer Networks*, vol. 126, pp. 200–217, 2017.
- [33] X. Lin, J. Wu, S. Mumtaz, S. Garg, J. Li, and M. Guizani, "Blockchain-based on-demand computing resource trading in IoV-assisted smart city," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1373–1385, 2022.
- [34] L. Ma, M. Huang, S. Yang, R. Wang, and X. Wang, "An adaptive localized decision variable analysis approach to large-scale multiobjective and many-objective optimization," *IEEE Transactions on Cybernetics*, vol. 52, pp. 1–13, 2021.
- [35] H. Zheng, Z. Shi, C. Zhou, M. Haardt, and J. Chen, "Coupled coarray tensor CPD for DOA estimation with coprime L-shaped array," *IEEE Signal Processing Letters*, vol. 28, pp. 1545–1549, 2021.
- [36] W. Sun, X. Zhang, and X. He, "Lightweight image classifier using dilated and depthwise separable convolutions," *Journal of Cloud Computing-Advances Systems and Applications*, vol. 9, no. 1, 2020.

## Research Article

# Anomaly Detection and Restoration for AIS Raw Data

Shuguang Chen <sup>1</sup>, Yikun Huang <sup>2</sup>, and Wei Lu<sup>3</sup>

<sup>1</sup>Department School of Management, Xi'an University of Finance and Economics, Xi'an 710100, China

<sup>2</sup>Concord University College of Fujian Normal University, Fuzhou 350117, China

<sup>3</sup>School of Information, Xi'an University of Finance and Economics, Xi'an 710100, China

Correspondence should be addressed to Yikun Huang; [fjnuhyk@163.com](mailto:fjnuhyk@163.com)

Received 18 December 2021; Accepted 19 February 2022; Published 30 March 2022

Academic Editor: Pei-Wei Tsai

Copyright © 2022 Shuguang Chen 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 wide application of location detection sensors in maritime surveillance, a large amount of raw automatic identification system (AIS) data is produced by many moving ships. Anomaly detection and restoration of the big AIS data are important issues in marine data mining, because they offer a reliable support to users to mining the behaviors of ships. This paper develops a novel approach to detect anomaly AIS data based on the ships' maneuverability, such as the maximum acceleration, the minimum acceleration, the maximum distance, and the maximum angular displacement, which were designed to detect the anomaly AIS data. Furthermore, the performance of the developed approach is compared with that of Daiyong-Zhang's method and Behrouz-Haji-Soleimani's method to assess its detection efficiency. The results show that the proposed approach can be applied to easily extract the abnormal data. Finally, based on the developed approach to detect the anomaly data and cubic spline interpolation method to restore the AIS data, experiments are conducted on the AIS data of Xiamen Port of Fujian Province, China, that prove to be effective for marine intelligence research.

## 1. Introduction

With the rapid development and maturity of positioning technology, communication technology, and network technology, various types of mobile intelligent terminals with positioning and navigation functions are becoming more and more widely used, and the location of mobile objects (including people, vehicles, ships, and animals) relevant information is increasingly accessible and can be collected on a large scale. This type of location data usually contains information such as geographic coordinates, speed, direction, and time, and it continues to increase and update rapidly over time [1]. It is called trajectory big data. Given that trajectory big data records the movement of moving objects over time and can objectively reflect the activities of individuals or groups of moving objects, as well as their impact on the environment, it has led to the concern of scholars in various fields such as natural sciences, social sciences, and environmental science [2–4]. With the rapid increase on international trade, an increasing number of vessels have

been come into service; as a result, the safety and security of marine transportation have become the most dominating attention of marine surveillance. Since 2002, the International Maritime Organization (IMO) requires automatic identification system (AIS) transponders to be aboard vessels that are above 300 gross tonnages on international voyages, cargo ships over 500 gross tonnages in all waters, and all the passenger ships regardless of size [5]. The AIS tracks vessel movement by means of electronic exchange of navigation data between vessels, with onboard transceiver, terrestrial, and satellites. This navigation data is related to the ship itself (including its static parameters and dynamic activity records, such as ship name, ship maritime mobile service identity, ship size, ship type, speed, location, time, heading, and rate of turn) and includes other features such as the sea state. For all reasons mentioned above, a massive amount of AIS data is produced. The use of this massive AIS data is an important part of intelligent marine transportation system and conducts research on ship collision avoidance [6–8], ship behavior analysis [9, 10], ship emission analysis,



trajectory analysis [7, 10–21], maritime surveillance [10, 11, 20, 22–24], accident investigation [8, 25], etc. Anomaly detection and restoration [26, 27]–[20, 23, 24, 28] are the fundamental key research problems in the marine intelligent transport system, which aims to identify and restore the abnormal data in the AIS data generated by the users through multiple aboard transceivers.

The identification and restoration of anomaly AIS data play a vital role in the intelligent analyses of AIS data, because User Datagram Protocol (UDP) is adopted for the AIS data packet transmission, during which packet-disordering and data packet dropouts occur. Another related reason deals with the quality of the raw AIS data, e.g., error and anomaly, and it is well known that the raw AIS data may be tampered to inform false types of movements, such as fishing activity in protected areas. Consequently, the error and lost AIS data will interfere with maritime management due to the misjudgment of the maritime state. Besides, it will decrease the effectiveness of analysis on ship behavior and traffic flow based on the AIS data.

Recently, many studies have focused on using techniques to detect and restore the anomaly AIS data based on an optimal trajectory calculated from classification algorithms, which designed a ranking score inventory considering the difference between the optimal trajectory and the real ones, but lots of existing works ignored the impact of ship speed or just considered the impact of ship's locations on the optimal trajectory [28]. Besides, with regard to the data veracity of ship's movement features, some works [29] focused on the threshold of ship movement features based on the ship's navigation data, such as ship speed, ship location over time, and ship heading. To deal with the anomaly detection of AIS data, the current methods proposed in the literature can mainly be classified into the following two categories as follows:

- (i) One is to design a near-optimal path to evaluate the matching between the real trajectory and the near-optimal path, such as Behrouz-Haji-Soleimani's graph search algorithm [28], classification methods, or clustering algorithm [7, 14, 20, 30]. However, these methods need to predict the main route used by ships
- (ii) Another one is to design the rules to detect the unreasonable track points based on ships' design specifications [23, 24, 29]. Nevertheless, both the unreasonable drift track points with slow speed and the unreasonable acceleration points under certain limit can hardly be detected

In this work, Daiyong-Zhang's method is generalized and extended its unreasonable acceleration from the maximum acceleration to the minimum acceleration. It must be mentioned that this paper aims to compensate the detection of drift track point when its average speed does not exceed the maximum speed, while [29] only considers to detect the drift track point when its average speed exceeds the maximum speed. In order to avoid a misjudgment of the drift track point, the proposed mode uses speed integral to obtain the maximum drift distance in moving. Besides, the detec-

tion of unreasonable turn point is simplified from the detection rate of turn in [29] to the measurement for the angular displacement of turn. Considering characteristic of the packet dropouts and data cleaning, the cubic spline interpolation method is employed to compensate the consecutive of the trajectories, which aims to minimize the accelerate vector along longitude, latitude, and velocity in the AIS data. Case studies over the AIS data sets are carried out to verify the effectiveness of the proposed method.

In conclusion, the main contributions of this paper are summarized as follows:

- (i) This paper proposes a model to detect the anomaly drift track point, which builds a maximum distance and a minimum distance between the drift point and its adjacent track points
- (ii) The minimum acceleration of the ship is modeled by using the design specifications of ships, such as the distance for a ship to decelerate from the design speed to zero. Moreover, the maximum acceleration of the ship is investigated for the detection of unreasonable acceleration
- (iii) An efficient and effective model which is just only based on the difference of heading between the drift point and its adjacent track points is proposed to detect the unreasonable track point of turn
- (iv) The 156-AIS data with a cruise of length 110 m and at interval of 10 s in December 22, 2018, from Xiamen International Cruise Center is employed to verify the effectiveness of the proposed detection models. The simulation results demonstrate that the number of anomaly AIS data by using our method is less than Daiyong-Zhang's method, but our proposed method is superior than Daiyong-Zhang's method when anomaly drift track point has slow speed. Besides, the simulation results also present that Behrouz-Haji-Soleimani's method can also be effective for discriminating the anomaly state of the objected trajectory, but it needs lots of trajectories to build the optimal trajectory
- (v) For a case study of a passenger ship (call sign: 6285, ship name: MIN LONG YU 9 777, MMSI: 412596777, ship's length is 19 m, and transmission time interval is 10 s) in December 22, 2018–January 3, 2019, from Xiamen Port, experiments show that a new smooth and reasonable trajectory is reconstructed based on the combination method of our proposed detection approach with the cubic spline restore algorithm

The remainder of this paper is organized as follows. The detection modeling of anomaly activity, such as stop state, acceleration, drift track point, and turn, is presented in Section 2. Section 3 depicts cubic spline interpolation method for data restoration. In Section 4, case studies are conducted and experimental results are shown. Section 5 concludes the paper.



## 2. Classification and Determination of Outliers in Ship AIS Trajectory Data

**2.1. AIS Trajectory Representation.** Trajectory is an important type of spatiotemporal data. It is used to represent the history and continuous state information of a moving object changing with time. It can also be considered as a time-to-state mapping. In other words, given a time  $t (t \in R^+)$ , the state space of moving target at time  $t$  can be obtained by using a continuous function  $F$  of time  $t$ . For a  $d$ -dimensional state space vector, the mapping can be expressed as  $F : R^+ \rightarrow S^d$ . A trajectory  $T_r$  of a ship is a finite set  $T_r = \{(s_1, t_1), (s_2, t_2), \dots, (s_n, t_n)\}$ , where  $s_i \in [1, n]$  is a state at a trajectory point and  $t_i \in [1, n]$  is a timestamp at a trajectory point. Yet, traditional studies on state space of ship trajectory only consider position information of the targets. In order to identify the anomaly AIS data accurately, longitude, latitude, speed, and heading are taken into account when ship's state space is studied in this paper.

**2.2. Classification and Judgment of Abnormal Points of Ship Trajectory.** By analyzing the state space of the raw AIS data sets provided by the VTEExplorer website, such as changes in longitude, latitude, speed, and heading, rules to identify the inaccurate AIS data according to the ship's maneuverability are shown in the following subsection.

**2.2.1. Abnormal Stop.** According to the message format specified by the International Telecommunication Union communication standard, AIS transponders may accept a duplicate message in the packet forwarding mechanism. In another word, two adjacent AIS trajectory points are almost the same with each other except their timestamps, i.e.,  $(s_i = s_j, t_i \neq t_j, i \neq j, i, j \in [1, n])$ . If these duplicate or abnormal messages are not processed, the false judgement of ship's running states as its stop states will occur. With regard to the detection of these abnormal messages, the determination method of the abnormal stopping point can be given as follows. For the AIS sequence, if the speed of the  $i$ -th point is greater than 2 knots (1 knot = 1 nautical mile/hour), but its geographic coordinates ( $\text{lon}_i$  and  $\text{lat}_i$ ), speed  $v_i$ , and heading  $\text{cog}_i$  are the same as that of the  $i+1$ -th point, then the  $i+1$ -th point is judged as an abnormal stopping point. The judgement rules are shown in the following:

$$\begin{cases} v_i > 2, \\ |\text{lon}_{i+1} - \text{lon}_i| = 0, \\ |\text{lat}_{i+1} - \text{lat}_i| = 0, \\ |v_{i+1} - v_i| = 0, \\ |\text{cog}_{i+1} - \text{cog}_i| = 0. \end{cases} \quad (1)$$

**2.2.2. Abnormal Acceleration Point.** According to the current design standards for ships, the distance between the stationary and the design speed trajectory points for a ship in full load condition is about 20 times of the length of the ship, while the distance between the stationary and the design

speed trajectory points for a ship in no load condition is reduced to 1/2~2/3 times of the original distance. The stopping stroke of a ship, which is affected by its displacement, is generally 8-20 times of the length of the ship. As is shown in equation (2), to obtain the maximum and the minimum acceleration, the minimum distance equals to 10 times and 8 times of the length of the ship, respectively. Assuming that the length of the ship is  $L$ , the design speed is  $V_d$ , the maximum acceleration is  $a_{\max}$ , the minimum acceleration is  $a_{\min}$ , the time interval from the stationary to the design speed  $V_d$  with a uniformly accelerating is  $t_{i \max}$ , and the time interval from the design speed  $V_d$  with a uniformly decelerating to the stationary is  $t_{i \min}$ .

$$\begin{cases} V_d = a_{\max} \times t_{i \max} = -a_{\min} \times t_{i \min}, \\ 10L = \frac{1}{2} a_{\max} t_{i \max}^2, \\ 8L = \frac{1}{2} a_{\min} t_{i \min}^2. \end{cases} \quad (2)$$

Based on equation (2), the maximum acceleration  $a_{\max}$  and the minimum acceleration  $a_{\min}$  of the ship can be obtained as follows.

$$\begin{cases} a_{\max} = \frac{V_d^2}{2 \times 10L}, \\ a_{\min} = -\frac{V_d^2}{2 \times 8L}. \end{cases} \quad (3)$$

After the speed and time difference between the  $i$ -th point and the  $i+1$ -th point, transient acceleration at time  $i+1$ -th can be derived by (4). If the calculated transient acceleration is greater than the maximum acceleration or less than the minimum acceleration, then the  $i+1$ -th point is an abnormal acceleration point.

$$\begin{cases} a = \frac{v_{i+1} - v_i}{t_{i+1} - t_i}, \\ (a_{\max} - a)(a - a_{\min}) < 0. \end{cases} \quad (4)$$

**2.2.3. Anomalous Drift Point.** Theoretically, the travelling reachability distance between two trajectory points can be obtained by the integral calculation with the speed change on the route. For a trajectory data sequence, if the distance between two adjacent AIS trajectory points exceeds their maximum reachability distance, the trajectory point is judged as an abnormal drift point. Assume that the timestamp and speed for the  $i$ -th trajectory point is  $t_i$  and  $v_i$ , respectively. At the same time,  $t_{i+1}$  and  $v_{i+1}$  represent the timestamp and speed for the  $i+1$ -th trajectory point, respectively. In theory, the travelling reachability distance between the  $i$ -th and  $i+1$ -th trajectory points should be  $\int_{t_i}^{t_{i+1}} v dt$ . But in a fact that the velocity variation pattern cannot be obtained through the measurement, the maximum reachability distance between the two trajectory points is

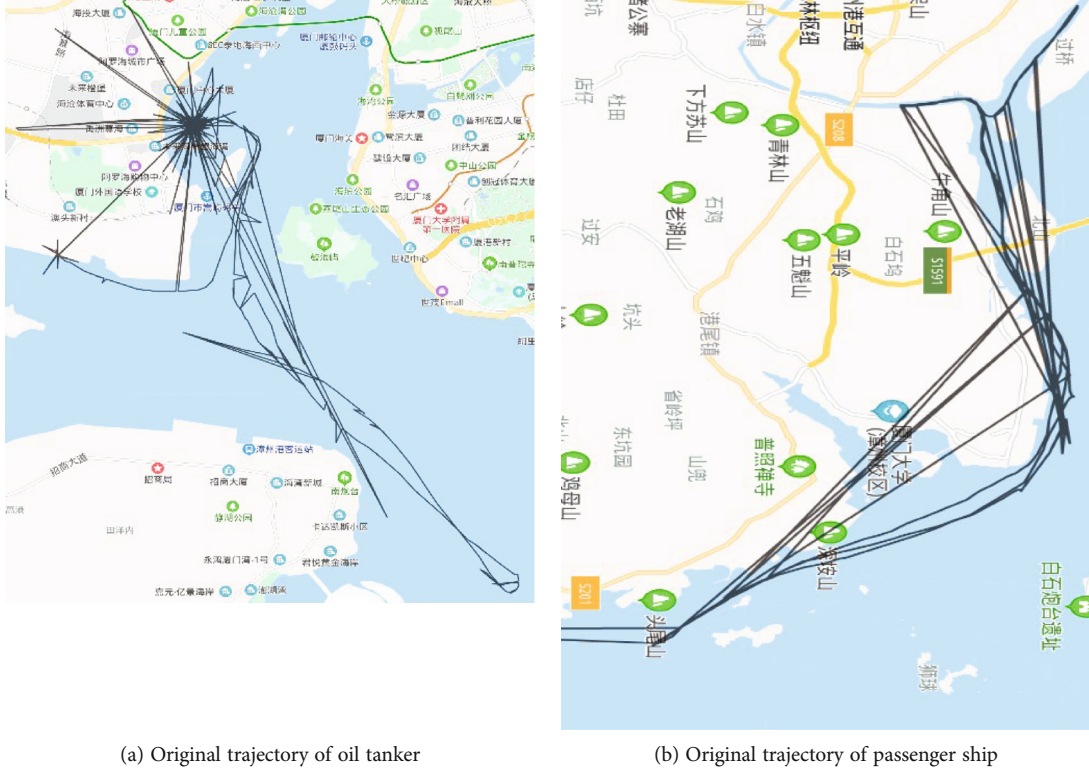


FIGURE 1: Original trajectory diagram.

estimated.

$$s_{\max} = \int_{t_i}^{t_m} (v_i + a_{\max}(t - t_i))dt + \int_{t_m}^{t_{i+1}} (v_{i+1} + a_{\min}(t_{i+1} - t))dt, \quad (5)$$

where  $t_m$  satisfies the condition of  $v_{i+1} = v_i + a_{\max}(t_m - t_i) + a_{\min}(t_{i+1} - t_m)$ , ship is in the uniformly accelerative motion with maximum accelerate  $a_{\max}$  in time interval  $[t_i, t_m)$ , and ship is in the uniformly decelerative motion with minimum accelerate  $a_{\min}$  in time interval  $[t_m, t_{i+1})$ .

After the integral calculation with the speed change on the route, the maximum reachability distance of the  $i$ -th trajectory point and the  $i + 1$ -th trajectory point can be derived by (5). As is shown in (6), if the calculated spherical distance  $dis(i, i + 1)$  between two adjacent AIS trajectory points is greater than the maximum reachability distance, then the  $i + 1$ -th point is an abnormal drift point.

$$dis(i, i + 1) > \int_{t_i}^{t_m} (v_i + a_{\max}(t - t_i))dt + \int_{t_m}^{t_{i+1}} (v_{i+1} + a_{\min}(t_{i+1} - t))dt. \quad (6)$$

**2.2.4. Anomalous Turning Point.** As is well known, the swing diameter is an important parameter for evaluating the steering capability of a ship. Generally speaking, the maximum swing diameter  $d$  of a ship can be obtained by the formula  $d = k \times l$ ,  $k \in [2, 4]$  based on the design specification of ships, in which  $l$  is the length of ship,  $k$  is the coefficient to mea-

TABLE 1: Trajectory outlier distribution.

Trajectory point	Ship type	Quantity
Normal trajectory point	Tanker	7800
	Ship	503
Anomalous drift point	Tanker	4035
	Ship	67
Abnormal stop point	Tanker	0
	Ship	0
Abnormal acceleration point	Tanker	12
	Ship	0
Abnormal turning point	Tanker	3118
	Ship	0

sure the ship maneuverability, and usually the value range of  $k$  is  $[2, 4]$ . If the speed of the ship is  $v$  and the maximum swing diameter is  $d$ , then the maximum rate of turn is  $r = 360v/\pi kl$ ; as shown in (7), the maximum angle of turn  $\omega_{\max}$  from  $i$ -th trajectory point to the  $i + 1$ -th trajectory point can be obtained through the integral calculation with the maximum rate of turn.

$$\omega_{\max} = \int_{t_i}^{t_{i+1}} \frac{360v}{\pi kl} dt \leq \frac{360}{\pi kl} s_{\max}. \quad (7)$$

In equation (7),  $s_{\max}$  is the maximum reachability distance between the two trajectory points.

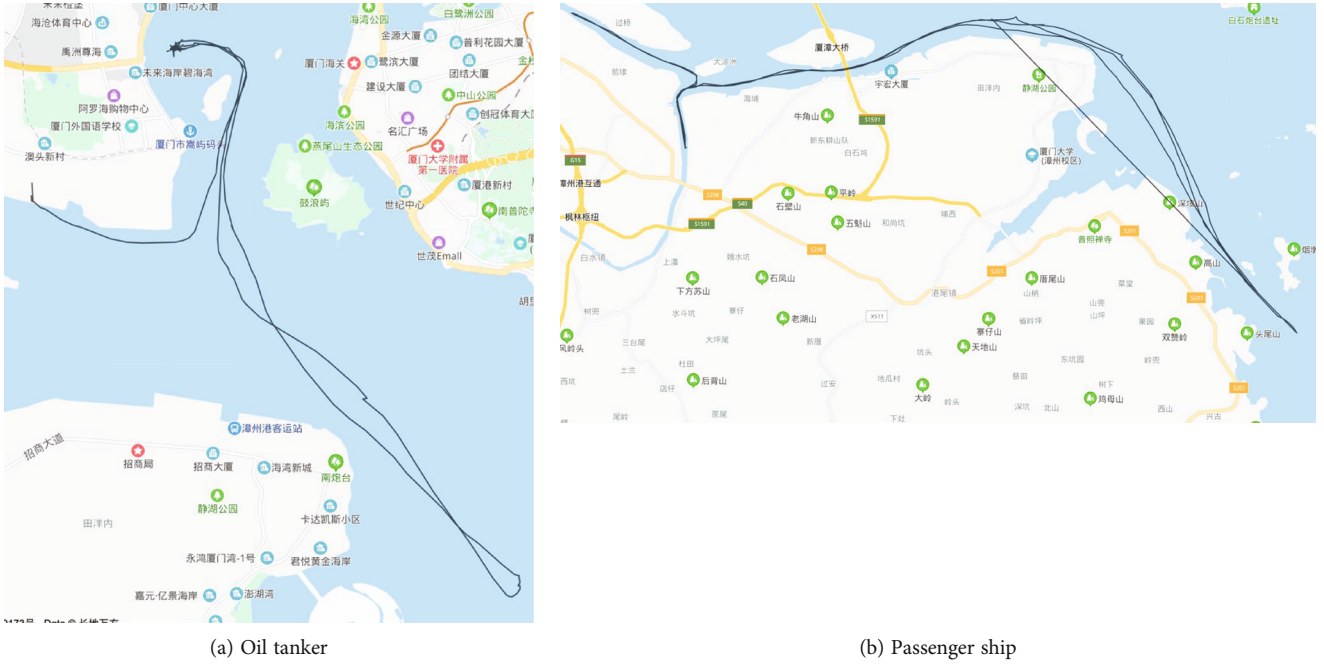


FIGURE 2: The trajectory after clearing the abnormal point.

Note that an abnormal turning point can be easily identified by using the heading difference  $|\text{cog}_{i+1} - \text{cog}_i|$  between two adjacent AIS trajectory points, as shown in the following.

$$|\text{cog}_{i+1} - \text{cog}_i| > \omega_{\max} = \int_{t_i}^{t_{i+1}} \frac{360v}{\pi kl} dt \leq \frac{360}{\pi kl} S_{\max}. \quad (8)$$

### 3. Ship AIS Trajectory Data Repair

The ship trajectory established based on AIS data is a sequence of points including discrete space-time information. In order to satisfy the subsequent research and application based on trajectory, it is necessary to delete the abnormal points in the raw data. However, the deletion of the abnormal points will cause the trajectory sequence to become discontinuous; the loss of AIS messages can also lead to this situation. Therefore, synchronous interpolation is needed to obtain continuous trajectories in practical applications. Cubic spline interpolation is one of the commonly used methods for spatiotemporal trajectory interpolation and synchronization. In the case of spatiotemporal data with few missing or intermittent missing, the cubic spline interpolation method has good repair and synchronization effects.

According to the International Telecommunication Union communication standards, the time interval for sending navigation and location-related messages is related to the type of ship and its speed. For class A ships, the interval for sending AIS messages should be no more than 10 seconds; for class B ships, the interval should be no more than 30 seconds. In the case of normal navigation, the trajectory data that can be obtained is relatively dense. The number of discontinuity trajectory points caused by message loss and out-

lier deletion is relatively small. Therefore, the cubic spline interpolation method is suitable for trajectory point repair and synchronization. But when the ship is berthing, the AIS message sending time interval will become 3 minutes. Interpolation is not necessary in this case. In addition, the ship may also shut down its AIS radio station on its own initiative. This will lead to a long segment of missing trajectory points. In this case, the ship behavior is uncertain, so it is not suitable for interpolation too.

According to the AIS message characteristics mentioned above, we segment the trajectory data after deleting outliers. Based on the time interval and speed of the points, the trajectory is divided into normal navigation section, stop section, and closed radio section. The rules to divide segments are as follows: (1) AIS messages in which interval time is less than 3 minutes are segments of normal navigation; (2) AIS messages in which interval time is greater than or equal to 3 minutes and less than or equal to 5 minutes and speed is less than 1 knot are segments of stop; (3) AIS messages in which interval time is greater than 5 minutes are segments of turning off AIS radio. In the process of trajectory data repairing, we will only use cubic spline interpolation for the first case. For the second and third cases, interpolation is not carried out. In the first case, the interpolation is carried out according to different ship types and speeds, and the specified time interval of AIS message is used as the step size.

Suppose there are  $m$  trajectory points in the AIS spatiotemporal sequence and the corresponding times of the  $m$  points are  $t_i (i = 1, 2, \dots, m)$ ,  $x_i$ ,  $y_i$ ,  $v_i$ , and  $\theta_i$  that represent the longitude, latitude, speed, and heading angle of point  $i$ , then the velocity of point  $i$  in the longitude direction is  $v_{xi} = v_i \cos \theta_i$ , and the velocity of point  $i$  in the latitude direction is  $v_{yi} = v_i \sin \theta_i$ . Setting the time starting point of the

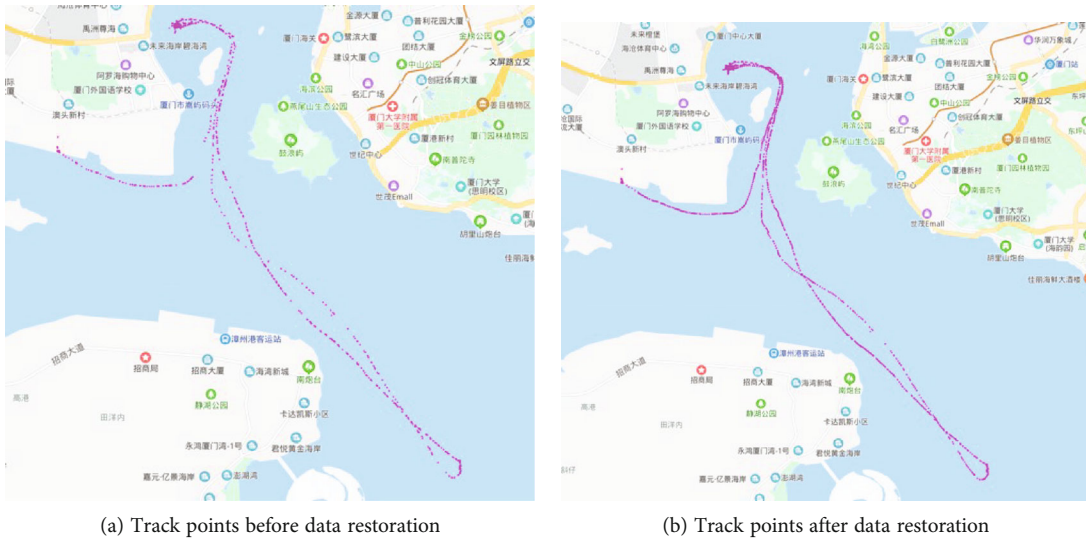


FIGURE 3: Track trajectory diagram of oil tanker.

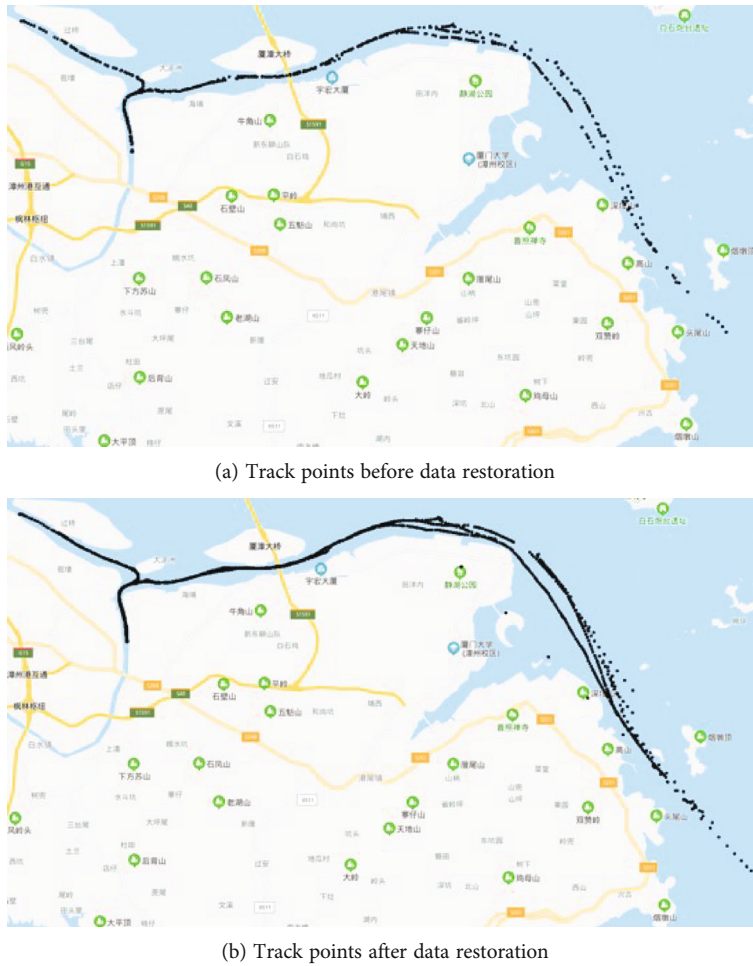


FIGURE 4: Track trajectory diagram of passenger ship.

sequence to be interpolated as zero time, and using the corresponding latitude and longitude coordinates as the origin of the coordinates, for the space-time sequence to be inter-

polated, the derivatives at the endpoints of the latitude and longitude directions are  $v_{xi}$  and  $v_{yi}$ . The coefficient matrix can be obtained by substituting  $v_{xi}$  and  $v_{yi}$  into the cubic



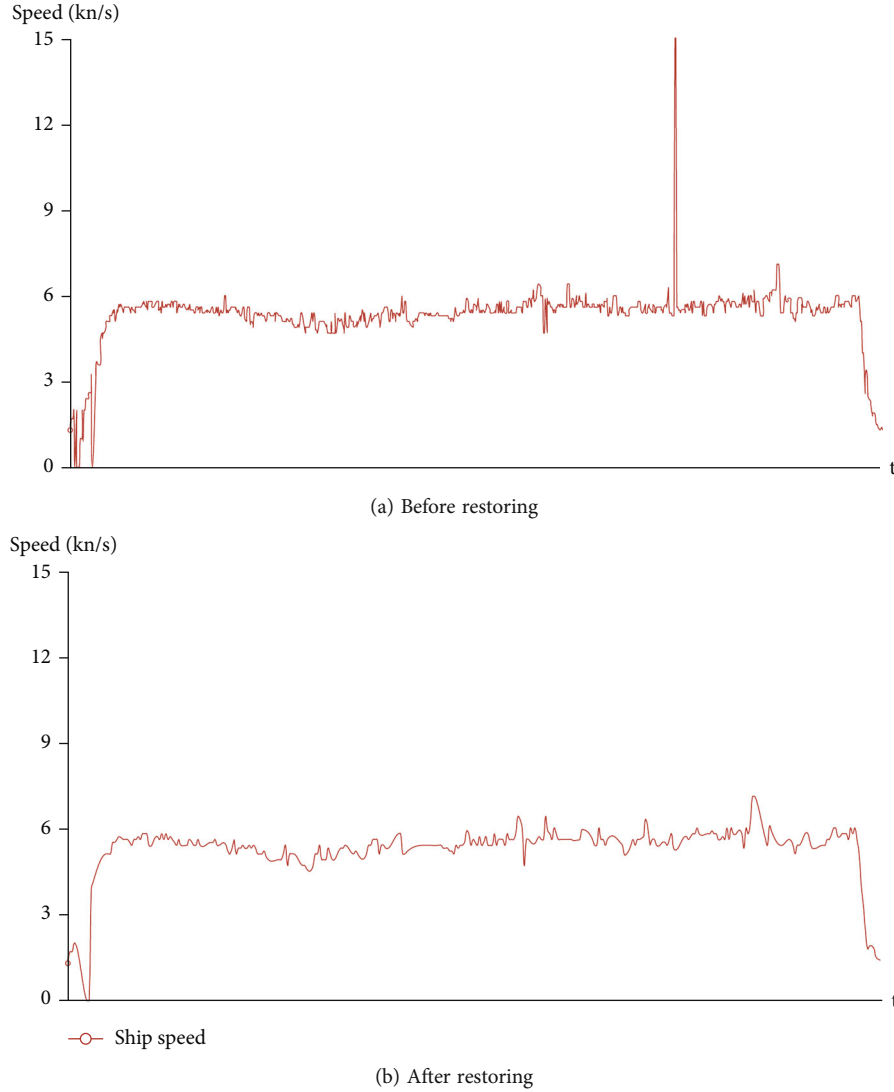


FIGURE 5: Comparison of speed changes.

spline function for each segment. Then, the coordinates of longitude and latitude corresponding to any time  $t$  can be obtained by cubic spline function in each segment.

Take longitude calculation of ships as an example. Let the longitude coordinate of ship  $y(t)$  be a function of time, and satisfy the cubic spline function  $y(t) = a\Delta t^3 + b\Delta t^2 + c\Delta t + d$  in the time period  $[t_i, t_j]$ ; then,  $v_y = dy(t)/dt = 3a\Delta t^2 + 2b\Delta t + c$ , and  $a_y = dv_y(t)/dt = 6a\Delta t + 2b$ .

According to the boundary conditions, for the specific time  $t_1$  and  $t_2$ , it should be satisfied:

$$\begin{cases} y(t_1) = at_1^3 + bt_1^2 + ct_1 + d, \\ y(t_2) = a(t_2 - t_1)^3 + b(t_2 - t_1)^2 + c(t_2 - t_1) + d, \\ v_y(t_1) = 3at_1^2 + 2bt_1 + c, \\ v_y(t_2) = 3a(t_2 - t_1)^2 + 2b(t_2 - t_1) + c. \end{cases} \quad (9)$$

According to the boundary values  $y(t_1), y(t_2), v(t_1), v(t_2)$

of the sequence to be interpolated, the spline function coefficients can be obtained. Therefore, we can obtain the function expression of longitude coordinate  $y(t)$  with respect to time  $t$ . According to the interpolation time interval, we can further obtain the longitude coordinates of the corresponding trajectory points using the function  $y(t)$ . The interpolation methods of other parameters of the trajectory sequence (latitude, speed, and heading) are the same and will not be described again.

#### 4. Experiment Analysis

In order to verify the effect of the method for processing and repairing the abnormal points of ship trajectory data proposed in this paper, some original AIS data were downloaded through the VTE Explorer website for experiments. We select the part of data about Xiamen Port and surrounding waters. The time range is from 11:46:37 on December 21, 2018, to 7:30:22 on January 3, 2019; the spatial range is  $117.7737^\circ$  E and  $24.08784^\circ$  N to  $118.63037^\circ$  E and



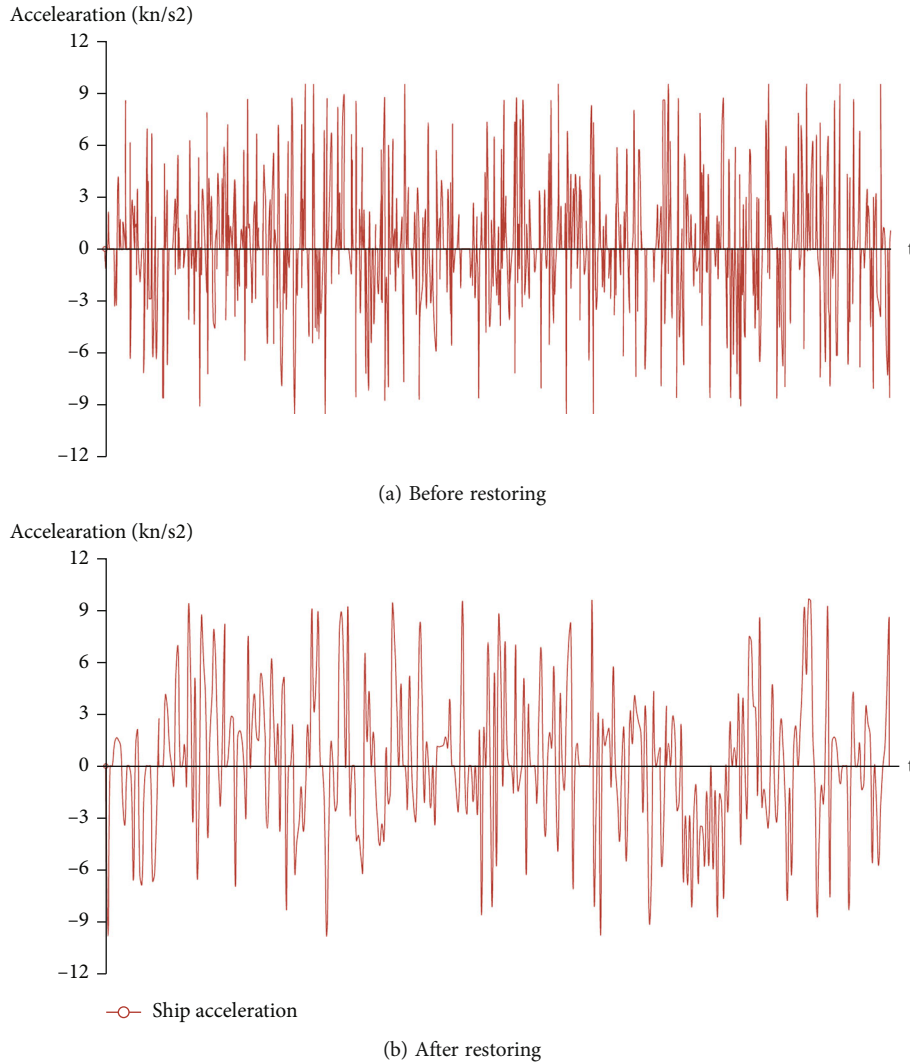


FIGURE 6: Comparison of acceleration changes.

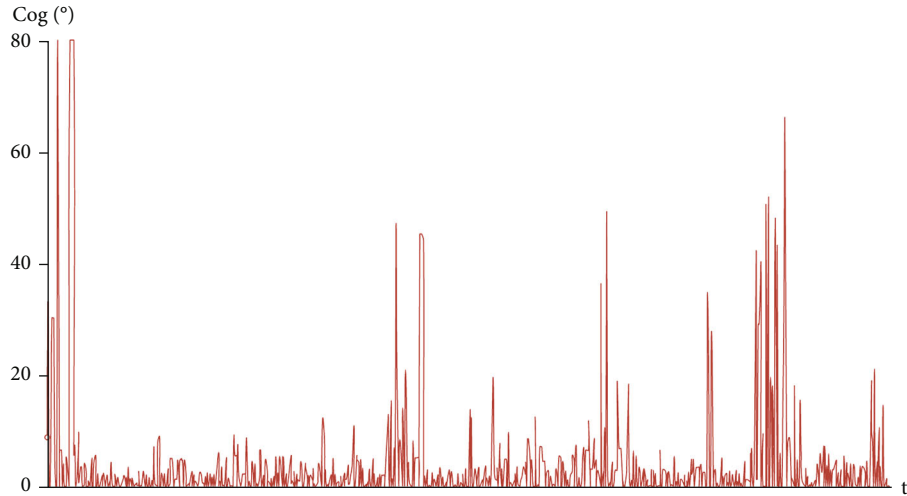
24.691° N, including 12158622 pieces of position data and 387745 pieces of static data. The experiment selects the trajectory data of an oil tanker and a passenger ship for analysis, processing, and comparison. The MMSI number of the oil tanker is 413698470, the call sign is BVHW8, the name is HAI GONG 167, the length of the ship is 32 m, and the time span of the track point is from December 21, 2018, hours 46 minutes 41 seconds to December 22, 2019 23:59 minutes 50 seconds, a total of 14,965 position data; passenger ship MMSI number is 412596777, call sign is 6285, the name is MIN LONG YU 9 777, the length of the ship is 19 m, the track point time span is from 12:52:57 on December 22, 2018, to 3:15:32 on January 3, 2019, and there are 570 pieces of location data.

**4.1. Outlier Identification and Elimination.** Before processing of the AIS original position data, the trajectories of the two ships are shown in Figures 1(a) and 1(b) within the set time span.

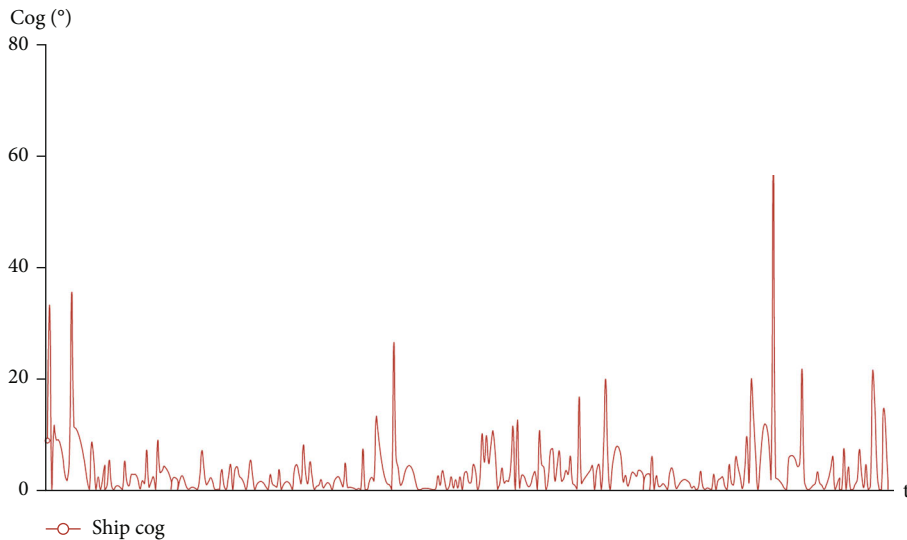
As can be seen from Figure 1, the trajectories established based on the original position data sequence of the two ships are somewhat messy, and some abnormal drift points can be

seen intuitively. The existence of anomalous drift points makes some trajectory segments even cross land, which is obviously not credible. If these ship's original position data are used as the data source of statistical analysis system, it will cause the statistical analysis results to deviate from the actual situation. For the ship management, monitoring, and analysis system, it will cause erroneous alarms due to abnormal ship behavior frequently. Using the discriminate method for abnormal trajectory points proposed in this paper, we can find that the different types of abnormal points and their number in the original position data of the two ships are shown in Table 1.

It can be seen from the distribution of the abnormal point types of the two ships in Table 1 that the abnormal points of oil tankers account for a relatively large amount, exceeding 50%, while the abnormal points of passenger ships account for a relatively small amount, around 10%; anomalous drift points account for the highest proportion of all types of abnormal points. In addition to the abnormal drift point, the oil tanker also has some abnormal turning points and a few abnormal acceleration points, while the passenger



(a) Before restoring



(b) After restoring

FIGURE 7: Comparison of heading changes.

ship has not found any other abnormal points; no abnormal stopping point has been found in two ships' trajectories. The main reason for the large difference in the distribution of abnormal points between two different types of ships may be that the passenger ship we selected has a short voyage period and basically sailing along the coast, so AIS data transmission is relatively standardized and normal; the voyage period of the oil tanker is relative long and includes the process of entering and leaving the port, which results in abnormalities in AIS data transmission and reception. In addition, there is no abnormal stopping point for both ships. The main reason may be that there is no repeated message forwarding.

According to the abnormal point processing method we proposed in this paper, we can remove all kinds of abnormal points found in the original AIS data. After removing all the abnormal points, the trajectories of the two ships are shown in Figures 2(a) and 2(b). It can be seen from Figure 2 that after clearing the abnormal points, the trajectories of both

ships became clear and identifiable. But for the passenger ship trajectory, there is still a trajectory line across the land. By querying AIS data, we can find that the timestamps of the two points are 13:27:59 on December 22, 2018, and 13:32:19 on December 22, 2018. The time span of the two points does not exceed the normal range, but the ship speed is faster and the track point span is somewhat large, so the track line passes through the land. This problem will not exist after the subsequent track point repairing that we will discuss in the next part.

*4.2. Trajectory Missing Point Repair.* The overall trajectory of the two ships becomes clear after identifying and deleting the abnormal trajectory points, but the time interval between the trajectory points is somewhat large and uneven, which sometimes cannot meet the requirements of local trajectory analysis and application. It can be seen from the static data that these two ships belong to class B vessels. Therefore, we can repair the trajectories using the cubic spline

interpolation method given in Section 2 of this paper and set the AIS message transmission interval to 30 seconds for interpolation. For the oil tanker, a total of 5,419 trajectory points were inserted, and the ratio of the inserted trajectory points to the total trajectory points is 36.21%. The comparison of the scatter points of the ship trajectory before and after data restoration is shown in Figures 3(a) and 3(b). For passenger ship, a total of 654 trajectory points are inserted, and the ratio of the inserted trajectory points to the total trajectory points is 57.93%. The comparison of scatter points of ship trajectory before and after data restoration is shown in Figures 4(a) and 4(b).

As can be seen from Figures 3 and 4, the density of the ship's trajectory points increased significantly after the data was repaired, so the trajectory became more continuous. However, some segments in trajectories are still in discontinuous state after repairing. As shown in the rectangular marking part of Figures 3(b) and 4(b), there is an obvious gap in each trajectory. The main reason why the trajectory segments have not been repaired is that the time interval between two adjacent track points in the segment is too long, more than 5 minutes for this case. It indicates that the ship is in the state of shutting down AIS equipment during this period, and its behavior is uncertain, so it will not be repaired.

In addition to restoring the abnormal points of position data, we also restore the abnormal points of data such as speed anomalies, acceleration anomalies, and heading anomalies. In the two ships we selected, only the oil tanker has acceleration and heading abnormal points, so only AIS data of the oil tanker was processed for speed, acceleration, and heading anomalies. Before and after clearing and repairing the abnormal points, the comparison of the ship's speed, acceleration, and heading changes can be illustrated by a part of trajectory data, as shown in Figures 5–7 (data range is from 2018-12-22 8:22:51 to 2018-12-22 9:37:45).

As can be seen from Figures 5–7, the abnormal changes of speed, acceleration, and direction beyond the range of ship's maneuverability in the original data have been eliminated. The ship's speed, acceleration, and direction changes tend to be continuous, smooth, and all within a reasonable range after restoration.

## 5. Conclusion

With the application and popularization of AIS equipment on ship, AIS data has become one of the important data sources for ship traffic flow analysis, maritime supervision, and accident analysis. However, it is difficult for upper-layer applications to apply these AIS data directly because of its unreliability. Aiming at the problem of ship trajectory construction based on AIS data, a method of abnormal point detection and repair in AIS data is proposed in this paper. The proposed method classifies AIS abnormal point and processes them separately according to the longitude and latitude, speed, acceleration, and direction information in AIS data. It is worth noting that the proposed method only needs the AIS data of the ship itself and does not need the support of the historical track data. In addition, the cubic

spline interpolation method is used to repair the trajectory after eliminating the abnormal points, which further improves the continuity and integrity of the trajectory.

The results of processing actual ship trajectories show that the method proposed in this paper can identify all kinds of trajectory abnormal points in AIS data effectively. The interpolation processing method after removing abnormal points can effectively eliminate the sudden changes in position, speed, acceleration, and heading. The trajectory data after being restored are in a reasonable range in terms of latitude and longitude, speed, acceleration, and heading, and the changes are continuous and smooth.

## Data Availability

In order to verify the effect of the method for processing and repairing the abnormal points of ship trajectory data proposed in this paper, some original AIS data were downloaded through the VTE Explorer website for experiments. We select the part of data about Xiamen Port and surrounding waters. The time range is from 11:46:37 on December 21, 2018, to 7:30:22 on January 3, 2019; the spatial range is 117.7737° E and 24.08784° N to 118.63037° E and 24.691° N, including 12158622 pieces of position data and 387745 pieces of static data.

## Conflicts of Interest

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

## Acknowledgments

This work was supported in part by the Fujian Province young and middle-aged teachers education research project (Nos. JAT210651 and JAT210647), the Scientific Research Project and Research Innovation Team of Concord University College of Fujian Normal University in 2020 (Nos. KY20200203 and 2020-TD-001).

## References

- [1] X. Xue and C. Jiang, "Matching sensor ontologies with multi-context similarity measure and parallel compact differential evolution algorithm," *IEEE Sensors Journal*, vol. 21, no. 21, pp. 24570–24578, 2021.
- [2] X. Xue, P.-W. Tsai, and Y. Zhuang, "Matching biomedical ontologies through adaptive multi-modal multi-objective evolutionary algorithm," *Biology*, vol. 10, no. 12, pp. 1216–1287, 2021.
- [3] X. Xue and J. Zhang, "Matching large-scale biomedical ontologies with central concept based partitioning algorithm and adaptive compact evolutionary algorithm," *Applied Soft Computing*, vol. 106, p. 107343, 2021.
- [4] X. Xue and J. Chen, "Matching biomedical ontologies through compact differential evolution algorithm with compact adaption schemes on control parameters," *Neurocomputing*, vol. 458, pp. 526–534, 2021.
- [5] R. Connolly, "Safety of Life at Sea," *Radiouser*, vol. 7, no. 4, 2012.

- [6] P. Chen, G. Shi, S. Liu, and Y. Zhang, "Decision support based on artificial fish swarm for ship collision avoidance from AIS data," in *2018 International Conference on Machine Learning and Cybernetics (ICMLC)*, Chengdu, China, 2018.
- [7] B. Murray and L. P. Perera, "A dual linear autoencoder approach for vessel trajectory prediction using historical AIS data," *Ocean Engineering*, vol. 209, p. 107478, 2020.
- [8] L. Du, F. Goerlandt, and P. Kujala, "Review and analysis of methods for assessing maritime waterway risk based on non-accident critical events detected from AIS data," *Reliability Engineering and System Safety*, vol. 200, pp. 1–23, 2020.
- [9] J. L. Shepperson, N. T. Hintzen, C. L. Szostek, E. Bell, L. G. Murray, and M. J. Kaiser, "A comparison of VMS and AIS data: the effect of data coverage and vessel position recording frequency on estimates of fishing footprints," *ICES Journal of Marine Science*, vol. 75, no. 3, pp. 988–998, 2018.
- [10] F. Wang, Y. Lei, Z. Liu, X. Wang, S. Ji, and A. K. Tung, "Fast and parameter-light rare behavior detection in maritime trajectories," *Information Processing and Management*, vol. 57, pp. 1–10, 2020.
- [11] Q. Lu and K. D. Kim, "Autonomous and connected intersection cross-sing traffic management using discrete-time occupancies trajectory," *Applied Intelligence*, vol. 49, pp. 1621–1635, 2019.
- [12] Y. Gong, E. Chen, and X. Zhang, "Antmapper: an ant colony-based map matching approach for trajectory-based applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 390–401, 2018.
- [13] E. Tu, G. Zhang, L. Rachmawati, E. Rajabally, and G. B. Huang, "Exploiting AIS data for intelligent maritime navigation: a comprehensive survey from data to methodology," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1559–1582, 2018.
- [14] G. Yuan, P. Sun, J. Zhao, D. Li, and C. Wang, "A review of moving object trajectory clustering algorithms," *Artificial Intelligence Review*, vol. 47, no. 1, pp. 123–144, 2017.
- [15] C. Anagnostopoulos and S. Hadjiefthymiades, "Intelligent trajectory classification for improved movement prediction," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 10, pp. 1301–1314, 2014.
- [16] S. Qiao, N. Han, W. Zhu, and L. A. Gutierrez, "TraPlan: an effective three-in-one trajectory-prediction model in transportation networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 3, pp. 1188–1198, 2015.
- [17] M. Schreier, V. Willert, and J. Adamy, "An integrated approach to maneuver-based trajectory prediction and criticality assessment in arbitrary road environments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2751–2766, 2016.
- [18] G. Yuan, J. Zhao, S. Xia, Y. Zhang, and W. Li, "Multi-granularity periodic activity discovery for moving objects," *International Journal of Geographical Information Science*, vol. 31, no. 3, pp. 435–462, 2017.
- [19] X. Yin and Q. Chen, "Trajectory generation with spatio-temporal templates learned from demonstrations," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 4, pp. 3442–3451, 2017.
- [20] H. Rong, A. P. Teixeira, and G. G. Soares, "Data mining approach to shipping route characterization and anomaly detection based on AIS data," *Ocean Engineering*, vol. 198, pp. 1–12, 2020.
- [21] Z. Yan, Y. Xiao, L. Cheng et al., "Exploring AIS data for intelligent maritime routes extraction," *Applied. Ocean. Research*, vol. 101, pp. 1–10, 2020.
- [22] V. F. Arguedas, G. Pallotta, and M. Vespe, "Maritime traffic networks: from historical positioning data to unsupervised maritime traffic monitoring," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 722–732, 2018.
- [23] C. Iphar, C. Ray, and A. Napoli, "Data integrity assessment for maritime anomaly detection," *Expert Systems with Applications*, vol. 147, no. 113219, pp. 113219–113219, 2020.
- [24] X. Pan, H. Wang, X. Cheng, X. Peng, and Y. He, "Online detection of anomaly behaviors based on multidimensional trajectories," *Information Fusion*, vol. 58, pp. 40–51, 2020.
- [25] T. Watawana and A. Caldera, "Analyse near collision situations of ships using automatic identification system dataset," in *In 2018 5th International Conference on Soft Computing & Machine Intelligence (ISCMI)*, Nairobi, Kenya, 2018.
- [26] J. Mao, C. Jin, Z. Zhang, and A. Zhou, "Anomaly detection for trajectory big data: advancements and framework," *Journal of Software*, vol. 28, no. 1, pp. 17–34, 2017.
- [27] T. Zhang, S. Zhao, and J. Chen, "Ship trajectory outlier detection service system based on collaborative computing," in *In 2018 IEEE World Congress on Services (SERVICES)*, pp. 15–16, San Francisco, CA, 2018.
- [28] T. Watawana and A. Caldera, "Anomaly detection in maritime data based on geometrical analysis of trajectories," in *In 2015 18th International Conference on Information Fusion (Fusion)*, pp. 1100–1105, Washington DC, USA, 2015.
- [29] D. Zhang, J. Li, Q. Wu, X. Liu, X. Chu, and W. He, "Enhance the AIS data availability by screening and interpolation," in *In 2017 4th International Conference on Transportation Information and Safety (ICTIS)*, pp. 981–986, Banff, Canada, 2017.
- [30] X. Zhang, Y. He, R. Tang, J. Mou, and S. Gong, "A novel method for reconstruct ship trajectory using raw AIS data," in *In 2018 3rd IEEE International Conference on Intelligent Transportation Engineering (ICITE)*, pp. 192–198, Singapore, 2018.

## Research Article

# UAV Path Planning Based on Multicritic-Delayed Deep Deterministic Policy Gradient

Runjia Wu,<sup>1</sup> Fangqing Gu ,<sup>1</sup> Hai-lin Liu,<sup>1</sup> and Hongjian Shi<sup>2</sup>

<sup>1</sup>*School of Mathematics and Statistics, Guangdong University of Technology, Guangzhou, China*

<sup>2</sup>*Beijing Normal University-Hong Kong, Baptist University United International College, Zhuhai, China*

Correspondence should be addressed to Fangqing Gu; [fqgu@gdut.edu.cn](mailto:fqgu@gdut.edu.cn)

Received 3 January 2022; Accepted 17 February 2022; Published 14 March 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Runjia 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.

Deep deterministic policy gradient (DDPG) algorithm is a reinforcement learning method, which has been widely used in UAV path planning. However, the critic network of DDPG is frequently updated in the training process. It leads to an inevitable overestimation problem and increases the training computational complexity. Therefore, this paper presents a multicritic-delayed DDPG method for solving the UAV path planning. It uses multicritic networks and delayed learning methods to reduce the overestimation problem of DDPG and adds noise to improve the robustness in the real environment. Moreover, a UAV mission platform is built to train and evaluate the effectiveness and robustness of the proposed method. Simulation results show that the proposed algorithm has a higher convergence speed, a better convergence effect, and stability. It indicates that UAV can learn more knowledge from the complex environment.

## 1. Annotation Demo Section

In recent years, unmanned aerial vehicles (UAVs) have been widely applied, and their high maneuverability and rapidly deployable UAVs have been applied to search and rescue [1], multi-UAV cooperation [2], formation flight [3], remote surveillance [4], and other fields [5–7]. UAV faces a variety of complex challenges and complicated tasks. Among them, path planning is the first problem faced by UAV. How to make the UAV safely fly to the destination in an unknown working environment becomes a hot topic for researchers. Faced with complex and uncertain environments, many algorithms have been proposed for solving the UAV navigation problems. The most common method is the motion control problem in the unknown environment such as A-Star [8], artificial potential fields [9], rapidly exploring random tree (RRT) algorithm [10], and so on [11–15]. However, due to the constraints of model mismatch, insufficient measurement means, high accurate cost, and model migration, it is difficult to obtain an accurate dynamic model of aircraft in practical engineering. These model-based strate-

gies can hardly be applied to practice in a complex uncertain environment.

In order to overcome the limitations of model-based strategies in uncertain environments, some researchers introduce learning-based methods to overcome these shortcomings. The first is supervised learning that uses large amounts of data to simulate real situations [16, 17]. However, supervised learning needs enough data; it cannot simulate a lot of changes in the real world [18]. The other is reinforcement learning (RL), which uses the interactive learning of the mapping from environment to behavior to seek the most accurate or optimal action decision by maximizing the state-value function and the action-value function [19]. Reinforcement learning has been applied in UAV path planning problem [20]. It transforms the UAV online path planning problem into a decision problem. The next time series of UAV actions are decided according to the current environment and its own state determined by sensors or external information. In the unknown complex environment, UAV has little prior knowledge related to the environment. Therefore, it is required to have strong



adaptive ability to such uncertainty. RL provides a better idea for this kind of problem by using historical data to obtain the nonlinear function relationship between approximate fitting state and overall performance [21–24].

Reinforcement learning has been extensively studied in recent years. For example, DeepMind innovatively proposed a deep reinforcement learning (DRL) through the combination of deep learning (DL) and RL. DRL transforms high-dimensional input into a lower-dimensional state. It achieved a promising result. Currently, Mnih et al. [25] proposed a deep Q-network (DQN) algorithm, which utilized the powerful function fitting ability of deep neural network to avoid the huge storage space of Q table. DQN enhances the stability of training process by using experiential replay memory and target network. Double DQN [26] and dueling DQN [27] are proposed gradually along with DRL research to overcome the defect of overestimation. DQN algorithm has achieved great success in discrete space. However, in high-dimensional continuous space, DQN will increase exponentially with the increase of discrete degree of action segmentation, resulting in training difficulties. Actor-critic (A-C) algorithm [28] adopts a method similar to policy gradient and uses actor network to output the probability value of the action, while critic network is responsible for evaluating the output action. A-C algorithm uses critic network approximate value function to guide agent update and provide low variance learning knowledge [29, 30]. Lillicrap et al. [31] proposed a deep deterministic policy gradient (DDPG) algorithm to improve the stability of A-C algorithm evaluation by using the target network and empirical replay of DQN. DDPG can be applied in the applications with continuous action space and achieve great success [32]. However, the performance of DDPG in practical applications is not very stable.

Reinforcement learning is independent of environmental models and prior knowledge. Thus, it can effectively solve the UAV path planning problem in unknown environments. The research of reinforcement learning in UAV path planning has received extensive attention from scholars. UAV navigation was modeled as a reinforcement learning problem and validated autonomous flight in unknown environments in [33]. Junell et al. [34] used reinforcement learning method to solve the flight test of quadrotor aircraft in an unknown environment. The continuity of DDPG is widely used in path planning, but its convergence is often unstable in complex environments. Model-free reinforcement learning algorithms are based on time division or Monte Carlo [35]. It suffers from the problem of overestimation. In large state space, the application of policy gradient method will bring a high variance of estimation results. It makes policy learning more sensitive and even leads to training failure.

Numerous researchers improved the accuracy of numerical estimation by improving the neural network. For example, double DQN [26] is guaranteed not to overestimate Q value via two critic networks. Twin-delayed DDPG (TD3) [36, 37] algorithm solves the overestimation problem by introducing three key technologies. In a real UAV flight, paper [38] solved the problem of slow convergence caused by sparse rewards by introducing the reward function of an artificial potential field. Paper [39] started with DDPG experience base combined with

simulated annealing algorithm, and accelerated the learning process of DRL through multiexperience pool (MEP). Papers [38, 40] use LSTM to approximate the critic network by combining the current training observation sequence with the historical observation sequence, so that the UAV can break away from the U-shaped obstacle in large path planning. Paper [41] presented three improvements, environmental noise, delay learning, and hybrid exploration techniques, to enhance the robustness of DDPG. Nevertheless, robustness is still a great challenge for UAV path planning. In the TD3 algorithm, the algorithm solves the critic's overestimation problem by the method of clipped double Q-learning for actor-critic. However, only using low estimation often leads to slow convergence.

In order to solve the problem that actor network relies heavily on critic network, which makes DDPG performance very sensitive to critic learning, this paper proposes a multicritic-delayed DDPG method for solving UAV path planning. It uses the average estimation of multicritics network to reduce DDPG's dependence on critic network and delayed learning method to reduce the overestimation problem of DDPG and reduce the error accumulation of the target network. Considering the sensitivity of the UAV to parameters in the real environment, adding Gaussian noise to action and state increases the robustness of the UAV. The main contributions of this paper are as follows:

- (1) We propose a multicritic-delayed DDPG method, which includes two improvement techniques. The first is to add state noise and regularize it, which increases the robustness training network. The second is to use multicritic to average error and solve the error accumulation caused by overestimation
- (2) We apply the proposed multicritic-delayed deep deterministic policy gradient method for solving UAV path planning. A nonsparse reward mode is designed
- (3) A UAV mission platform is built to train and evaluate the effectiveness and robustness of the proposed method. Simulation results show that the proposed algorithm is effective with strong robust and adaptive capability for solving the path planning of the UAV flying destination under complex environment

The remainder of this paper is organized as follows: In Section 2, we briefly review reinforcement learning methods, i.e., DDPG, TD3, and MCDDPG, for solving UAV path planning. Section 3 gives a detailed description of the proposed multicritic-delayed deep deterministic policy gradient method. Section 4 provides the simulation results and analyses the empirical results. Finally, we draw a conclusion and future work in Section 5.

## 2. Reinforcement Learning for Solving UAV Path Planning

*2.1. UAV Motion Model.* Motion model of UAV is a basis of path planning problem. The UAV system is usually controlled by six degrees of freedom, representing three

coordinates of the UAV position  $[x, y, z]$  and controlling the three freedoms of the yaw angle  $\psi$ , the roll angle  $\delta$ , and the pitch angle  $\phi$ . Six degrees of freedom kinematics mode  $[x, y, z, \psi, \delta, \phi]$  is used to describe the internal state of the UAV.

For the sake of brevity and without loss of generality, we adopt the kinematic model of three degrees of freedom instead of six degrees of freedom. Assume that the UAV is fixed at a horizontal altitude, so that UAV's activity is confined to the  $x - y$  plane. Ignoring the momentum impact of the UAV during flight, assuming that the UAV adopts a constant velocity  $v$ , the vector  $\zeta = [x, y, \psi]$  is used to simplify the description of the position and motion of the UAV. Therefore, the vector  $\zeta$  can be expressed as:

$$\begin{cases} x_{t+1} &= x_t + v_t \times \cos \psi_t, \\ y_{t+1} &= y_t + v_t \times \sin \psi_t, \\ \psi_{t+1} &= \psi_t + \Delta\psi. \end{cases} \quad (1)$$

where  $\Delta v$  is the change in velocity and  $\Delta\psi$  is the change in yaw angle.

The state obtained by the environment of UAV is composed of three parts, i.e., the internal state, the interactions with the environment, and the location of the target. There are six coordinates  $\xi_u = [x, y, x_v, y_v, v, \psi]$  representing the information about the their internal state, where  $(x, y)$  is the absolute position of the UAV,  $(x_v, y_v)$  is the velocity component of the corresponding coordinate,  $v$  is the speed of flight, and  $\psi$  is the yaw angle.  $(v, \psi)$  is used to control the internal information of the UAV flight. The surrounding environment state is determined by radar, range finder, and other tools in the real-time interaction between UAV and environment. In this paper, we use range finders to receive the environment state  $\xi_f = [d_0, \dots, d_N]$ , where  $d_i$  is the  $i$ th range finder mounted on the UAV as shown in Figure 1(b). Besides, the target position of the UAV is expressed as  $\xi_T = [x_T, y_T]$  as shown in Figure 1(a). Through the combination of the three observation methods, we can obtain the final description of the state  $s$  by combining  $\xi_u$ ,  $\xi_f$ , and  $\xi_T$ .

$$\mathbf{s} = [x, y, x_v, y_v, v, \psi, d_0, \dots, d_N, x_T, y_T]. \quad (2)$$

The control of UAV is complicated in the actual situation, which requires multiple commands to achieve the motion of the UAV. In this paper, we appropriately selected the UAV's speed and roll as the motion control commands. The control vector of UAV is  $\mathbf{a} = [a_v, a_\psi]$ , where  $a_v \in [-1, 1]$  denotes the ratio of the current speed to the maximum speed and  $a_\psi \in [-1, 1]$  is a steering signal that can be selected to turn the UAV to the desired roll angle.

**2.2. Reinforcement Learning.** In reinforcement learning, the agent changes its state through interaction with the environment so as to obtain returns and achieve the optimal strategy. The model is usually expressed using five tuples  $S, A, P, R, \gamma$  of a Markov decision process (MDP), where  $S$  is a collection of environmental state descriptions of  $\mathbf{s}$  and  $A$  is a set of all pos-

sible actions  $\mathbf{a}$ .  $P : S \times A \times S \rightarrow [0, 1]$  represents the transition probability of taking an action from  $S$  to the next  $S$ .  $R = S \times A$  represents the immediate reward after the agent takes action. Reinforcement learning is designed to maximize future rewards, and a set of rewards can be expressed as  $R_t^y = \sum_{i=t}^T \gamma^{i-t} r(\mathbf{s}_i, \mathbf{a}_i)$ . Based on the reward, reinforcement learning introduces two functions, the state-valued function when an agent adopts the policy  $\pi$ :

$$V_\pi(\mathbf{s}_t) = \mathbb{E} \left[ \sum_{l=0}^{\infty} \gamma^l r(\mathbf{s}_{t+l}, \mathbf{a}_{t+l}) | \mathbf{s}_t \right], \quad (3)$$

where  $\pi$  is to map system states to a probability distribution over the actions.

And the action value function:

$$Q_\pi(\mathbf{s}_t, \mathbf{a}_t) = \mathbb{E} \left[ \sum_{l=0}^{\infty} \gamma^l r(\mathbf{s}_{t+l}, \mathbf{a}_{t+l}) | \mathbf{s}_t, \mathbf{a}_t \right], \quad (4)$$

where  $\gamma \in [0, 1]$  is the discounting factor which represents the difference in importance between future rewards and present rewards.

The value functions are used to measure the advantages and disadvantages of a certain state or action state, that is, whether it is worth an agent to select a certain state or execute an action in a certain state. Figure 2 illustrates the control of the agent under reinforcement learning model.

**2.3. Nonsparse Reward Model.** The reward function of traditional RL uses a simple sparse reward model, that is, an agent gets the reward only when the agent reaches the destination. This paper utilizes a nonsparse reward method to provide guidance for model learning. Obviously, nonsparse rewards provide more navigation domain knowledge than sparse rewards and do not change the policy invariance of rewards.

The nonsparse reward consists of four constructions:

$$r(\mathbf{s}, \mathbf{a}) = \lambda_1 r_1 + \lambda_2 r_2 + \lambda_3 r_3 + \lambda_4 r_4, \quad (5)$$

where  $\lambda_1, \lambda_2, \lambda_3$ , and  $\lambda_4$  are the contribution rates of the four items, and

$$\begin{aligned} r_1 &= d_{\text{pre}} - d_{\text{cur}}, \\ r_2 &= -r_{\text{step}}, \\ r_3 &= -\Delta\psi + r_{\text{free}}, \\ r_4 &= -e^{-\omega d_{\text{obs}}}, \end{aligned} \quad (6)$$

where  $r_1$  represents the change in distance between the current position and the destination and  $d_{\text{pre}}$  and  $d_{\text{cur}}$  are the previous and current relative distance between UAV and the target. When  $d_{\text{pre}} > d_{\text{cur}}$ ,  $r_1$  is a reward that is related to speed, guiding the UAV to its destination quickly.  $r_2$  is a constant penalty advance to the UAV reaching its destination with a minimum number of steps. The UAV should

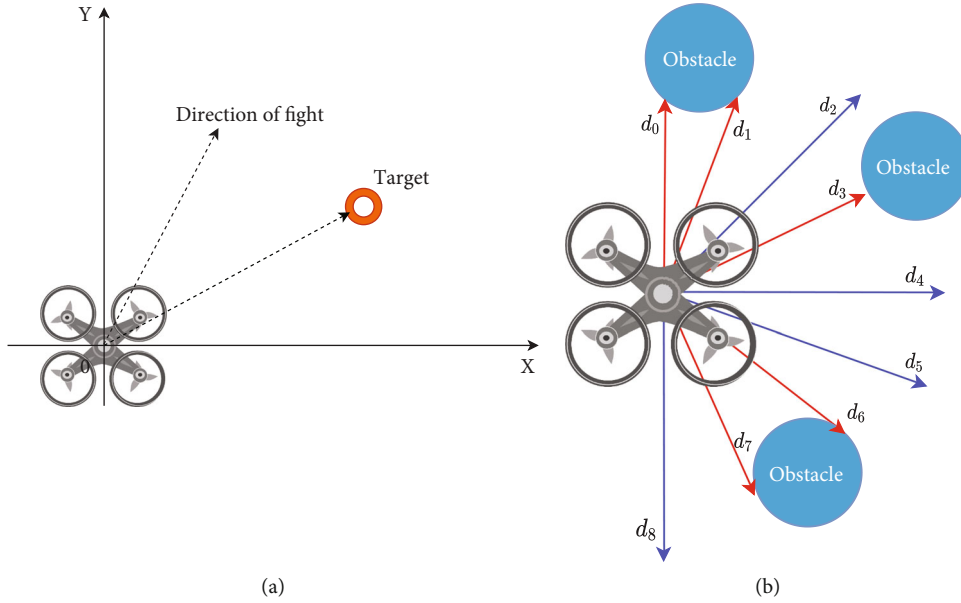


FIGURE 1: (a) The direction of the UAV on the coordinate axis. (b) The sensor of the UAV.

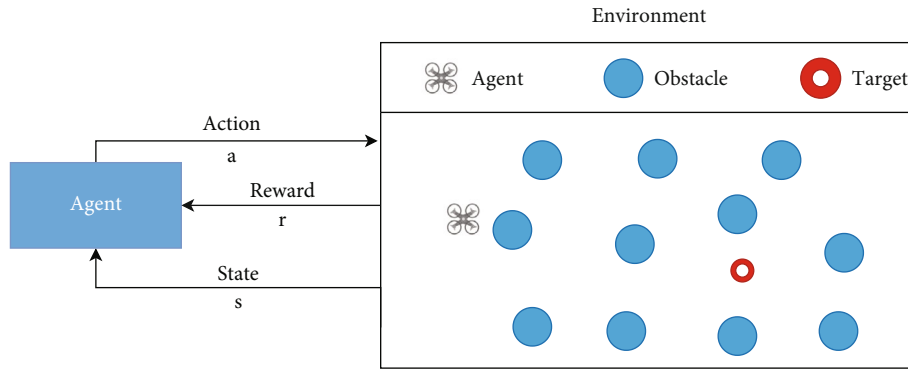


FIGURE 2: Control of agent under reinforcement learning model.

be encouraged to complete its missions as quickly as possible and punished after each transition.  $r_{\text{free}}$  represents a reward for flying without obstacles. Encourage UAV to fly to accessible places to explore more space.  $r_3$  encourages UAV to shorten its range but at the same time ensure it can explore more space. The UAV should be able to fly toward its targets as soon as possible with penalties for deviations, but it should be encouraged to move towards free space if there are obstacles in the direction of flight.  $r_4$  prevents the UAV from getting too close to the obstacle, and  $d_{\text{obs}}$  is the minimum distance between the UAV and the obstacle, and  $\omega$  is a constant which is to control the size of the distance. Exponential function is used to prevent UAV from getting too close to obstacles, but it can also fly near obstacles. The UAV should actively avoid obstacles. If it gets close to obstacles, it will be punished greatly, so as to ensure that the UAV can stay away from obstacles.

**2.4. Deep Deterministic Policy Gradient.** Deep deterministic policy gradient (DDPG) adopts the network framework of actor-critic reinforcement learning, the critic can judge the

value of the action based on the actor, and the actor can modify the probability of the action based on the value of the critic. The convolution neural networks  $Q$  network and  $\mu$  network of DDPG method are used to approximate the state-action value function (3) and state-value function (4), respectively.  $\theta^Q$  and  $\theta^\mu$  are the parameters of the  $Q$  network and  $\mu$  network. The critic network learns the state-action value by minimizing time-difference (TD) errors:

$$L = \left( r(\mathbf{s}_t, \mathbf{a}_t) + \gamma Q'(\mathbf{s}_{t+1}, \mathbf{a}_{t+1} | \theta^{Q'}) - Q(\mathbf{s}_t, \mathbf{a}_t | \theta^Q) \right)^2. \quad (7)$$

In addition, deep Q-learning target network is used to remove the coupling during the update of formula (7). Figure 3 illustrates the DDPG motion control framework, where  $Q'$  represents the target critic network and  $\mu'$  represents the target actor network.

Obviously, we know that the samples obtained by the agent in RL are highly correlated. The researcher uses the reply buffer to address this problem. The correlation of samples is broken by storing experiences  $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1})$ , and

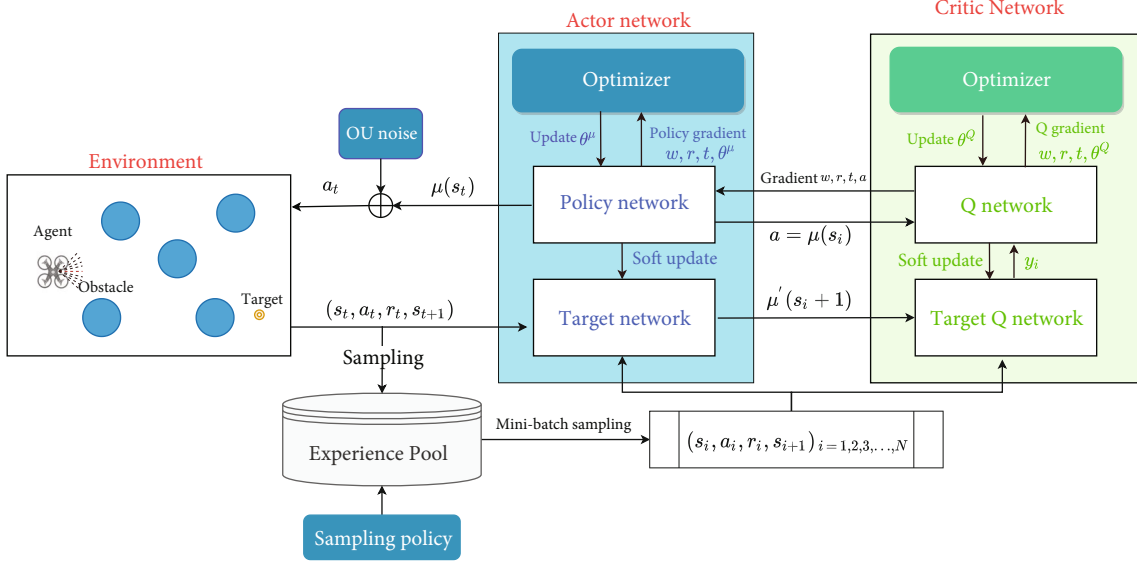


FIGURE 3: DDPG motion control framework.

then, random samples are taken from experience reply when the network trains. DDPG is derived from the deterministic policy gradient theorem for MDP. In this theorem, for MDP with continuous action space, the deterministic policy gradient exists. When the variance of probability policy approaches zero, it is deterministic action  $\mathbf{a}_t = \mu(\mathbf{s}_t | \theta^\mu)$ , i.e.,

$$\nabla_{\theta^\mu} J(\theta^\mu) = \mathbb{E}_{\mathbf{s} \sim \rho^\pi} \left[ \nabla_{\theta^\mu} \mu(\mathbf{s} | \theta^\mu) \nabla_{\mathbf{a}} Q(\mathbf{s}, \mathbf{a} | \theta^Q) \Big|_{\mathbf{a}=\mu_\theta(\mathbf{s})} \right], \quad (8)$$

where  $\rho^\pi$  denotes the state distribution under a selected policy  $\pi$ . The actor network guides the choice of actions by maximizing performance objectives (8). DDPG trains the network with the stochastic gradient descent (SGD) algorithm with minibatch and then updates the target network with the soft update algorithm:

$$\begin{cases} \theta^{Q'} & \leftarrow \tau \theta^Q + (1 - \tau) \theta^{Q'}, \\ \theta^{\mu'} & \leftarrow \tau \theta^\mu + (1 - \tau) \theta^{\mu'}, \end{cases} \quad (9)$$

where  $\tau$  is the updating rate.

**2.5. Twin-Delayed Deep Deterministic Policy Gradient.** Twin-delayed deep deterministic policy gradient (TD3) adopts an improved clipped variant of double Q-learning to reduce network overestimation problems. Following the idea of double Q-learning, TD3 uses two critics with the same pool of experience. Algorithm 1 contains the pseudocode of TD3. The minimum values of the two networks are used to update the critic networks:

$$L_{cd} = \left( r(\mathbf{s}_t, \mathbf{a}_t) + \gamma \min_{i=1,2} Q'(\mathbf{s}_{t+1}, \mathbf{a}_{t+1} | \theta_i^{Q'}) - Q(\mathbf{s}_t, \mathbf{a}_t | \theta_i^Q) \right)^2, \quad (10)$$

where  $\theta_1^Q$  and  $\theta_2^Q$  are the parameters of two different critic networks and  $\theta_1^{Q'}$  and  $\theta_2^{Q'}$  are the parameters of the corresponding target networks, respectively. The action network is updated only according to  $\theta_1^Q$ . Both  $\theta_1^Q$  and  $\theta_2^Q$  are updated by using Equation (10) to minimize the loss function. When the deterministic policy is updated, the value estimate of narrow peaks will occur. TD3 smooths the small area around the action and adds noise to the Q value of the target action:

$$\begin{aligned} \mathbf{a}_{t+1} &= \mu(\mathbf{s}_{t+1} | \theta^{\mu'}) + \varepsilon, \\ \varepsilon &\sim \text{clip}(\mathcal{N}(0, \sigma), -c, c), \end{aligned} \quad (11)$$

where  $\mathcal{N}$  is a Gaussian distribution with mean 0 and variance  $\sigma$ . Noise  $\varepsilon$  can be seen as a regularization method, which makes value function updates smoother.  $c$  is for controlling the size of noise.

Deterministic policy is prone to errors caused by function approximation errors and increases the variance of the target. The regularization of Equation (11) can smooth the target policy. On the other hand, TD3 uses a stable target network approach to reduce error accumulation. The change is to update only the policy and target network after updating the target critic at a fixed frequency  $d$ . The method of delayed updating the target network can interrupt the accumulation of errors and ensure the TD error is small so as to slowly update the target network  $\theta_i^{Q'} \leftarrow \tau \theta_i^Q + (1 - \tau) \theta_i^{Q'}$ ,  $i = 1, 2$ .

**2.6. MCDDPG.** Although DDPG is widely used in UAV path planning because it can well solve the continuous motion space, a large number of researchers have improved DDPG in UAV application due to the poor stability of the algorithm, and the convergence rate is slow. Because actor's learning ability depends on the judgment, paper [42] proposed a multicritic deep deterministic policy gradient (MCDDPG) to

```

1 ◊ Initialize the critic networks  $Q_1, Q_2$  and actor network  $\mu$  with parameters  $\theta_1^Q, \theta_2^Q$ , and  $\theta^\mu$ , separately.
2 ◊ Initialize the target critic networks  $\theta_1^{Q'} \leftarrow \theta_1^Q, \theta_2^{Q'} \leftarrow \theta_2^Q$  and actor target network  $\theta^{\mu'} \leftarrow \theta^\mu$ , separately.
3 ◊ Initialize the reply buffer  $R$ , maximum flight time  $T$ .
4 forepisode = 1 :  $M$  do.
5 ◊ Reset environment and receive initial observation state  $\mathbf{s}$ .
6 fort = 1 :  $T$  do.
7 ◊ Select action  $\mathbf{a}_t = \mu(\mathbf{s}_t | \theta^\mu) + \mathcal{N}(0, \sigma)$  and obtain the reward  $r_t$  and new state  $\mathbf{s}_{t+1}$ .
8 ◊ Store transition  $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1})$  in  $R$ .
9 ◊ Sample a random minibatch of  $N$  transitions  $(\mathbf{s}_i, \mathbf{a}_i, r_i, \mathbf{s}_{i+1})$  from  $R$ .
10 ◊  $\mathbf{a}_{t+1} \leftarrow \mu(\mathbf{s}_{t+1} | \theta^{\mu'}) + \epsilon, \epsilon \sim \text{clip}(\mathcal{N}(0, \sigma), -c, c)$ .
11 ◊ Update the critic networks by minimizing loss function in Equation (10).
12 if  $t \bmod d$  then
13 ◊ Update the actor network with policy gradient Equation (8).
14 ◊ Update the parameters of target networks with updating rate  $\tau$ .
15 end
16 end
17 end

```

ALGORITHM 1: TD3 method.

```

1 ◊ Initialize the  $K$  critic networks  $Q_i$  and actor network  $\mu$  with parameters  $\theta_i^Q$  and  $\theta^\mu$ , separately.
2 ◊ Initialize the  $K$  target critic networks  $Q_i'$  and actor target network  $\mu'$  with parameters  $\theta_i^{Q'} \leftarrow \theta_i^Q$  and  $\theta^{\mu'} \leftarrow \theta^\mu$ , separately.
3 ◊ Initialize the reply buffer  $R$ , maximum flight time  $T$ , parameters  $\alpha, \beta, \eta$ , updating rate  $\tau$ .
4 forepisode = 1 :  $M$  do
5 ◊ Reset environment and receive initial observation state  $\mathbf{s}$ .
6 fort = 1 :  $T$  do
7 ◊ Select action  $\mathbf{a}_t = \mu(\mathbf{s}_t | \theta^\mu) + \mathcal{N}_t(0, \sigma)$  according to the current policy and exploration noise.
8 ◊ Obtain the reward  $r_t$  and observe new state  $\mathbf{s}_{t+1}$ .
9 ◊ Store transition  $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1})$  in  $R$ .
10 ◊ Sample a random minibatch of  $N$  transitions  $(\mathbf{s}_i, \mathbf{a}_i, r_i, \mathbf{s}_{i+1})$  from  $R$ .
11 ◊ Update the  $K$  critic networks by minimizing loss function in Equation (14).
12 ◊ Update the actor network with policy gradient Equation (8).
13 ◊ Update the parameters of target networks with updating rate  $\tau$ .
14 end
15 end

```

ALGORITHM 2: MCDDPG method.

overcome the sensitivity of training the critic network. MCDDPG is applied for solving UAV path planning in [20]. Algorithm 2 contains the pseudocode of MCDDPG. Specifically, it uses the average of the value of  $K$  critics to approximate instead of the action-value function.

$$Q_{\text{avg}}(\mathbf{s}, \mathbf{a} | \theta^Q) = \frac{1}{K} \sum_{i=1}^K Q_i(\mathbf{s}, \mathbf{a} | \theta_i^Q), \quad (12)$$

where  $\theta_i^Q \in \theta^Q$  is the parameter of the  $i$ th critic. The average of all critics can diminish the impact of the overestimation problem caused by the individual critic. We further rewrote the TD error according to Equation (7). Thus, the average TD error is:

$$L_{\text{avg}} = \left( r(\mathbf{s}_t, \mathbf{a}_t) + \gamma Q_{\text{avg}}(\mathbf{s}_{t+1}, \mathbf{a}_{t+1} | \theta^Q) - Q_{\text{avg}}(\mathbf{s}_t, \mathbf{a}_t | \theta^Q) \right)^2, \quad (13)$$

where the  $Q_{\text{avg}}^Q$  is the average of the target critic networks. Using the same TD error update can cause critics to lose diversity, while a separate update can make a big difference. Therefore, different from DDPG critic network, the local error and global error must be considered when calculating the loss of critic network. According to Equation (13), the loss function for  $i$ th critic is defined as:

$$L_{\text{mc}} = \alpha L_{\text{avg}} + \beta L + \eta (Q_i(\mathbf{s}_t, \mathbf{a}_t | \theta_i) - Q_{\text{avg}}(\mathbf{s}_t, \mathbf{a}_t | \theta)), \quad (14)$$

where  $\alpha, \beta$ , and  $\eta$  represent the weighting factor. The values of  $\alpha, \beta$ , and  $\eta$  are between 0 and 1. And the sum of  $\alpha$  and  $\beta$  is 1. When  $K = 1$ ,  $L_{\text{mc}}$  should be the same as formula (7).

### 3. Multicritic-Delayed DDPG Method

The delayed updating of the critic network is very important in practical applications. The critic network of the traditional DDPG method is frequently updated in the training



```

1 ◊ Initialize the  $K$  critic networks  $Q_i$  and actor network  $\mu$  with parameters  $\theta_i^Q$  and  $\theta^\mu$ , separately.
2 ◊ Initialize the  $K$  target critic networks  $Q_i'$  and actor target network  $\mu'$  with parameters  $\theta_i^{Q'} \leftarrow \theta_i^Q$  and  $\theta^{\mu'} \leftarrow \theta^\mu$ , separately.
3 ◊ Initialize the reply buffer  $R$ , maximum flight time  $T$ , parameters  $\alpha, \beta, \eta$ , updating rate  $\tau$ .
4 forepisode = 1 :  $M$ do
5 ◊ Reset environment and receive initial observation state  $\mathbf{s}$ .
6 fort = 1 :  $T$ do
7 ◊ Select action  $\mathbf{a}_t = \mu(\mathbf{s}_t | \theta^\mu) + \mathcal{N}_t(0, \sigma)$  according to the current policy and exploration noise.
8 if episode >  $T_m$  then
9 ◊  $\mathbf{a}_t = \mu(\mathbf{s}_{\text{noise}} | \theta^\mu)$  according to Equation (15).
10 end
11 ◊ Obtain the reward  $r_t$  and observe new state  $\mathbf{s}_{t+1}$ , and Store transition  $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1})$  in  $R$ .
12 ◊ Sample a random minibatch of  $N$  transitions  $(\mathbf{s}_i, \mathbf{a}_i, r_i, \mathbf{s}_{i+1})$  from  $R$ .
13 ◊ Update the  $K$  critic networks by minimizing loss function in Equation (14).
14 if mod  $d$  then
15 ◊ Update the actor network with policy gradient Equation (8).
16 ◊ Update the parameters of target networks with updating rate  $\tau$ .
17 end
18 end
19 end

```

ALGORITHM 3: MCD method.

process. It may lead to increase the training steps and cause overestimation. If overestimated actions occur in the learning process, agents will get lost in the learning process, leading to training failure. From the perspective of UAV, delayed update strategy is equivalent to global guidance of UAV flight, while traditional DDPG guides UAV flight through small correction. The existing studies show that the delayed update strategy is more in line with actual UAV flight guidance. Thus, this paper proposes a multicritic-delayed DDPG method, named, MCD, for solving the UAV path planning problem. Algorithm 3 contains the pseudocode of MCD. It uses the delayed update strategy to improve the robustness of the algorithm. TD3 using clipped double Q network can effectively solve the overestimation problem caused by neural network, it also leads to underestimation at the same time. We adopt the error updating method formula (14) of multicritic to approximate the Q value of critic network in the proposed MCD.

MCD prevents network underestimation by retaining the global mean error of multicritic networks and preserving at the same time the error between the average and the individual guarantees diversity. Another improvement is to add noise to the state. DDPG increases agent's exploration of the environment by adding an Ornstein-Uhlenbeck (OU) noise. In fact, the acquisition of the real environment by UAV is often inaccurate. If the UAV is overly dependent on the training model, the deviation of state will often cause the UAV to crash. We simulate the deviation of the environmental state input in the real situation by adding a Gaussian noise:

$$\mathbf{s}_{\text{noise}} = \mathbf{s} + \mathcal{N}(0, \sigma_s). \quad (15)$$

These noises obey the standard deviation  $\sigma_s$  of Gaussian distribution. When the model is trained to  $T_m$ , the robustness of the state noise training network is introduced. Introducing noise after the network has been trained to a certain extent

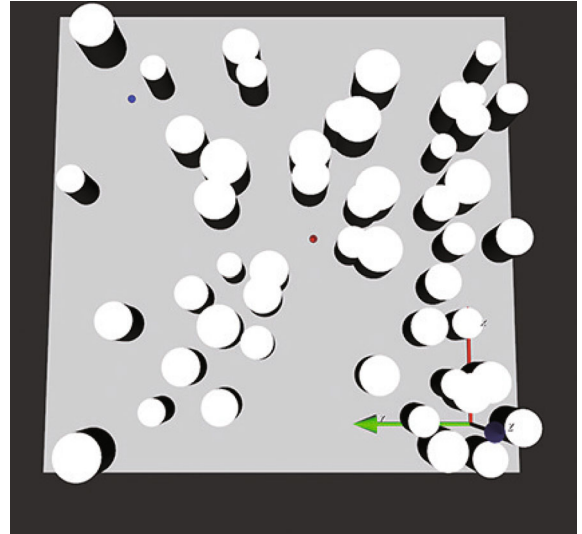


FIGURE 4: UAV's obstructed environment.

can ensure that the initial network will not be trained to fail by noise.

## 4. Experiments

In this section, in order to verify the performance of the proposed MCD, we compare it with three algorithms, i.e., DDPG, TD3, and MCD, on a synthetic test problem.

*4.1. Experimental Platform Setting.* We built a random environment of different complexity. The terrain of each environment is a rectangular area of  $1000 \times 1000$ . As shown in Figure 4. The simulation environment is randomly generated by 49 cylindrical obstacles with diameters of (30, 60) in the rectangular region (there may be overlapping of cylindrical obstacles). The UAV is fixed at horizontal altitude of

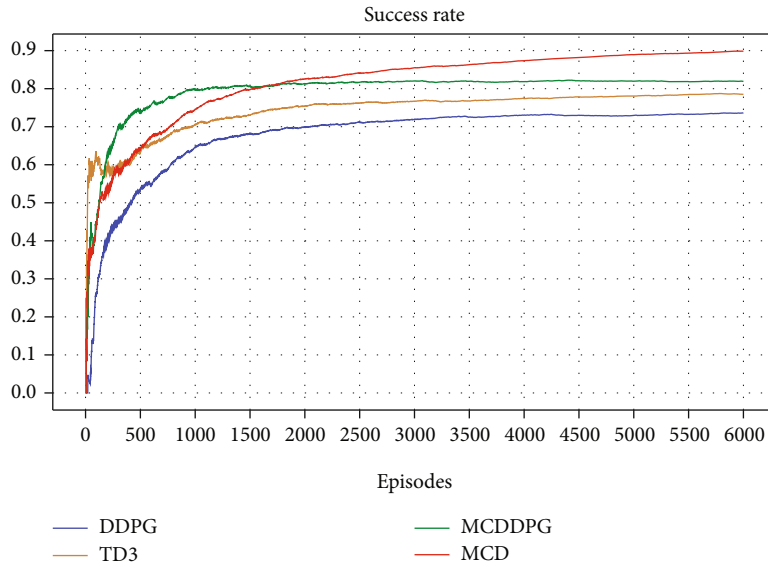
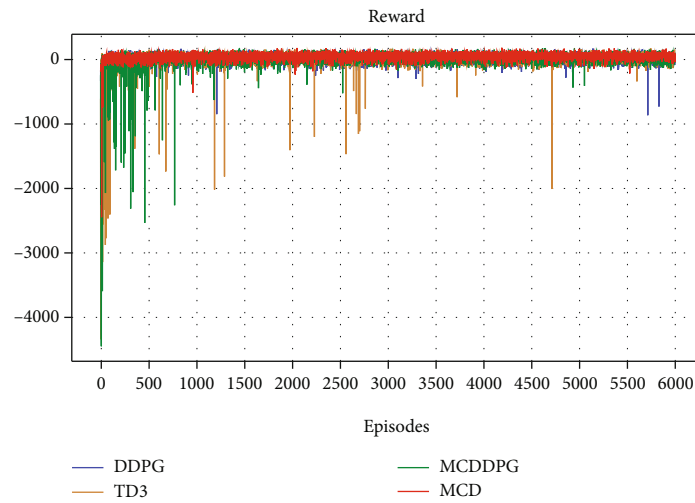
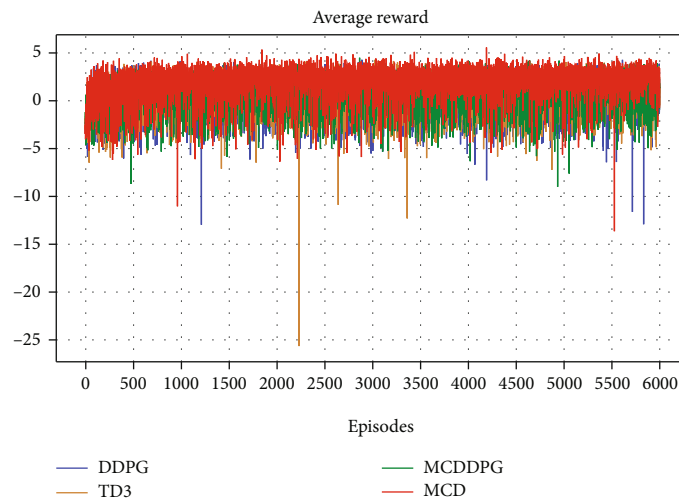


FIGURE 5: The success rate of reaching the target in training.



(a)



(b)

FIGURE 6: The 3000 set of the training.

TABLE 1: The result of algorithms.

	Learning stage			Exploiting stage		
	Success	Collision	Loss	Success	Collision	Loss
DDPG	73.6%	19.3%	7.1%	80.5%	10.1%	9.4%
TD3	78.5%	17.1%	4.4%	88.4%	5.6%	6.0%
MCDDPG	81.9%	15.8%	2.3%	92.1%	3.4%	4.5%
MCD	89.8%	10.1%	0.1%	94.3%	1.9%	3.8%

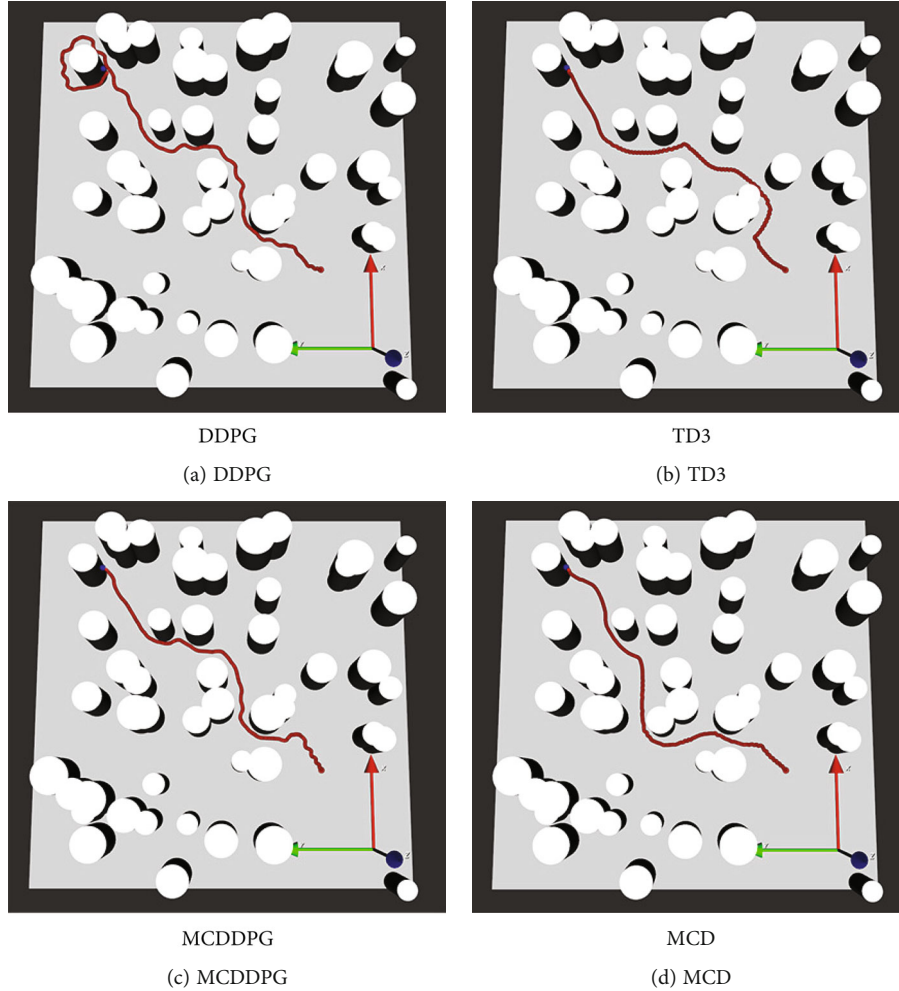


FIGURE 7: Performance of four algorithms in the same environment.

100 meters and equipped with nine sensors with a detection range of 100 meters. The UAV flies from its initial position to its designated target. The maximum speed of the UAV is limited to  $a_{v,max} = 45$  m/s, and the maximum yaw is  $a_{\psi,max} = \pi/2$ .

The critic networks adopt the same network structure of  $19 \times 200 \times 300 \times 1$ . The observed states as inputs are normalized to 19 dimensions, and the actor network composed of  $19 \times 200 \times 300 \times 2$  uses 2 dimensions output action to control the UAV. The parameters  $\alpha, \beta, \eta$  are set to 0.8, 0.2, 0.1 in MCDDPG and MCD. The number  $K$  of the critic networks in Equation (12) is set to be  $K = 3$ . When  $K$  is set too high, the overestimation ability of the algorithm will be greatly reduced but the operation efficiency will be too low.

TABLE 2: The length and the steps of the flight paths obtained by the compared algorithms.

	Path length	Step
DDPG	1432.2 m	112
TD3	78.5 m	242
MCDDPG	81.9 m	124
MCD	89.8 m	153

According to paper [42],  $K = 3$  is a more appropriate value. UAV observation and action are normalized to  $[-1, 1]$ . Adam optimizer [43] is used to learn network parameters. The learning rates of the actor and critic networks are set

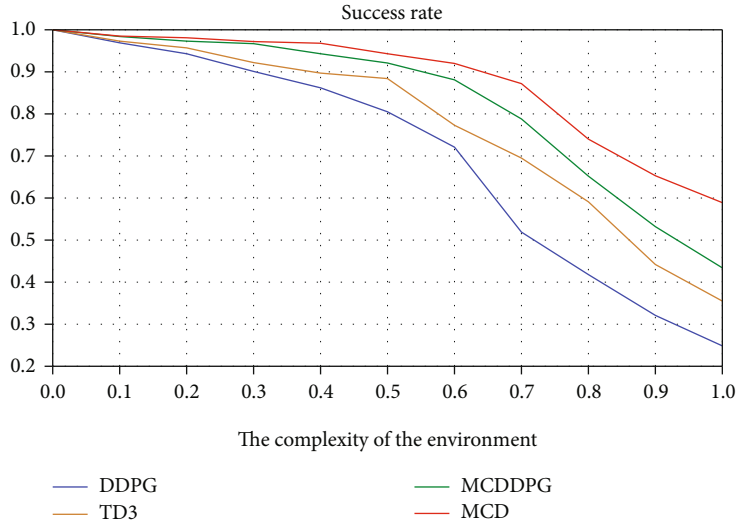


FIGURE 8: Success rates in different environmental complexities under 2000 sets of training.

to be  $10^{-3}$ . In addition, the discount factor is  $\gamma = 0.99$ , and the soft update rate is  $\tau = 0.001$ . The other hyperparameters are given as follows: minibatch size  $N = 64$ , experience replay  $R = 10000$ . In addition, Gaussian distribution  $\mathcal{N}(0, 0.25)$  is used to increase motion detection space, and the Gaussian distribution deviation  $\sigma_s$  of the state noise is 0.2. When the training number reaches  $T_m = 4000$ , the state noise will be enabled. In TD3, the action adds smoothing  $\mathcal{N}(0, 0.2)$  and is clipped to  $(-0.5, 0.5)$ . The parameter of reward is set to  $\lambda_1 = 1.2$ ,  $\lambda_2 = 1.5$ ,  $\lambda_3 = 32$ , and  $\lambda_4 = 25$ , and the maximum value of iteration is  $T = 1000$ .

**4.2. Performance of Multicritic Delayed.** As shown in Figure 5, 3000 sets were used to train both models. MCD has a success rate of 89.8%, which is significantly better than that of the compared algorithms. In the training of MCD, the training effect is not as good as that of TD3 and MCDDPG in the early, which is due to the minimization of multicritic networks. However, it ensures that the MCD estimation is not too high and can grow steadily, and it obviously exceeds other algorithms in the later period. For more specific verification, we trained the model three times, intercepted 3000 episodes in 6000 sets and averaged them, and selected the number of the average reward per episode and the total reward per episode, as shown in Figure 6. From this figure, we can see that MCD fluctuates much less and rewards better than the other three algorithms.

In order to prove the generalization ability of the algorithms, we further calculated the success rate, collision rate, and loss rate of agents. Table 1 lists the results obtained by the compared algorithms. The results showed that the generalization of MCD is better than that of TD3. The success rate is 94.3% which is more than 88.4% of TD3. It is proved that the algorithm can effectively improve the success rate and obtain more environmental information by using the average critic network. We can note that the loss rate of model exploiting is greatly improved compared with the mode

training. The model training can avoid the collision of UAV with obstacles, which is useful in practical situations.

**4.3. Testing of Different Algorithms.** In this part, we will use the actor network completed in the training in the last section to observe the influence of different algorithms on UAV flight. We loaded the actor network with DDPG, TD3, MCDDPG, and MCD algorithms training onto the UAV to guide the UAV flight. From the same starting position (230, 277), the UAV flies through 49 cylindrical obstacles composed of radius (30, 60) and reaches the target position (820, 809). Figure 7 plots the flight paths obtained by the algorithms. Table 2 lists the length and the steps of the flight paths.

From this figure, we can see that UAV moves fast in a way that is close to the obstacle obtained by DDPG algorithm. UAV approaches the obstacle with fewer steps and faster speed, but at the same time, makes it easier to hit the obstacle obtained by DDPG. And because the target point is near the obstacle, DDPG tends to avoid the obstacle when planning the path, resulting in loss in the environment far from the obstacle. UAV flies in a safer manner and can correct the course if the target navigation goes wrong at a small cost obtained MCDDPG. TD3's path planning is more conservative, but it sacrifices time for longer path planning. During the first half of the flight, hesitancy resulted in a zig-zag flight path. Multicritic delay absorbs the advantages of the above three algorithms to reach the destination with the shortest path and takes less turns than TD3, which is not far from the other two algorithms. Obviously, we can get that multicritic delay is superior to other algorithms because it enables the UAV to complete the task with minimal path cost and lower time.

**4.4. Testing of Complex Environment.** To further verify the robustness of MCD in more complex environments, we set up a more complex environmental threat test to investigate the robustness of MCD. We set up a series of environments

with different numbers of obstacles. As shown in Figure 4, it represents a complex environment with a density of 0.5. Density 1 represents 100 obstacles, and 10 obstacles are reduced for every decrease of 0.1. We repeated 2,000 episodes of the four algorithms in the same obstacle environment, redeploying drones and targets in each episode. The success rate for 2000 sets is shown in Figure 8. Obviously, as the number of obstacles rose, the success rate of all four algorithms began to decline. However, MCD algorithm declined the most slowly and still maintained a success rate of 58.9% in the most complex environment. The other three algorithms MCDDPG, TD3, and DDPG are reduced to 43.4%, 35.5%, and 24.8%, respectively. Therefore, MCD is highly adaptable to complex environments.

## 5. Conclusion

In this paper, we proposed a reinforcement learning method, named, MCD, for solving the UAV path planning problem under a complex environment. It uses multicritic networks and delayed learning methods to reduce the overestimation problem of DDPG and adds noise to improve the robustness in the real environment. Moreover, a UAV mission platform is built to train and evaluate the effectiveness and robustness of the proposed method. Simulation results show that the proposed algorithm is superior to the traditional DDPG in path planning. However, some issues remain to be resolved, such as MCD hyperparameter settings, improvements to nonsparse rewards, and experience replay settings.

## Data Availability

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

## Conflicts of Interest

We declare that there is no conflict of interest.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (62172110), in part by the Natural Science Foundation of Guangdong Province (2021A1515011839), and in part by the Programme of Science and Technology of Guangdong Province (2021A0505110004 and 2020A0505100056).

## References

- [1] T. Tomic, K. Schmid, P. Lutz et al., "Toward a fully autonomous UAV: research platform for indoor and outdoor urban search and rescue," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 46–56, 2012.
- [2] Q. Yang, Y. Zhu, J. Zhang, S. Qiao, and J. Liu, "UAV air combat autonomous maneuver decision based on DDPG algorithm," in *2019 IEEE 15th International Conference on Control and Automation (ICCA)*, pp. 37–42, Edinburgh, UK, 2019.
- [3] H. Sira-Ramirez, R. Castro-Linares, and G. Puriel-Gil, "An active disturbance rejection approach to leader-follower controlled formation," *Asian Journal of Control*, vol. 16, no. 2, pp. 382–395, 2014.
- [4] R. Stevens, F. Sadjadi, J. Braegelmann, A. Cordes, and R. Nelson, "Small unmanned aerial vehicle (UAV) real-time intelligence, surveillance and reconnaissance (ISR) using onboard pre-processing," *Proceedings of SPIE*, vol. 6967, 2008.
- [5] C. Wu, S. Shi, S. Gu, L. Zhang, and X. Gu, "Deep reinforcement learning-based content placement and trajectory design in urban cache-enabled UAV networks," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8842694, 11 pages, 2020.
- [6] R. R. Murphy, E. Steimle, M. Hall et al., "Robot-assisted bridge inspection," *Journal of Intelligent & Robotic Systems*, vol. 64, no. 1, pp. 77–95, 2011.
- [7] D. Hausamann, W. Zirnig, G. Schreier, and P. Strobl, "Monitoring of gas pipelines – a civil UAV application," *Aircraft Engineering and Aerospace Technology*, vol. 77, no. 5, pp. 352–360, 2005.
- [8] J. K. Howlett, T. W. Mclain, and M. A. Goodrich, "Learning real-time \* path planner for unmanned air vehicle target sensing," *Journal of Aerospace Computing Information and Communication*, vol. 3, no. 3, pp. 108–122, 2006.
- [9] G. Luo, J. Yu, Y. Mei, and S. Zhang, "UAV path planning in mixed-obstacle environment via artificial potential field method improved by additional control force," *Asian Journal of Control*, vol. 17, no. 5, pp. 1600–1610, 2015.
- [10] R. Kala and K. Warwick, "Planning of multiple autonomous vehicles using RRT," in *IEEE International Conference on Cybernetic Intelligent Systems*, pp. 20–25, London, UK, 2011.
- [11] F. J. Rubio, F. J. Valero, J. L. Suñer, and V. Mata, "Simultaneous algorithm for trajectory planning," *Asian Journal of Control*, vol. 12, no. 4, pp. 468–479, 2010.
- [12] A. E. Oguz and H. Temeltas, "On the consistency analysis of A-SLAM for UAV navigation," in *Unmanned Systems Technology XVI, vol. 9084 International Society for Optics and Photonics*, Baltimore, Maryland, United States, 2014.
- [13] R. Strydom, S. Thurrowgood, and M. V. Srinivasan, "Visual odometry: autonomous UAV navigation using optic flow and stereo," in *Proceedings of Australasian conference on robotics and automation*, Melbourne, Melbourne, Australia, 2014.
- [14] J. Tisdale, Z. Kim, and J. K. Hedrick, "Autonomous UAV path planning and estimation," *IEEE Robotics & Automation Magazine*, vol. 16, no. 2, pp. 35–42, 2009.
- [15] V. Roberge, M. Tarbouchi, and G. Labonté, "Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 132–141, 2013.
- [16] N. Smolyanskiy, A. Kamenev, J. Smith, and S. Birchfield, "Toward low-flying autonomous MAV trail navigation using deep neural networks for environmental awareness," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) IEEE*, pp. 4241–4247, Vancouver, BC, Canada, 2017.
- [17] G. Kahn, T. Zhang, S. Levine, and P. Abbeel, "Plato: policy learning using adaptive trajectory optimization," in *2017 IEEE International Conference on Robotics and Automation (ICRA) IEEE*, pp. 3342–3349, Singapore, 2017.
- [18] A. Loquercio, A. I. Maqueda, C. R. Del-Blanco, and D. Scaramuzza, "Dronet: learning to fly by driving," *IEEE*



- Robotics and Automation Letters*, vol. 3, no. 2, pp. 1088–1095, 2018.
- [19] 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*, vol. 21, no. 4, pp. 3133–3174, 2019.
- [20] R. Wu, F. Gu, and J. Huang, “A multi-critic deep deterministic policy gradient UAV path planning,” in *Proceedings of 2020 16th International Conference on Computational Intelligence and Security*, pp. 6–10, Guangxi, China, 2020.
- [21] Y. Zhu, H. Liu, B. Ren, H. Duan, X. She, and Z. Wu, “A model-free flat spin recovery scheme for miniature fixed-wing unmanned aerial vehicle,” in *2019 IEEE International Conference on Unmanned Systems (ICUS) IEEE*, pp. 623–630, Beijing, China, 2019.
- [22] Y. Zeng, X. Xu, S. Jin, and R. Zhang, “Simultaneous navigation and radio mapping for cellular-connected UAV with deep reinforcement learning,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 7, pp. 4205–4220, 2021.
- [23] R. Xie, Z. Meng, Y. Zhou, Y. Ma, and Z. Wu, “Heuristic Q-learning based on experience replay for three-dimensional path planning of the unmanned aerial vehicle,” *Science Progress*, vol. 103, no. 1, pp. 1–18, 2019.
- [24] O. Walker, F. Vanegas, F. Gonzalez, and S. Koenig, “A deep reinforcement learning framework for UAV navigation in indoor environments,” in *2019 IEEE Aerospace Conference IEEE*, pp. 1–14, Big Sky, MT, USA, 2019.
- [25] V. Mnih, K. Kavukcuoglu, D. Silver et al., “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [26] H. Van Hasselt, A. Guez, and D. Silver, “Deep reinforcement learning with double Q-learning,” in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pp. 2094–2100, Arizona, USA, 2015.
- [27] Z. Wang, T. Schaul, M. Hessel, H. V. Hasselt, M. Lanctot, and N. D. Freitas, “Dueling network architectures for deep reinforcement learning,” in *Proceedings of the 33rd International Conference on International Conference on Machine Learning*, pp. 1995–2003, New York, USA, 2016.
- [28] V. R. Konda and J. N. Tsitsiklis, “Actor-critic algorithms,” in *Advances in Neural Information Processing Systems*, pp. 1008–1014, La Jolla, CA, 2000.
- [29] I. Grondman, L. Busoniu, G. A. Lopes, and R. Babuska, “A survey of actor-critic reinforcement learning: standard and natural policy gradients,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1291–1307, 2012.
- [30] X. Liu, X. Wang, and Y. M. Cheung, “FDDH: fast discriminative discrete hashing for large-scale cross-modal retrieval,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2021.
- [31] T. P. Lillicrap, J. J. Hunt, A. Pritzel et al., “Continuous control with deep reinforcement learning,” 2015, <https://arxiv.org/abs/1509.02971>.
- [32] V. Mnih, K. Kavukcuoglu, D. Silver et al., “Playing atari with deep reinforcement learning,” 2013, <https://arxiv.org/abs/1312.5602>.
- [33] S. Ross, N. Melik-Barkhudarov, K. S. Shankar et al., “Learning monocular reactive UAV control in cluttered natural environments,” in *2013 IEEE International Conference on Robotics and Automation*, pp. 1765–1772, Karlsruhe, Germany, 2013.
- [34] J. L. Junell, E. J. Van Kampen, C. C. de Visser, and Q. P. Chu, “Reinforcement learning applied to a quadrotor guidance law in autonomous flight,” in *AIAA Guidance, Navigation, and Control Conference*, p. 1990, Kissimmee, Florida, 2015.
- [35] V. Mnih, A. P. Badia, M. Mirza et al., “Asynchronous methods for deep reinforcement learning,” in *International conference on machine learning*, pp. 1928–1937, New York, New York, USA, 2016.
- [36] S. Fujimoto, H. Van Hoof, and D. Meger, “Addressing function approximation error in actor-critic methods,” 2018, <https://arxiv.org/abs/1802.09477>.
- [37] X. Liu, Z. Hu, H. Ling, and Y. M. Cheung, “MTFH: a matrix tri-factorization hashing framework for efficient cross-modal retrieval,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 3, pp. 964–981, 2021.
- [38] B. Li and Y. Wu, “Path planning for UAV ground target tracking via deep reinforcement learning,” *IEEE Access*, vol. 8, pp. 29064–29074, 2020.
- [39] Z. Hu, W. Kaifang, X. Gao, Y. Zhai, and Q. Wang, “Deep reinforcement learning approach with multiple experience pools for UAV’s autonomous motion planning in complex unknown environments,” *Sensors*, vol. 20, no. 7, p. 1890, 2020.
- [40] C. Wang, J. Wang, Y. Shen, and X. Zhang, “Autonomous navigation of UAVs in large-scale complex environments: a deep reinforcement learning approach,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2124–2136, 2019.
- [41] K. Wan, X. Gao, Z. Hu, and G. Wu, “Robust motion control for UAV in dynamic uncertain environments using deep reinforcement learning,” *Remote Sensing*, vol. 12, no. 4, p. 640, 2020.
- [42] J. Wu, R. Wang, R. Li, H. Zhang, and X. Hu, “Multi-critic DDPG method and double experience replay,” in *Proceedings of 2018 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 165–171, Miyazaki, Japan, 2018.
- [43] D. P. Kingma and J. Ba, “Adam: a method for stochastic optimization,” in *The 3rd International Conference for Learning Representations*, pp. 1–15, Arizona, USA, 2015.

## Research Article

# Solving Sensor Ontology Metamatching Problem with Compact Flower Pollination Algorithm

Wenwu Lian <sup>1</sup>, Lingling Fu <sup>2</sup>, Xishuan Niu <sup>3</sup>, Junhong Feng <sup>3</sup>,  
and Jian-Hong Wang <sup>4</sup>

<sup>1</sup>School of Physics and Telecommunication Engineering, Yulin Normal University, Yulin 537000, China

<sup>2</sup>Business School, Yulin Normal University, Yulin 537000, China

<sup>3</sup>School of Computer Science and Engineering, Yulin Normal University, Yulin 537000, China

<sup>4</sup>Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung 411030, Taiwan

Correspondence should be addressed to Lingling Fu; [sxyfll@126.com](mailto:sxyfll@126.com) and Xishuan Niu; [cookes2000@163.com](mailto:cookes2000@163.com)

Received 22 November 2021; Accepted 1 February 2022; Published 14 March 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Wenwu Lian 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.

To implement co-operation among applications on the Internet of Things (IoT), we need to describe the meaning of diverse sensor data with the sensor ontology. However, there exists a heterogeneity issue among different sensor ontologies, which hampers their communications. Sensor ontology matching is a feasible solution to this problem, which is able to map the identical ontology entity pairs. This work investigates the sensor ontology meta-matching problem, which indirectly optimizes the sensor ontology alignment's quality by tuning the weights to aggregate different ontology matchers. Due to the largescale entity and their complex semantic relationships, swarm intelligence (SI) based techniques are emerging as a popular approach to optimize the sensor ontology alignment. Inspired by the success of the flower pollination algorithm (FPA) in the IoT domain, this work further proposes a compact FPA (CFPA), which introduces the compact encoding mechanism to improve the algorithm's efficiency, and on this basis, the compact exploration and exploitation operators are proposed, and an adaptive switching probability is presented to trade-off these two searching strategies. The experiment uses the ontology alignment evaluation initiative (OAEI)'s benchmark and the real sensor ontologies to test CFPA's performance. The statistical comparisons show that CFPA significantly outperforms other state-of-the-art sensor ontology matching techniques.

## 1. Introduction

To implement the co-operations among applications on the Internet of Things (IoT) [1], we need to describe the meaning of diverse sensor data with the sensor ontology and express them in a machine-interpretable way. As the kernel technique of the semantic web [2], sensor ontologies, such as the CSIRO sensor ontology (CSIRO) [3], semantic sensor network ontology (SSN) [4], and MMI device ontology (MMI) [5], have been widely used in the IoT domain. Although they own lots of overlapped information, the heterogeneity issue also exists among them. For example, a sensor concept might be defined

with different terminologies or contexts, which hampers their communications. Sensor ontology matching is a feasible solution for this problem, which is able to map the identical ontology entity pairs [6].

To find the semantically identical sensor concepts, it is necessary to use the ontology matcher to measure the two concepts' similarity value. However, due to the limitations of the natural language processing domain, one single ontology matcher is not able to ensure its effectiveness in various heterogeneous contexts. The problem of how to comprehensively combine different ontology matchers to make their advantages and disadvantages complement each other, so as to enhance the final ontology alignment's quality, i.e., sensor

ontology meta-matching problem, has attracted many researchers' attention. Fernandez et al. [7] propose a fuzzy theory-based method of aggregating the ontology matchers. Later on, the global and local ontology alignment extracting technique [8] is presented to find the alignment from diverse alignments obtained by different ontology matchers. Their proposal is able to take each correspondence's preference into consideration, which improves the alignment's quality. The generative adversarial network (GAN) [9] is used to iteratively combine different matchers. The Siamese neural network (SNN) [10] is also used to train problem-specific ontology matchers on the basis of the existing ones. The semisupervised learning-based method [11] first requires the expert to provide the partial alignment and then use it to train the Bayesian probability model and determine the rest of the alignment. Multi-objective evolutionary algorithm (MOEA) [12], coevolutionary algorithm (CEA) [13], evolutionary tabu search algorithm (ETSA) [14], co-firefly algorithm (CFA) [15], and differential evolution algorithm (DEA) [16] are also proposed to optimize the sensor ontology alignment. Particle swarm optimization (PSO) [17] is also used to determine highquality sensor ontology alignment, which introduces the simulated annealing strategy (SA) to improve the algorithm's performance by trading off the exploitation and exploration. Inspired by the success of swarm intelligence (SI) in the ontology matching domain, this work investigates a newly emerging SI algorithm, i.e., flower pollination algorithm (FPA) [18], which has successfully been applied in wireless sensor network (WSN) [19] to address complex optimization problem. To overcome the population-based FPA's disadvantages, such as slow convergence speed [20], this work proposes a compact FPA (CFPA) and uses it to address the sensor ontology meta-matching problem. In particular, CFPA uses a probability vector (PV) [21] to present the whole population, and on this basis, it stimulates the original FPA's search process. Since it does not need to tune any parameters and significantly simplify the population-based FPA's evolving operations, which are helpful to improve FPA's searching efficiency. To be specific, the contributions made in this work are as follows: (1) we present the mathematical formula for the sensor ontology meta-matching problem; (2) we propose a problem-specific CFPA to efficiently address the problem, which uses the compact exploitation operator and compact exploration operator to mimic FPA's evolving process and an adaptive switching probability to trade-off CFPA's exploitation and exploration; and (3) we employ CFPA on ontology alignment evaluation initiative (OAEI)'s benchmark and the task of matching sensor ontologies. The results reveal that CFPA is able to efficiently solve the sensor ontology meta-matching problem.

The rest of the paper is organized as follows: the sensor ontology meta-matching problem is defined in Section 2; CFPA is presented in Section 3 in detail; the statistical experimental results are shown in Section 4; and finally, Section 5 draws the conclusions.

## 2. Sensor Ontology Meta-Matching Problem

The ontology matcher measures two sensor concepts' similarity values with a real number in  $[0, 1]$ . The higher the similarity value, the more possible it is that two concepts are identical. In general, there are three kinds of ontology matchers, which are based on a string, linguistic, and ontology structure [22]. An ontology matcher calculates the similarity value by taking into consideration only one or two linguistic features, and thus none of them is able to ensure the result's confidence when facing different heterogeneous contexts. Usually, it is necessary to comprehensively aggregate their results, which is of help to enhance the final value's confidence. For the convenience of this work, a sensor ontology  $O$  is defined as a 3-tuple  $(C, P, R)$ , where  $C$ ,  $P$ , and  $R$  are respectively the sensor concept set, concept's property set, and the concepts' relationship set [23]. To overcome two sensor ontologies' heterogeneity issues, we need to find their entity mappings, and each correspondence is defined as 4-tuple  $(e_1, e_2, \text{simValue}, \text{rel})$ , where  $e_1$  and  $e_2$  are two ontologies' entities,  $\text{simValue}$  is their similarity value and  $\text{rel}$  is two entities' semantic relationship [24]. In this work, we aim at finding the identical sensor concepts from two ontologies, and thus, a correspondence's  $\text{rel}$  is an equivalence. Given two ontologies  $O_1$  and  $O_2$ , an ontology matcher is executed to determine their corresponding alignment, which is a set of entity correspondences [25]. In this work, the alignment is denoted by a matrix with real numbers in  $[0, 1]$  as its elements, whose rows and columns are two entity sets, and its element is two corresponding entities' similarity value.

To combine these matchers, we assign the weights for their corresponding similarity matrices and then aggregate these matrices into the final one. The sensor ontology meta-matching problem investigates how to find an optimal weight set to determine a highquality alignment [26]. Here, we model the sensor ontology metamatching problem as a singleobjective optimization problem, which takes maximizing the alignment's quality as the objective. Given a sensor alignment, the more correspondences it has and the higher the mean similarity value of all the correspondences is, the better quality it owns. Based on this, we use the following two quality metrics on an alignment  $A$ :

$$f_1(A) = \sqrt{\frac{|A|}{\max\{|O_1|, |O_2|\}}} \in [0, 1], \quad (1)$$

$$f_2(A) = \sqrt{\frac{\sum \text{sim}_i}{|A|}} \in [0, 1],$$

where  $|O_1|$ ,  $|O_2|$ , and  $|A|$  are the number of two ontologies' entities and the correspondences in the alignment and  $\text{sim}_i$  is  $i$ -th correspondence's similarity value. After that, we calculate two metrics' harmony mean to comprehensively measure the alignment's quality, which is defined as follows:

$$f(A) = \frac{2 \times f_1(A) \times f_2(A)}{f_1(A) + f_2(A)} \in [0, 1]. \quad (2)$$

On this basis, the mathematical model of the problem is defined as follows:

$$\begin{cases} \max F(W), \\ \text{s.t. } W = (w_1, w_2, \dots)^T, \\ w_i \in [0, 1] \\ \sum w_i = 1, \end{cases} \quad (3)$$

where  $w_i$  is the  $i$ -th weight of the ontology matcher's corresponding matching matrix and  $F(W)$  first uses  $W$  to aggregate all the matching matrices and then use the function  $f()$  to calculate the final matrix's corresponding alignment's quality.

### 3. Compact Flower Pollination Algorithm

FPA is inspired by the pollination of natural flowers, and its evolving process consists of two distinct operators, i.e., global pollination and local pollination, whose formulas are defined in the following equations:

$$x_i^{t+1} = x_i^t + L(x_i^t - x^*), \quad (4)$$

$$x_i^{t+1} = x_i^t + \text{rand}(0, 1)(x_p^t - x_q^t), \quad (5)$$

where  $t$  is current generation,  $x_i^t$  is  $i$ -th pollen in  $t$ -th generation,  $x_p^t$  and  $x_q^t$  are two neighbor pollens,  $x^*$  is the best pollen found, and  $L$  is the step length that draws from Levy distribution [27]. FPA's exploration and exploitation are controlled by a switching probability  $\text{sp} \in [0, 1]$ . In each generation, for each pollen, FPA generates a random number in  $[0, 1]$  and compares it with  $p$  to decide the operation on it, and after that, FPA tries to update the best pollen. Classic FPA suffers from low converging speed, and to overcome this drawback, this work proposes a CFPA, whose main components, i.e., the encoding mechanism and exploration and exploitation operators, which are presented in the following sections, respectively.

**3.1. Encoding Mechanism.** CFPA uses the gray code (GC) [28], a popular binary encoding mechanism, to encode pollen. To be specific, we use GC to encode the integers in  $[0, 100]$ , and when decoding, we normalize all the integers to obtain the corresponding weights. For example, given four ontology matchers and we need to encode four integers in a pollen, assuming 20, 20, 40, and 80, and the aggregating weights for the matching matrices are 0.125, 0.125, 0.25, and 0.5, respectively. In this work, we utilize one PV to describe a population, whose dimension is equal to the length of pollen, and its element is the probability of being 1 on the corresponding bit of the pollen. In the beginning, all PV's elements are initialized as 0.5, which is updated at the end of each generation according to the best pollen found.

Given a PV  $(0.2, 0.4, 0.6, 0.8)^T$ , generate four random numbers in  $[0, 1]$ , e.g., 0.1, 0.3, 0.5 and 0.9 since  $0.1 > 0.2$ , the

first bit of new pollen is 1; similarly, since  $0.9 > 0.8$ , the last bit of newly generated pollen is 0. When updating PV, if the value of the elite pollen is 1 (or 0), its corresponding PV's element will be increased (decreased), which can make the new pollen generated hereafter closer to the elite pollen. It is obvious that when all the probabilities are close to 1 or 0, the CFPA converges.

**3.2. Exploration and Exploitation Operators.** The exploitation operator aims at searching for particular pollen's neighboring places, while the exploitation operator tries to search in an unexplored position. The pseudocode of the exploration and exploitation operators is shown in Algorithm 1 and Algorithm 2, respectively.

Here, we introduce the exponential crossover operator (EC) [29] to implement CFPA's exploration and exploitation operators. Given two pollens, EC randomly copies a certain number of sequential bits' values from the first one to the second one. Essentially, the obtained new pollen is generated by the turbulence on its parents, which is very exploitative. With respect to the exploration operator, we use EC to mix a newly generated pollen  $\text{pollen}^{\text{new}}$  and the elite pollen  $\text{pollen}^{\text{elite}}$ , while in the exploitation, we first mix two newly generated pollens  $\text{pollen}^p$  and  $\text{pollen}^q$  to obtain the mediate pollen, then we mix it with the pollen  $\text{pollen}^{\text{new}}$ . Essentially, the exploration operator generates new pollen by moving it towards the global optima, and the exploitation operator moves the newly generated pollen to the direction determined by its neighbor pollens.

**3.3. Pseudocode of Compact Flower Pollination Algorithm.** The pseudocode of CFPA is presented in Algorithm 3. CFPA first initializes all the elements of PV as 0.5 and then uses them to initialize the elite pollen  $\text{pollen}^{\text{elite}}$ . In each generation, CFPA adaptively updates the switching probability  $\text{sp}$  and then uses it to decide whether to execute on exploration or exploitation. Here,  $\text{sp}$  is the probability of executing the exploitation operator. In the early phase, the algorithm mainly focuses on exploration, i.e.,  $\text{sp}$  is large, while in the late phase, CFPA puts the emphasis on exploitation, i.e.,  $\text{sp}$  is small. At the end of each iteration, CFPA tries to update the  $\text{pollen}^{\text{elite}}$  and PV. Finally, when reaching the maximum iteration number  $\text{max}T = 3000$ , the algorithm terminates and returns  $\text{pollen}^{\text{elite}}$ . Here, we update PV with the step that is determined by the pollen's length, and how to adaptively set the optimal step length for updating PV is one of our future research directions.

## 4. Experimental Results and Analysis

**4.1. Experimental Setup.** The benchmark of the ontology alignment evaluation initiative (OAEI) [30] and the task of matching three real sensor ontologies are used to test CFPA's performance. In Tables 1–4, we compare CFPA with five state-of-the-art sensor ontology matching techniques, i.e., compact coevolutionary algorithm (CCEA) [13], compact evolutionary tabu search algorithm (CETSA) [14], compact co-firefly algorithm (CCFA) [15], compact differential

```

(1)  $pollen^{new} = generatePollen(PV)$ ;
(2)  $int\ num = round(random(0,1) \times pollen.length)$ ;
(3)  $int\ index = 0$ ;
(4) for ( $int\ i = 0; i < pollen.length, i = i + 1$ )
(5)   if ( $index + 1 > num$ )
(6)     break;
(7)   end if
(8)   if ( $(index + 1 > pollen.length)$ )
(9)      $num = 0$ ;
(10)  end if
(11)   $pollen_i^{new} = pollen_i^{elite}$ ;
(12) end for
(13) return  $pollen^{new}$ ;

```

ALGORITHM 1: Exploration operator.

```

(1)  $pollen^{new} = generatePollen(PV)$ ;
(2)  $pollen^p = generatePollen(PV)$ ;
(3)  $pollen^q = generatePollen(PV)$ ;
(4)  $int\ num = round(random(0,1) \times pollen.length)$ ;
(5)  $int\ index = 0$ ;
(6) for( $int\ i = 0; i < pollen.length, i = i + 1$ )
(7)   if ( $index + 1 > num$ )
(8)     break;
(9)   end if
(10)  if ( $(index + 1 > pollen.length)$ )
(11)     $num = 0$ ;
(12)  end if
(13)   $pollen_i^p = pollen_i^q$ ;
(14) end for
(15)  $int\ num = round(random(0,1) \times pollen.length)$ ;
(16)  $int\ index = 0$ ;
(17) for( $int\ i = 0; i < pollen.length, i = i + 1$ )
(18)   if ( $index + 1 > num$ )
(19)     break;
(20)   end if
(21)   if ( $(index + 1 > pollen.length)$ )
(22)      $num = 0$ ;
(23)   end if
(24)    $pollen_i^{new} = pollen_i^p$ ;
(25) end for
(26) return  $pollen^{new}$ ;

```

ALGORITHM 2: Exploitation operator.

evolution algorithm (CDEA) [16], and simulated annealing particle swarm optimization (SAPSO) [17], on all testing cases in terms of recall, precision, and f-measure, respectively.

Three kinds of ontology matchers used in this work are the N-gram distance [31] (string-based ontology matcher), wordnet-based distance [32] (linguistic-based matcher), and profile-based distance [33] (structure-based ontology matcher), and the configurations of SIs are referred to in their literature. The results shown in the tables are the average of thirty independent runs.

To fairly compare with other matching techniques, we use recall, precision, and f-measure [34] to evaluate the

obtained alignments. In Table 5, we briefly describe the testing cases used in the experiment, and in Tables 1–4, the testing cases 1XX, 2XX, and 3XX are the ones starting with the numbers 1, 2, and 3, respectively.

*4.2. Statistical Experiment.* We utilize the statistical testing method *T*-test [35] to compare different competitors' performances in terms of recall, precision, and f-measure, respectively. Tables 1 and 2 show the six SI-based sensor ontology matching techniques' mean recall, precision, and f-measure, and the corresponding standard deviation on all the testing cases, and Tables 3 and 4, respectively, present the *t* value and *p* value on recall, precision, and f-measure.



```

(1) **Initialization**
(2) generation  $t = 0$ ;
(3) set all elements in PV as 0.5;
(4)  $pollen^{elite} = generatePollen(PV)$ ;
(5) **Iteration**
(6) while  $t < \max T$  do
(7)    $sp = e^{-t/\max T}$ ;
(8)   ***UpdatePollen***
(9)   if  $random(0, 1) < sp$ 
(10)     $pollen^{new} = exploration$ ;
(11)   else
(12)     $pollen^{new} = exploration$ ;
(13)   end if
(14)    $[winner, loser] = compete(pollen^{new}, pollen^{elite})$ ;
(15)   if ( $winner = pollen^{new}$ )
(16)     $pollen^{elite} = pollen^{new}$ ;
(17)   end if
(18)   **** Update PV****
(19)   for ( $i = 0$ ;  $i < PV.length$ ;  $i = i + 1$ )
(20)    if  $pollen_i^{elite} = 1$  then
(21)      $PV_i = PV_i + (1/pollen.length)$ ;
(22)    else
(23)      $PV_i = PV_i - (1/pollen.length)$ ;
(24)    end if
(25)   end for
(26)    $t = t + 1$ ;
(27) end while
(28) return  $pollen^{elite}$ ;

```

ALGORITHM 3: Compact flower pollination algorithm.

TABLE 1: Comparison among swarm intelligence-based sensor ontology matching techniques in terms of alignment quality.

Testing Case	CCEA $f(r, p)$	CETSA $f(r, p)$	CCFA $f(r, p)$	CDEA $f(r, p)$	SAPSO $f(r, p)$	CFPA $f(r, p)$
1XX	1.00 (1.00, 1.00)	0.81 (0.72, 0.90)	0.95 (0.94, 0.95)	1.00 (1.00, 1.00)	1.00 (1.00, 1.00)	1.00 (1.00, 1.00)
2XX	0.92 (0.91, 0.93)	0.72 (0.84, 0.61)	0.82 (0.79, 0.86)	0.91 (0.88, 0.94)	0.73 (0.72, 0.74)	0.94 (0.95, 0.94)
3XX	0.78 (0.81, 0.74)	0.42 (0.37, 0.48)	0.71 (0.67, 0.77)	0.89 (0.85, 0.94)	0.82 (0.85, 0.79)	0.92 (0.89, 0.96)
MMI-SSN	0.90 (0.86, 0.94)	0.92 (0.90, 0.95)	0.92 (0.90, 0.95)	0.94 (0.92, 0.95)	0.88 (0.90, 0.87)	0.95 (0.91, 0.98)
CSIRO-SSN	0.92 (0.89, 0.96)	0.94 (0.94, 0.95)	0.94 (0.94, 0.94)	0.94 (0.94, 0.95)	0.90 (0.88, 0.93)	0.96 (0.95, 0.96)
MMI-CSIRO	0.86 (0.88, 0.91)	0.90 (0.87, 0.94)	0.90 (0.87, 0.94)	0.92 (0.90, 0.93)	0.90 (0.87, 0.94)	0.94 (0.92, 0.97)
Average	0.89 (0.88, 0.92)	0.78 (0.77, 0.80)	0.87 (0.85, 0.90)	0.93 (0.91, 0.95)	0.87 (0.87, 0.87)	0.95 (0.93, 0.96)

The symbols  $f$ ,  $r$ , and  $p$ , respectively, stand for f-measure, recall, and precision.

TABLE 2: Comparison among swarm intelligence-based sensor ontology matching techniques in terms of standard deviation.

Testing Case	CCEA $f_d(r_d, p_d)$	CETSA $f_d(r_d, p_d)$	CCFA $f_d(r_d, p_d)$	CDEA $f_d(r_d, p_d)$	SAPSO $f_d(r_d, p_d)$	CFPA $f_d(r_d, p_d)$
1XX	0.01 (0.01, 0.01)	0.03 (0.02, 0.02)	0.01 (0.03, 0.02)	0.01 (0.01, 0.01)	0.01 (0.01, 0.01)	0.01 (0.01, 0.01)
2XX	0.01 (0.02, 0.01)	0.04 (0.02, 0.04)	0.02 (0.01, 0.03)	0.02 (0.02, 0.02)	0.03 (0.02, 0.03)	0.01 (0.01, 0.01)
3XX	0.02 (0.01, 0.03)	0.05 (0.02, 0.04)	0.02 (0.01, 0.03)	0.01 (0.02, 0.01)	0.02 (0.02, 0.01)	0.02 (0.01, 0.01)
MMI-SSN	0.01 (0.02, 0.02)	0.03 (0.03, 0.02)	0.03 (0.03, 0.02)	0.02 (0.03, 0.01)	0.04 (0.03, 0.02)	0.01 (0.01, 0.01)
CSIRO-SSN	0.03 (0.01, 0.01)	0.03 (0.02, 0.01)	0.03 (0.02, 0.02)	0.05 (0.03, 0.02)	0.03 (0.02, 0.02)	0.02 (0.01, 0.02)
MMI-CSIRO	0.03 (0.02, 0.01)	0.03 (0.01, 0.01)	0.02 (0.02, 0.03)	0.01 (0.01, 0.01)	0.04 (0.02, 0.04)	0.01 (0.01, 0.01)

The symbols  $f_d$ ,  $r_d$ , and  $p_d$ , respectively, stand for the standard deviation of f-measure, recall, and precision.

As can be seen from Tables 1 and 2, CFPA's results are much better than those of other SI-based sensor ontology matching techniques. Thanks to the adaptive switching

probability, CFPA is able to better trade-off the algorithm's exploitation and exploration, which not only ensures the solution's quality but also the algorithm's stability. From

TABLE 3:  $T$ -test's  $t$  value.

Testing Case	(CCEA, CFPA) $f_t(r_t, p_t)$	(CETSA, CFPA) $f_t(r_t, p_t)$	(CCFA, CFPA) $f_t(r_t, p_t)$	(CDEA, CFPA) $f_t(r_t, p_t)$	(SAPSO, CFPA) $f_t(r_t, p_t)$
1XX	0.00 (0.00, 0.00)	-180 (-375, -134)	-106 (-56, -67)	0.00 (0.00, 0.00)	0.00 (0.00, 0.00)
2XX	-42 (-53, -21)	-160 (-147, -240)	-160 (-339, -75)	-40 (-93, 0.00)	-199 (-308, -189)
3XX	-148 (-169, -208)	-278 (-697, -349)	-222 (-466, -180)	-0.40 (-53, -42)	-106 (-53, -360)
MMI-SSN	-106 (-67, -53)	-28 (-9, -40)	-28 (-9, -40)	-13 (-9, -63)	-50 (-9, -147)
CSIRO-SSN	-33 (-127, 0.00)	-16 (-13, -13)	-16 (-13, -21)	-11 (-9, -10)	-49 (-93, -31)
MMI-CSIRO	-75 (-53, -127)	-37 (-106, -63)	-53 (-67, -28)	-42 (-42, -84)	-29 (-67, -21)

The symbols  $f_t$ ,  $r_t$ , and  $p_t$ , respectively, stand for the  $t$ -value on f-measure, recall, and precision.

TABLE 4:  $T$ -test's  $p$  value.

Testing Case	(CCEA, CFPA) $f_p(r_p, p_p)$	(CETSA, CFPA) $f_p(r_p, p_p)$	(CCFA, CFPA) $f_p(r_p, p_p)$	(CDEA, CFPA) $f_p(r_p, p_p)$	(SAPSO, CFPA) $f_p(r_p, p_p)$
1XX	0.50 (0.50, 0.50)	0.001 (0.0008, 0.0023)	0.003 (0.005, 0.004)	0.50 (0.50, 0.50)	0.50 (0.50, 0.50)
2XX	0.007 (0.006, 0.015)	0.001 (0.002, 0.001)	0.001 (0.002, 0.001)	0.007 (0.003, 0.50)	0.001 (0.001, 0.001)
3XX	0.002 (0.001, 0.001)	0.001 (0.0004, 0.0009)	0.001 (0.0006, 0.001)	0.378 (0.006, 0.007)	0.003 (0.006, 0.0008)
MMI-SSN	0.003 (0.004, 0.006)	0.011 (0.035, 0.007)	0.011 (0.035, 0.007)	0.024 (0.035, 0.005)	0.006 (0.035, 0.002)
CSIRO-SSN	0.009 (0.002, 0.50)	0.019 (0.024, 0.024)	0.019 (0.024, 0.015)	0.028 (0.035, 0.031)	0.006 (0.003, 0.010)
MMI-CSIRO	0.004 (0.006, 0.002)	0.008 (0.003, 0.005)	0.006 (0.004, 0.011)	0.007 (0.007, 0.003)	0.010 (0.004, 0.015)

The symbols  $f_p$ ,  $r_p$ , and  $p_p$ , respectively, stand for the  $p$ -value on f-measure, recall, and precision.

TABLE 5: Descriptions of the ontologies in the testing cases.

Testing case	Ontology	Scale
OAIE's benchmark	Bibliographic ontology	97 entities
Real sensor ontology	CSIRO sensor ontology (CSIRO)	33,205 entities
	Semantic sensor network ontology (SSN)	32,298 entities
	MMI device ontology (MMI)	24,034 entities

Tables 3 and 4, except those testing cases where the results of CFPA and other competitors are the same, our approach outperforms other SI-based sensor ontology matching techniques on a 5% significant level. Since CFPS does not need to tune any parameters, it is more stable than other SIs. In addition, CFPA's adaptive switching mechanism and two compact evolutionary operators are able to significantly improve the algorithm's performance, which makes it efficiently search for better solutions.

## 5. Conclusion

To support communication among IoT applications, it is necessary to describe the sensor data at a semantic level. Recently, sensor ontology has become a popular knowledge modeling technique in the IoT, which is able to provide semantic meanings for diverse sensor data. However, there exists the heterogeneity issue between different sensor ontologies, which hampers IoT applications' co-operation. Sensor ontology matching is a feasible solution to this problem, which aims to find identical sensor concepts at the semantic level. This work investigates a sensor ontology meta-matching problem, which aims to indirectly optimize the sensor ontology alignment's quality by tuning the weights to aggregate different ontology matchers. Inspired

by the success of FPA in the IoT domain, we further propose a CFPA to efficiently address the sensor ontology meta-matching problem. In particular, we introduce the compact encoding mechanism to improve the algorithm's searching efficiency and the adaptive switching parameter to trade-off the algorithm's exploitation and exploration. The experiment compares CFPA with five state-of-the-art sensor ontology matching techniques based on SIs, and the experimental results show that CFPA outperforms other SI-based sensor ontology matching techniques.

In the future, we will further improve CFPA to match the largescale sensor ontologies, especially at the instance level. We are also interested in further improving CFPA to match the specific ontologies in the biomedical domain and geographical domain. When dealing with largescale matching tasks, efficiency-improving strategies should be introduced, such as ontology partition and correspondence pruning. Also, the problem that how to choose the suitable background knowledge base to distinguish the complex entity correspondence also needs to be addressed.

## Data Availability

The data used to support this study can be found in <http://oaei.ontologymatching.org>.

## Conflicts of Interest

The authors declare that they have no conflicts of interest in the work.

## Acknowledgments

This work was supported in part by the Improvement Project of Basic Ability for Young and Middle-aged Teachers in Guangxi Universities (No. 2017KY0535), the Science Research Foundation for High-level talents of Yulin Normal University (No. G2021ZK17), and the Construction of China-Ukraine Joint Carbon Black Research Center (I), (No. 2022HXZK01).

## References

- [1] P. Pande and A. Padwalkar, "Internet of Things—a future of Internet: a survey," *International Journal of Advance Research in Computer Science and Management Studies*, vol. 2, pp. 354–361, 2014.
- [2] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, vol. 284, no. 5, pp. 34–43, 2001.
- [3] H. Neuhaus and M. Compton, "The semantic sensor network ontology," in *Proceedings of the AGILE Workshop on Challenges in Geospatial Data Harmonisation*, pp. 1–33, Hannover, Germany, 2009.
- [4] M. Compton, P. Barnaghi, L. Bermudez et al., "The SSN ontology of the W3C semantic sensor network incubator group," *Journal of Web Semantics*, vol. 17, pp. 25–32, 2012.
- [5] C. Rueda, N. Galbraith, and R. A. Morris, "The MMI device ontology: enabling sensor integration," in *Proceedings of the AGU Fall Meeting Abstracts*, pp. 1–35, San Francisco, CA, USA, 2010.
- [6] P. Shvaiko and J. Euzenat, "Ontology matching: state of the art and future challenges," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 1, pp. 158–176, 2011.
- [7] S. Fernandez, I. Marsa-Maestre, J. Velasco, and B. Alarcos, "Ontology alignment architecture for semantic sensor web integration," *Sensors*, vol. 13, no. 9, pp. 12581–12604, 2013.
- [8] C. Touati, M. Benaissa, and Y. Lebbah, "An efficient model for extracting an optimal alignment with multiple cardinalities in ontology alignment," *International Journal of Metadata Semantics and Ontologies*, vol. 11, no. 2, pp. 71–81, 2016.
- [9] X. Xue and Q. Huang, "Generative adversarial learning for optimizing ontology alignment," *Expert Systems*, vol. 2022, pp. 1–12, 2022.
- [10] S. Berlemont, G. Lefebvre, S. Duffner, and C. Garcia, "Class-balanced siamese neural networks," *Neurocomputing*, vol. 273, pp. 47–56, 2018.
- [11] X. Zhu and A. B. Goldberg, "Introduction to semi-supervised learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 3, no. 1, pp. 1–130, 2009.
- [12] Y. Liu, G. G. Yen, and D. Gong, "A multimodal multiobjective evolutionary algorithm using two-archive and recombination strategies," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 4, pp. 660–674, 2018.
- [13] X. Xue and J.-S. Pan, "A compact co-evolutionary algorithm for sensor ontology meta-matching," *Knowledge and Information Systems*, vol. 56, no. 2, pp. 335–353, 2018.
- [14] D. Costa, "An evolutionary tabu search algorithm and the NHL scheduling problem," *INFOR: Information Systems and Operational Research*, vol. 33, no. 3, pp. 161–178, 1995.
- [15] X. S. Yang and X. He, "Firefly algorithm: recent advances and applications," *International Journal of Swarm intelligence*, vol. 1, no. 1, pp. 36–50, 2013.
- [16] J. Liu and J. Lampinen, "A fuzzy adaptive differential evolution algorithm," *Soft Computing*, vol. 9, no. 6, pp. 448–462, 2005.
- [17] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.
- [18] X.-S. Yang, "Flower pollination algorithm for global optimization," in *Proceedings of the International Conference on Unconventional Computing and Natural Computation*, pp. 240–249, Espoo, Finland, 2012.
- [19] T.-T. Nguyen, J.-S. Pan, and T.-K. Dao, "An improved flower pollination algorithm for optimizing layouts of nodes in wireless sensor network," *IEEE Access*, vol. 7, pp. 75985–75998, 2019.
- [20] J. Pan, J. Zhuang, and H. Luo, "Multi-group flower pollination algorithm based on novel communication strategies," *Journal of Internet Technology*, vol. 22, no. 2, pp. 257–269, 2021.
- [21] G. R. Harik, F. G. Lobo, and D. E. Goldberg, "The compact genetic algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 287–297, 1999.
- [22] J. Martinez-Gil and J. F. Aldana-Montes, "Evaluation of two heuristic approaches to solve the ontology meta-matching problem," *Knowledge and Information Systems*, vol. 26, no. 2, pp. 225–247, 2011.
- [23] L. Otero-Cerdeira, F. J. Rodríguez-Martínez, and A. Gómez-Rodríguez, "Ontology matching: a literature review," *Expert Systems with Applications*, vol. 42, no. 2, pp. 949–971, 2015.
- [24] W. Hu and Y. Qu, "Falcon-AO: a practical ontology matching system," *Journal of Web Semantics*, vol. 6, no. 3, pp. 237–239, 2008.
- [25] Y. R. Jean-Mary, E. P. Shironoshita, and M. R. Kabuka, "Ontology matching with semantic verification," *Journal of Web Semantics*, vol. 7, no. 3, pp. 235–251, 2009.
- [26] N. Ferranti N, S. S. R. F. Soares, and J. F. de Souza, "Meta-heuristics-based ontology meta-matching approaches," *Expert Systems with Applications*, vol. 173, pp. 1–15, 2021.
- [27] F. J. O'Reilly and R. Rueda, "A note on the fit for the Levy distribution," *Communications in Statistics-Theory and Methods*, vol. 27, no. 7, pp. 1811–1821, 1998.
- [28] R. Doran, "The gray code," *Journal of Universal Computer Science*, vol. 13, no. 11, pp. 1573–1597, 2007.
- [29] S.-Z. Zhao and P. N. Suganthan, "Empirical investigations into the exponential crossover of differential evolutions," *Swarm and Evolutionary Computation*, vol. 9, pp. 27–36, 2013.
- [30] M. Achichi, M. Cheatham, and Z. Dragisic, "Results of the ontology alignment evaluation initiative 2016," in *Proceedings of the 11th International Workshop on Ontology Matching Co-Located with the 15th International Semantic Web Conference (ISWC 2016)*, pp. 73–129, Kobe, Japan, 2016.
- [31] G. Kondrak, "N-gram similarity and distance," *String Processing and Information Retrieval*, Springer, Berlin, Germany, pp. 115–126, 2005.
- [32] G. A. Miller, "WordNet," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.
- [33] G. Acampora, V. Loia, and A. Vitiello, "Enhancing ontology alignment through a memetic aggregation of similarity measures," *Information Sciences*, vol. 250, pp. 1–20, 2013.
- [34] G. Hripcsak and A. Rothschild, "Agreement, the f-measure, and reliability in information retrieval," *Journal of the American Medical Informatics Association*, vol. 12, no. 3, pp. 296–298, 2005.
- [35] D. Semenic, "Tests and measurements: the T-test," *National Strength & Conditioning Association Journal*, vol. 12, no. 1, pp. 36–37, 1990.

## Research Article

# Improved Optimal Path Finding Algorithm of Multistorey Car Park Based on MCP Protocol

Zhendong Liu <sup>1</sup>, Dongyan Li <sup>1</sup>, Xi Chen,<sup>1</sup> Xinrong Lv,<sup>1</sup> Yurong Yang,<sup>1</sup> Ke Bai,<sup>1</sup> Mengying Qin,<sup>1</sup> Zhiqiang He,<sup>1</sup> Xiaofeng Li,<sup>1</sup> and Qionghai Dai<sup>2</sup>

<sup>1</sup>School of Computer Science and Technology, Shandong Jianzhu University, Jinan 250101, China

<sup>2</sup>Department of Automation, Tsinghua University, Beijing 100084, China

Correspondence should be addressed to Zhendong Liu; liuzd2000@126.com

Received 9 January 2022; Accepted 21 February 2022; Published 9 March 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Zhendong Liu 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.

To address the urgent issue of car owners wasting a lot of time because they cannot find empty parking spaces in the largely multistorey car park, the paper conceives a solution including algorithm ParkIG. This scheme uses the infrared photoelectric switch to monitor the parking space status. At the same time, we define a special communication protocol named MCP to analyze the parking space data when analyzing the data transmitted by the photoelectric switch. Algorithm ParkIG is used in finding the nearest empty parking space and optimal path planning. Based on the MCP protocol and navigation system, the algorithm ParkIG initially identifies the layer of multistorey where the car is positioned. Our algorithm ParkIG greatly reduces the time and space size of the search domain and update domain when searching for vacant parking spaces in optimal path planning and ends the algorithm running to avoid a lot of computations that are not essential when the algorithm found an optimal path from the entrance to empty of parking space. Simulation experiments show that the performance of our scheme has been significantly improved, such as the time that it takes for the car owner to search for a vacant parking space is decreased by about 25.8% and reduced the space size of search domain and update domain by 70% compared to the algorithm Dijkstra.

## 1. Introduction

With the rapid development of the economy, private car travel is becoming increasingly common. The increased number of vehicles not only puts a strain on traffic but also causes a slew of problems for car owners [1]. People's travel is increasingly inextricably linked to automobiles, so they will inevitably encounter parking during travel [2, 3]. In office buildings or shopping malls with heavy traffic, especially during the peak parking period, the problem of parking difficulties is more prominent. The main reasons for the "difficult parking" problem are the following: current urban parking spaces are in short supply, and more importantly, people can obtain less valuable parking spot info in the process of finding parking spaces [4]. This means that there is currently short of a parking space information man-

agement platform to provide reasonable parking guidance to vehicle drivers.

In outdoor car parks, there are already many technologies that can find and navigate empty parking spaces and guide car owners to find appropriate empty parking spaces to meet their parking demands [5, 6]. However, for some indoor car parks, there is no suitable solution due to technical limitations. According to the survey, the current popular map navigation can only reach the entrance of the car park and cannot see the empty parking spaces inside the car park [7]. The previous path planning algorithms can only be carried out in the same layer, which is more difficult for multistorey car parks. Indoor navigation technology and outdoor navigation technology are essentially the same, and both require three types of technical support: indoor positioning technology, indoor map, and route planning technology [8].

**Input:** reserve the first  $K$  free parking spaces as candidate empty parking spaces,  
**Output:** The route distance and route of the nearest empty parking space.  
**Begin**

1. Use IPS technology to locate the location information  $I$  of the entrance of the parking lot, locate the vehicle's location information  $I_{car}$  in real time, and the initial number of layers of the vehicle  $N_{layer} \leftarrow 0$ ;
2. if the vehicle is on the ground and  $I = I_{car}$
3.     then  $N_{layer} \leftarrow N_{layer} + 1$ ;
4. elseif the vehicle is underground and  $I = I_{car}$
5.     then  $N_{layer} \leftarrow N_{layer} - 1$ ;
6. end if
7. Load  $N_{layer}$  layer information;  $P \leftarrow$  Free parking space;
8.  $W \leftarrow$  the Manhattan distance between the entrance, intersection, and the candidate's empty parking space.; // The adjacency matrix
9.  $P \leftarrow$  {The first  $k$  parking spaces with the smallest Euclidean distance from the entrance in  $P$ };
10.  $S[1 \sim K] \leftarrow 0$ ;  $H[1 \sim K] \leftarrow \emptyset$ ; // Path distance  $S$ , path  $H$  of candidate parking spaces
11. for  $i=1$  to  $K$  do
12.      $S_i, H_i \leftarrow$  ParkD ( $P_i, W$ );
13. end for
14.  $S \leftarrow \min\{S\}$ ;  $I \leftarrow \text{index}(\min\{S\})$ ;  $H \leftarrow H_I$ ;
15. Return  $S, H$

**End**

ALGORITHM 1: ParkIG( $K$ ).

**Input:** Candidate vacant parking space  $P$ , an adjacency matrix  $W$  formed by the Manhattan distance between the entrance, intersection, and the candidate vacant parking space.  
**Output:** The path distance and path from the entrance to the candidate vacant parking space  $P$ .  
**Begin**

1.  $V \leftarrow$  entrance,  $n$  intersections, and candidate parking spaces; //a total of  $n+2$  nodes;
2. initialize mark set  $T[1 \sim n+1] \leftarrow 0$ , adjacent node set  $\text{Adj}(V_i)$ , path distance set  $D(V)$ , used to record the set  $L(V)$  of the previous node in the path;
3. while  $T(n+1) \neq 1$  do
4.      $m \leftarrow \{m | T(m) = 0 \text{ and } \min\{D(m)\}\}$ ;  $T(m) \leftarrow 1$ ;
5.     For each  $i \in \text{Adj}(m)$  and  $T(i) \neq 1$  do
6.          $D(i) \leftarrow \min\{D(i), D(m) + W(i, m)\}$ ;
7.         if  $D(m) + W(i, m) < D(i)$ ,  $L(i) \leftarrow m$ ;
8.     end for
9. end while
10.  $H \leftarrow$  backtracking according to  $L(n+1)$  to find the optimal path from the entrance to the candidate parking space;
11. Return  $D(n+1), H$

**End**

ALGORITHM 2: ParkD ( $P, W$ ).

Many car owners often circle the car park to park, but they never find a suitable empty parking space. This brings great trouble to the car owners. For car park managers, in the case of heavy traffic, they can only direct the car owners to park in person, and even the managers do not know where there are vacant parking spaces [9]. This situation not only greatly discounts the parking efficiency of car owners but also is not conducive to the work of car park managers.

This paper gives a series of solutions to this type of problem. Each parking place has an infrared photoelectric switch to gather parking data, and the MCP protocol is configured for data transfer. Simultaneously, the algorithm ParkIG is built and employed while planning the path from the entry

to the empty parking spot. In addition, our solution contains a set of car park navigation systems that are convenient for users to park and the management of the car park by the administrator. The car owner may utilize the empty parking spot navigation applet to find an empty parking place and park quickly, increasing the car owner's parking efficiency. The car park manager may also observe and manage the parking places in the car park using the multistorey parking lot management system, which makes the managers' jobs easier.

## 2. Materials and Methods

2.1. *Introduction to Algorithm ParkIG.* Algorithm Dijkstra is a classic single-source shortest route algorithm. As a greedy



strategy, it is often used in path planning problems [10]. Compared with other path planning algorithms, such as algorithm Floyd, algorithm ant colony, and algorithm A\* (A Star) [11–13], algorithm Dijkstra has certain advantages in both time complexity and space complexity. And for car park path planning, we only need to find the route from the outset to the destination and do not care about other paths [14]. Dijkstra is a single-source shortest path algorithm, so it is more suitable for solving car park path planning problems [15]. In addition, the algorithm Dijkstra is used to determine the shortest path, and other conditions such as the intersection of the car park lanes, parking spots, and their occupancy are configured. The empty parking space resources can be used more efficiently and with higher accuracy [16]. However, the search for empty parking spaces in a car park only requires the path and path distance from the outset to the parking space, while the traditional algorithm Dijkstra will compute the route from the outset to all the other dots, especially when the car park is very large and the number of nodes will be many. It will cause many unnecessary calculations and waste a lot of time [17]. Therefore, it is necessary to improve the algorithm Dijkstra to meet the demand of finding the nearest empty parking space in the car park [18, 19].

The traditional algorithm Dijkstra is used in the process of finding empty parking spaces in the car park. There will be a lot of redundant calculations, especially when the car park is relatively large; the time wasted will be very long, which will not improve the parking efficiency of the car owner [20]. Furthermore, the Dijkstra algorithm is unsuitable for multistorey parking garages. The car owner enters from the entrance of the car park, so the empty parking space closest to the entrance can be prioritized given the Euclidean distance. In an environment like a car park, the number of adjacent intersections at each intersection is generally no more than 4. The search domain can be modified to reduce the time complexity. In addition, determining which floor the vehicle is on is a key part of a multistorey car park.

Based on these ideas, this paper proposes the algorithm ParkIG; the detailed introduction of the algorithm is as follows: first, use IPS technology to locate the vehicle and determine which floor it is on. After the vehicle goes on a certain floor, load the parking lot layout of that floor. The entrance of this floor is taken as the starting point, and  $K$  parking spaces with Euclidean distance closest to the European-style entrance ( $K$  value can be determined according to the actual size of the car park) are selected as the endpoint. For each of the  $K$  empty parking spaces, an undirected weight map is formed together with the starting point and intersection, and the weight is the Manhattan distance among each node. When algorithm Dijkstra is applied to the undirected weighted graph, the concepts of the label set and adjacent node set are introduced to reduce the search domain and update domain from  $n$  to less than 4, which decreases the cost of time to a certain extent. In the ParkIG algorithm, for empty parking spots and intersections, the midpoint of parking spots is selected to calculate the distance and other data. The entrance, intersection, and candidate empty parking space are defined as nodes, and the

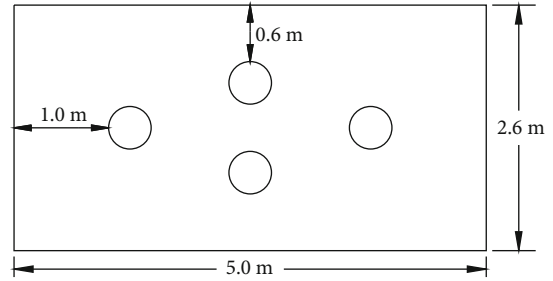


FIGURE 1: Distribution of four infrared photoelectric switches in parking spaces.

collection of adjacent nodes is defined as the collection of adjacent nodes of each node. The tag set is defined to mark whether the nearest route and route distance of the node are determined by the algorithm. If it is “0,” it means it is not determined, and “1” indicates that the path is determined. The path distance set defines the route distance of each node from the outset. The previous node set is used to record the previous node of the currently determined path. The comprehensive algorithm procedures are as follows:

The algorithm ParkIG will call the algorithm ParkD, which adopts the thought of the algorithm Dijkstra and improves it. Algorithm ParkD realizes the optimal path planning from the vehicle to each vacant parking space. As a result, you simply need to use the algorithm ParkD for each candidate’s available parking space to determine the closest distance and optimal path.

The time complexity of the algorithm ParkIG is  $O(n^3)$ , but due to the particularity of car park routes, the count of adjacent points of nodes is generally not more than 4 in the process of cyclic finding the nearest adjacent points. And because  $K$  is a constant, the choice of  $K$  is generally not exceeding 10 (depending on the size of the car park). The algorithm ends directly after finding the optimal path from the entrance to the parking space with no vehicles; that is, the maximum count of cycles is  $n$ . After adding these constraints, the overall time consumed in the algorithm is no more than  $4Kn$ , and the overall performance has been greatly improved.

**2.2. Car Park Hardware Deployment and MCP Protocol.** At present, there are many methods for obtaining the situation of parking spots. For example, the GPP and PGS2 systems use ultrasound, while the Siemens system is a parking sensor placed on the ground [21]. Considering that this system is mainly aimed at large- and medium-sized car parks, and ultrasonic sensors are very sensitive to temperature changes and extreme air, it is one of the wisest methods to monitor vehicles through infrared sensors [22]. Therefore, out of considerations such as funding and accuracy, diffuse reflection infrared photoelectric switches are used at the car park, and four infrared photoelectric switches are arranged in each parking space. Figure 1 depicts the location of the switches in the parking slots. In this figure, the length and width of each parking space are 5.0 meters and 2.6 meters, respectively. The four photoelectric switches are all on the centerline of the parking space. Two photoelectric switches are

TABLE 1: The table of byte descriptions in the MCP protocol.

Type	Province	City	County/district	Community/shopping market	Car park	Data collector number	Parking space data	Check code
Byte count	1	2	3	4-5	6	7	8-15	16

TABLE 2: Parking data table of a single byte in MCP protocol.

The upper 4 bits				The lower 4 bits			
D7	D6	D5	D4	D3	D2	D1	D0
One parking space information				One parking space information			

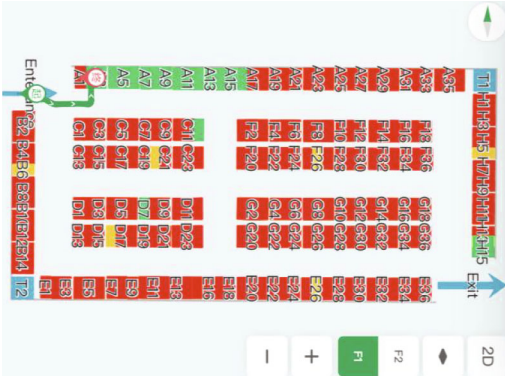


FIGURE 2: Examples of parking recommendations and real-time navigation.

60 cm from the nearest long side, and the remaining two photoelectric switches are 100 cm from the nearest wide side.

The scheme controller uses a single-chip microcomputer to output the parking status information collected by the infrared photoelectric switch. Each photoelectric switch returns one bit of data, and the overwritten data of the photoelectric switch is “0”; otherwise, it is “1.” The communication module adopts a 4G/5G wireless communication module to ensure the data transmission between the car park and the multistorey car park management system [23]. The MCP protocol is based on the distribution of photoelectric switches in the parking spaces. Because there are four infrared photoelectric switches, the management system can know whether the parking space is an empty parking space through the returned data. If it is not an unoccupied parking spot, the management system can determine whether the vehicle parked in the parking space is parked in a standard manner.

As shown in Table 1, the MCP protocol is a new multistorey car park protocol that specifies the following: a total of 16 bytes of data are transmitted, the first six bytes represent provinces, cities, counties, communities, and car parks, the seventh byte represents the data collector number, and the eighth byte to the 15th byte represents the data of parking spaces collected by the collector. For multistorey car parks, each floor has a specific car park number as the basis for judgment. As shown in Table 2, the high and low four digits of each byte represent a parking space, respectively, in the parking data transmitted; that is, the status of 16 parking spaces can be transmitted in one transmission. Such as

Equation (1), if the four-digit data is “1111,” it means an empty parking space, if it is “0000,” it means parking is regulated, and the rest of the data type means parking is not regulated. The last byte is the cumulative checksum.

$$\text{Status} = \begin{cases} \text{Empty parking space,} & \text{the four-digit data} = 1111, \\ \text{Regulate parking,} & \text{the four-digit data} = 0000, \\ \text{Irregular parking,} & \text{others.} \end{cases} \quad (1)$$

**2.3. Multistorey Car Park Management System.** The parking space management system for the car park is used by the car park administrator, including the two roles of ordinary car park administrator and senior administrator. Ordinary car park administrators can see the data-receiving page, manual entry page, parking space data report statistics page, rent management page, user management page, and personal information modification page. The data-receiving page is used to receive real-time data from the car park. The manual entry page can manually enter data in some fault situations. Parking space data report statistics page can be statistics of a variety of parking data reports, including a statistical table of vehicle entry and exit, a parking information statistical table of parking spaces, a statistical table of vehicle entry and exit time intervals, a statistical table of empty parking spaces within a certain period of time, a statistical table of irregular parking status within a certain period of time, a statistical table of standardized parking status within a certain period of time, a daily report of vehicle entry and exit records, a weekly report of vehicle entry and exit records, a monthly report of vehicle entry and exit records, and an annual report of vehicle entry and exit records. The rent management page is used to manage the rent of temporary parking and long-term parking. The user management page gets used to managing resident users. The personal information modification page is used for modifying personal information and login password. Senior administrators can see the ordinary car park administrator management page and parking space data report statistics page. The ordinary car park administrator management page is used to manage ordinary car park administrators. The parking space data report statistics page has similar functions to ordinary car park administrators. The difference is that senior administrators can manage all car park data.

**2.4. Navigation Mini Program for Empty Parking Spaces in Multistorey Car Park.** The user terminal includes a page for finding the nearest available parking space as well as a page for real-time navigation. The car owner can view the real-time situation of the parking space in the car park by scanning the QR code of the WeChat applet at the entrance.

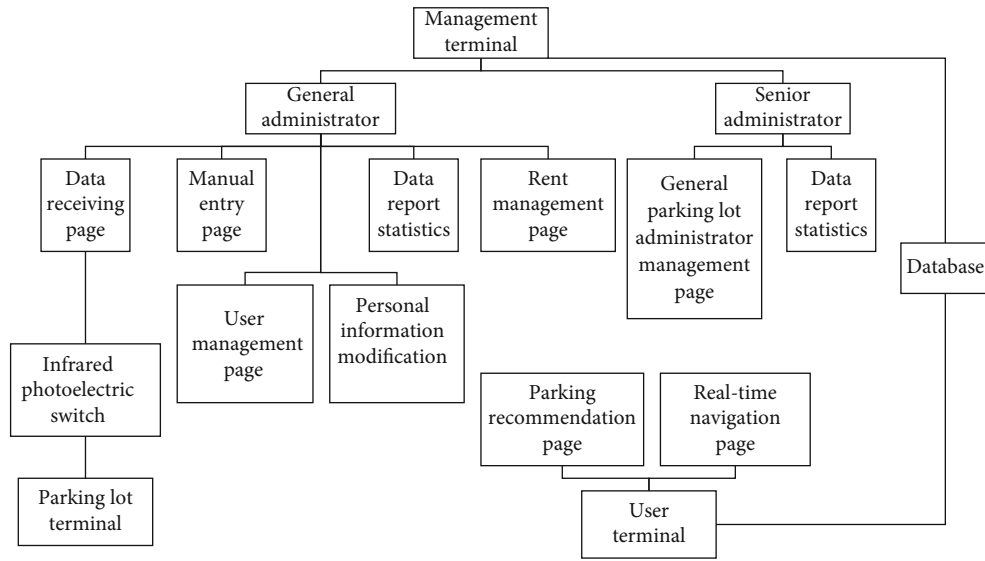


FIGURE 3: The diagram of system architecture.

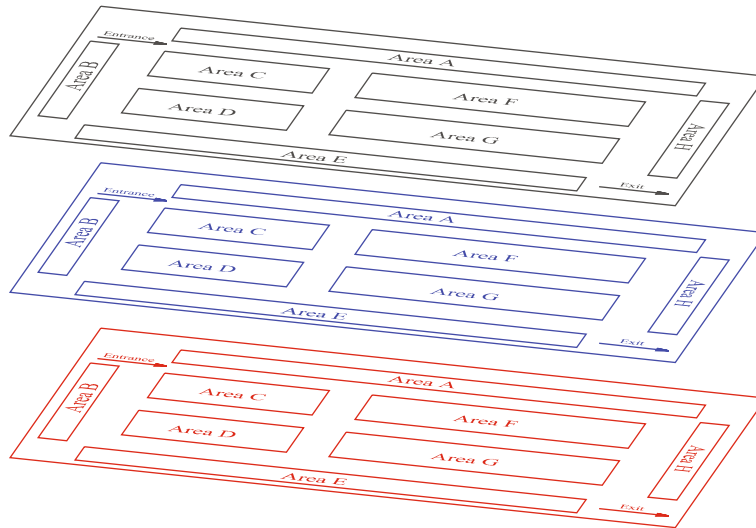


FIGURE 4: Layout of parking spaces in the multistorey car park.

As shown in Figure 2, the system will also recommend the nearest vacant parking space based on the algorithm ParkIG and implement real-time navigation to guide the user to the nearest vacant parking space to park.

**2.5. System Architecture Design.** The entire system design is mainly divided into three parts: the car park terminal, the parking space management system (management terminal), and the empty parking space navigation applet (user terminal). According to the above algorithm ParkIG, car park hardware deployment, MCP protocol, multistorey car park management system, and navigation miniprogram, the overall architecture of the solution is shown in Figure 3.

### 3. Results and Discussion

**3.1. Environmental Model Building of Multistorey Car Park.** Unlike urban roads, the roads of car parks are often not so

complicated to facilitate car owners to park [24, 25]. Figure 4 shows the layout of a common car park. The entrance and exit of the car park are set up separately, and there is only one. The parking spaces are distributed regularly, which is suitable for discussing path planning issues. According to the road environment of the car park, the entrance, intersection, and empty parking spaces are defined as nodes, and the distance between the intersection and the intersection is defined as the weight value to form a weighted undirected graph.

Many factors should be considered when empty parking spaces are recommended in the car park: the factors of the parking spot itself, the distance between the empty parking space and the pedestrian elevator, the distance among the unoccupied parking spot and the outlet, and the distance among the empty parking spot and the entrance [26]. However, priority was given to the factor of path distance from the entrance to the parking spot where no car is parked

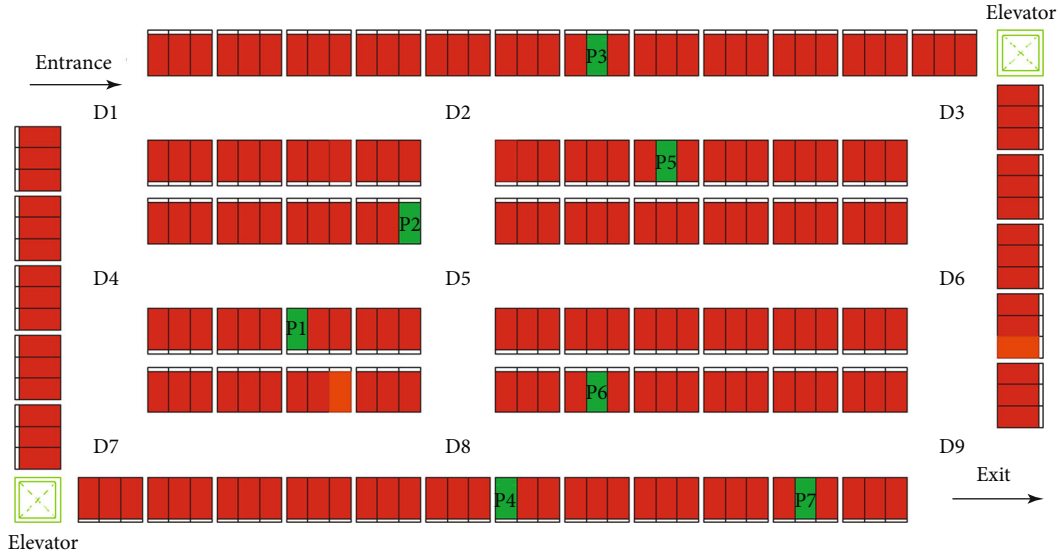


FIGURE 5: Occupancy of parking spaces at a certain time.

TABLE 3: Comparison of two algorithm results.

	Recommended parking space	Path	Path distance (m)	Time-consuming (ms)
Dijkstra	P1	M $\rightarrow$ D1 $\rightarrow$ D4 $\rightarrow$ P1	45.72	88.79
ParkIG	P1	M $\rightarrow$ D1 $\rightarrow$ D4 $\rightarrow$ P1	45.72	55.82

because of the urgent need of car owners to park. Set the empty parking space of the car park as  $P(i)$ , the length from the entrance to the empty parking space is  $S(i)$ , and then, the minimum path length is  $\min \{S(i)\}$ .

**3.2. Experimental Environment.** In the Windows operating system, PyCharm 2019.3.3 is used to write programs to simulate path planning, and MySQL database is used to store the relative coordinates of the parking space and whether the parking space is empty or not. Navicat Premium 12 was used to facilitate the operation of the database.

**3.3. Analysis and Simulation of ParkIG Algorithm.** Based on the algorithm ParkIG mentioned above, an example was selected for analysis. The hardware collector collected the car park occupation at a certain time, as shown in Figure 5. In the picture, the red parking spot represents that the parking spot has vehicles parked and the parking is regulated, the green parking space means that there is no parking in the parking space, and the orange parking spot means that the car is parked irregularly in the parking spot.

The algorithm ParkIG starts at the entrance and ends at  $K$  candidate parking spaces that may be the optimal empty parking spaces. When  $K = 5$  is set here, the results obtained by the two algorithms on the same test data are the same. Both algorithms can successfully find the nearest empty parking space and plan the optimal route. However, the algorithm ParkIG takes less time than the traditional Dijkstra algorithm. As can be seen from Table 3, when the park-

ing space is in this state, the time it takes to run the algorithm has been reduced by about 37%.

As shown in Figure 6, the same experiment was conducted for the other parking spaces in other states, and it was found that the results of the two algorithms were the same each time, and the running time of the algorithm ParkIG was relatively less. The algorithm ParkIG reduces the average time by about 25.8%.

In the same situation, we use the algorithm ParkIG and the algorithm Dijkstra to search for empty parking spaces and route planning and count the size of the update domain and search domain of each node. The results are shown in Figure 7. We have counted the data of 8 nodes. In any case, the size of the update field and search field of the algorithm Dijkstra is 10, but the algorithm ParkIG is between 2 and 5, which reduces invalid operations by 70% on average.

## 4. Conclusions

This paper proposes a set of complete solutions for the defects of the existing multilevel parking navigation technology. The scheme includes the layout control of the infrared photoelectric switch at the car park, the transmission protocol between the parking lot and the administrator, various statistical reports of the administrator, and the optimal empty parking space recommendation and route guidance based on the algorithm ParkIG. It not only greatly facilitates the car park management and statistics of the car park manager but also greatly facilitates the

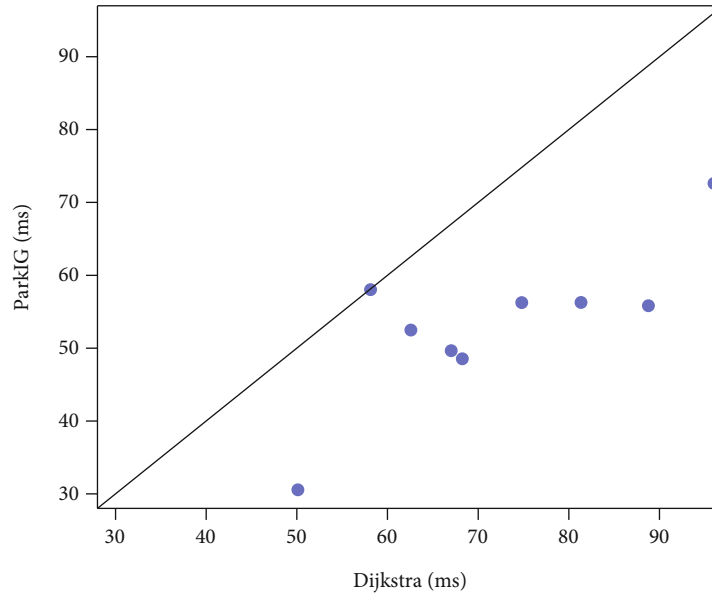


FIGURE 6: Comparison of the time spent by the two algorithms.

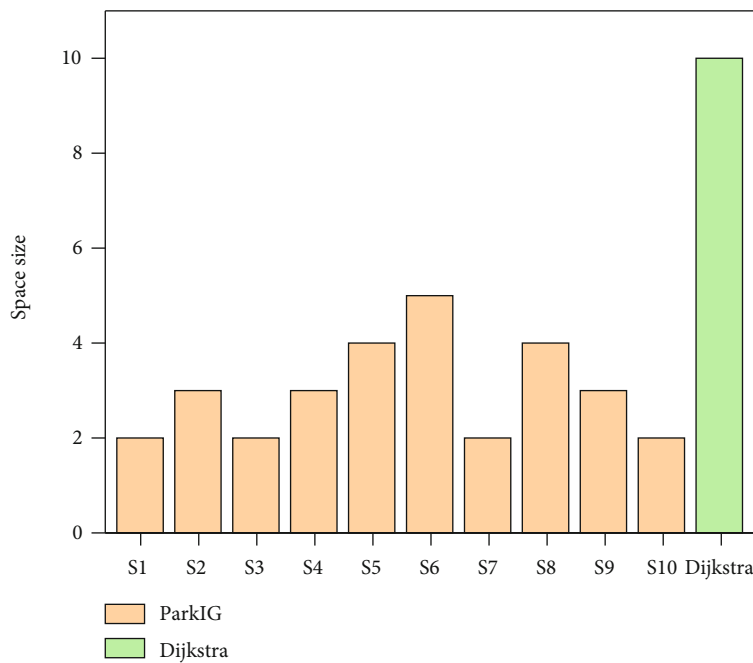


FIGURE 7: Comparison of search domain and update domain space size.

parking of the car owner. We also targeted a simulation for a classic car park, and the simulation results are achieved the expected level.

The main innovation of this paper is the arrangement of a certain regular infrared photoelectric switch to monitor the situation of the parking slots in real-time, and the MCP protocol is proposed to analyze the data transmitted by the hardware. Based on the MCP protocol and real-time parking space data, the algorithm ParkIG is proposed to achieve the search and path planning of empty parking spaces in multi-storey car parks. Thus, the efficiency of route planning is improved to ensure a good parking experience for car

owners, which has high practical application value. In addition, it also visualizes and analyzes the data in the car park, which is convenient for the management of the car park administrator.

Nevertheless, we still need to improve some areas in this paper:

- (1) Because the car owners are recommended for empty parking slots at the entrance and the path is calculated, we cannot predict the changes after the owner enters the car park, nor can we make adjustments based on the real-time situation



- (2) For very large car parks, this article does not provide further proof to ensure the effectiveness of the system, so the system is currently only suitable for small- and medium-sized car parks
- (3) In this paper, the weight calculation method is relatively simple, which needs further improvement and strict comparison with other methods, such as the predictive control method based on the neural network [27]
- (4) Regarding the parking payment module, the automatic payment is not completely realized in this paper [28]. The function of payment can be added to the user miniprogram, so that users do not need to pay at the exit, which greatly improves the user's parking experience [29]. Parking will become increasingly smarter in the future as smart parking solutions are continuously optimized

### Data Availability

All data are available within the article.

### Conflicts of Interest

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

### Authors' Contributions

Zhendong Liu, Dongyan Li, Xi Chen, Xinrong Lv, and Yurong Yang contributed equally to this work.

### Acknowledgments

The paper was greatly supported by the NNSF (National Natural Science Foundation of China), with Grant Nos.61672328 and 61672323, and the research is also supported by the Science and Research Plan of Luoyang Branch of Henan Tobacco Company (No. 2020410300270078).

### References

- [1] M. Saberi, H. Hamedmoghadam, M. Ashfaq et al., "A simple contagion process describes spreading of traffic jams in urban networks," *Nature Communications*, vol. 11, no. 1, article 1616, 2020.
- [2] L. Wang, J. Chen, X. Cao, J. Chen, and C. Zhang, "Vehicle delay model applied to dynamic and static traffic impact analysis of large parking lots," *Applied Sciences*, vol. 11, no. 20, p. 9771, 2021.
- [3] M. Pereda, J. Ozaita, I. Stavrakakis, and A. Sánchez, "Competing for congestible goods: experimental evidence on parking choice," *Scientific Reports*, vol. 10, no. 1, article 20803, 2020.
- [4] C. Lai, Q. Li, H. Zhou, and D. Zheng, "SRSP: a secure and reliable smart parking scheme with dual privacy preservation," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10619–10630, 2021.
- [5] G. W. Yoon, J. B. Kim, and B. K. Kim, "An efficient urban outdoor localization and navigation system for car-like mobile robots," *Journal of Institute of Control, Robotics, and Systems*, vol. 19, no. 8, pp. 745–754, 2013.
- [6] Y. Tazaki and Y. Yokokohji, "Outdoor autonomous navigation utilizing proximity points of 3D Pointcloud," *Journal of Robotics and Mechatronics*, vol. 32, no. 6, pp. 1183–1192, 2020.
- [7] T. Lei and Y. Chu, "Analysis on the current development of indoor navigation technology," *Modern Trade Industry*, vol. 41, no. 24, p. 153, 2020.
- [8] Z. Jiang, Y. Zhou, L. Wei, and X. Lu, "Research and realization of key technologies of indoor navigation system," *Computer Programming Skills and Maintenance*, vol. 16, pp. 26–28, 2017.
- [9] Z. Zhang, "Investigation and analysis of the status quo of parking lot planning and management in Handan City," *Technological Wind*, vol. 2018, no. 9, pp. 163–166, 2018.
- [10] M. Li, F. Zhang, and J. Fang, "Optimal path solution based on dijkstra algorithm," *Frontiers in Economics and Management*, vol. 2, no. 6, pp. 170–176, 2021.
- [11] J. Wang, Y. Sun, Z. Liu, P. Yang, and T. Lin, "Route planning based on Floyd algorithm for intelligence transportation system," in *2007 IEEE International Conference on Integration Technology*, pp. 544–546, Shenzhen, China, 2007.
- [12] M. F. Gerard, G. Stegmayer, and D. H. Milone, "Metabolic pathways synthesis based on ant colony optimization," *Scientific Reports*, vol. 8, no. 1, article 16398, 2018.
- [13] S. Wang, G. Tan, X. Jiang, and C. Su, "Mobile robot path planning based on improved a~\* algorithm," *Computer Simulation*, vol. 38, no. 9, pp. 386–389, 2021.
- [14] J. Zhao, Q. Wu, J. Chen, and Y. Huang, "Parking, intelligent parking system," *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, vol. 1, pp. 2262–2267, 2019.
- [15] S. Wang and A. Li, "Multi-adjacent-vertexes and multi-shortest-paths problem of Dijkstra algorithm," *Computer Science*, vol. 41, no. 6, pp. 217–224, 2014.
- [16] T. Fu, P. Liu, K. Liu, and P. Li, "Privacy-preserving vehicle assignment in the parking space sharing system," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8862652, 13 pages, 2020.
- [17] H. Wang, L. Zhu, and J. Wang, "Research on path planning of parking system based on Dijkstra-ant colony algorithm," *Journal of Engineering Design*, vol. 23, no. 5, pp. 489–496, 2016.
- [18] D. U. Pizzagalli, S. F. Gonzalez, and R. Krause, "A trainable clustering algorithm based on shortest paths from density peaks," *Science Advances*, vol. 5, no. 10, article eaax3770, 2019.
- [19] Z. Liu, D. Li, Y. Yang, X. Li, X. Lv, and X. Chen, "Design and implementation of the optimization algorithm in the layout of parking lot guidance," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6639558, 2021.
- [20] C. Han, F. Liu, H. Li, and L. Wang, "Application of Dijkstra algorithm in parking guidance," *China New Communications*, vol. 21, no. 6, p. 167, 2019.
- [21] W. Cho, S. Park, M. Kim et al., "Robust parking occupancy monitoring system using random forests," in *2018 International Conference on Electronics, Information, and Communication (ICEIC)*, pp. 1–4, Honolulu, HI, USA, 2018.
- [22] M. Bachani, U. M. Qureshi, and F. K. Shaikh, "Performance analysis of proximity and light sensors for smart parking," *Procedia Computer Science*, vol. 83, pp. 385–392, 2016.
- [23] Y. Cha, "Key technologies of cache and computing in 5G mobile communication network," *Procedia Computer Science*, vol. 83, pp. 385–392, 2016.

- [24] Y. Mostafa, "A new shape descriptor for road network separation from parking lots and intersection detection on VHR remote sensing images," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 3099272, 2021.
- [25] P. Colonna, P. Intini, N. Berloco, V. Fedele, G. Masi, and V. Ranieri, "An integrated design framework for safety interventions on existing urban roads—development and case study application," *Safety*, vol. 5, no. 1, p. 13, 2019.
- [26] Y. Zhang and S. Tian, "Application of Dijkstra optimization algorithm in parking guidance system of parking lot," *Computer Measurement and Control*, vol. 22, no. 1, pp. 191–193, 2014.
- [27] J. Shin, H. Jun, and J. Kim, "Dynamic control of intelligent parking guidance using neural network predictive control," *Computers & Industrial Engineering*, vol. 120, pp. 15–30, 2018.
- [28] M. Ge and Z. Zhao, "Internet of things indoor positioning system based on Bluetooth technology," *Internet of Things Technology*, vol. 11, no. 11, pp. 52–57, 2021.
- [29] S. Xiang and Z. Chunhua, "On-line monitoring and auxiliary system for intelligent parking based on embedded design," in *2020 12th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pp. 308–312, Phuket, Thailand, 2020.

## Research Article

# A Secured and Efficient Anonymous Roaming Scheme of Mobile Internet

Yao Cheng <sup>1</sup>, Li Yue <sup>1</sup>, Naixia Duan <sup>1</sup> and Chenglong Li <sup>2</sup>

<sup>1</sup>School of Information Engineering, Shaanxi Institute of International Trade & Commerce, Xi'an 712046, China

<sup>2</sup>School of Computer Science and Technology, Shandong Jianzhu University, Jinan 250000, China

Correspondence should be addressed to Chenglong Li; [chenglongli\\_sdu@163.com](mailto:chenglongli_sdu@163.com)

Received 9 December 2021; Accepted 14 February 2022; Published 9 March 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Yao Cheng 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.

An anonymous roaming scheme of mobile Internet was discussed in this paper aiming to improve the traditional authentication protocol that cannot satisfy the demand of user's identity authentication when the mobile terminal is roaming in mobile Internet. The authentication server of remote network will complete the identity legitimacy verification of mobile terminal with the help of the home network authentication server. A temporary identity is used to prevent user's anonymity protection from being tracked and eavesdropped, as well as other attacks, which can improve the confidentiality of user's identity and location considerably. This anonymous roaming scheme can achieve a high-level of safety efficiently. This will also satisfy the development of the network technology.

## 1. Introduction

The Internet of Things (IoT) realizes the ubiquitous connection between Things and Things, and between Things and people, and realizes the intelligent perception, recognition, and management of objects and processes. That is, IoT is an information carrier based on the Internet, which enables all objects that can be independently addressed to form an interconnected network.

With the rapid development of information network technology, the evolution of internet from wireless Internet to wireless mobile Internet has made the latter a development trend of the next generation network. Mobile Internet has the characteristics of dynamic topology, open link, and limited bandwidth. However, these features make it easier to intercept and monitor messages, and the mobile Internet faces security threats such as eavesdropping and replaying attacks; so, authenticating through a security portal becomes the top priority. Also, for IoT, the anonymous roaming authentication mechanism is a key technology to ensure communication security.

In the case of authentication server in the mobile Internet, the service is hoped to be provided to the legitimate

users or the completely trusted users. The authentication protocol enables the authentication server and the mobile terminal (mobile terminal, MT) to realize a two-way anonymous authentication. To ensure the legitimacy of MT identity, for example, in [1], an anonymous identity authentication and key agreement scheme for global mobility network is proposed to realize mutual anonymous authentication between the two parties. In [2], a wireless anonymous authentication protocol is proposed to avoid the risk of using the same key for a long time. In [3], an identity-based anonymous wireless authentication scheme is proposed to solve the authentication problem of mobile users while roaming, and the security and anonymity of the scheme are analyzed in detail by using the characteristics of bilinear pairs and elliptic curves. In [4], to reduce message flows of traditional anonymous authentication schemes, a new kind of delegation-based scheme is proposed for wireless roaming networks. In [5], an improved secure anonymous authentication scheme using shared secret keys between home agent and foreign agent was proposed. A broadcast authentication protocol for smart grid communications was created in [6], and the security of proposed scheme was proved with the formal method. In [7], an

improved lightweight authentication protocol for wireless body area networks was created, and the security of the above scheme was proved in the random oracle. In the past few years, many constructions on the roaming authentication were created [8–14].

However, the partial roaming authentication protocol [1, 2] only completes the authentication between the authentication server and the MT for local communication. When MT roams away from the local network to enter the remote network, two-way anonymous authentication between MT and remote network authentication server cannot be realized. Although part of roaming authentication protocol [3–5] can satisfy the requirement of MT identity legitimacy authentication while roaming, the large amount of computation will result in a low efficiency. At the same time, MT will repeatedly apply for service after entering the remote network. Frequent authentication will increase the execution load of MT, reduce the efficiency of roaming mechanism, and threaten the security of privacy information such as MT identity.

In view of the above shortcomings of roaming authentication protocol, this paper proposes an anonymous roaming mechanism for mobile Internet. In this mechanism, when MT roams into a remote network, the remote network authentication server will be assisted by its local network authentication server. Simultaneously, the security of MT privacy information will be guaranteed. The remote authentication server issues roaming certificate for authenticated MT. MT can repeatedly apply for roaming service to remote network during the validity period of the certificate. Based on roaming certificate, remote authentication server can verify the identity of MT. The use of certificate improves the efficiency of the mechanism and reduces the computing load of mobile terminal.

The main innovation of this paper is its verification of the identity legitimacy of MT in roaming process, meanwhile ensuring the anonymity of MT identity, which improves the process's security and efficiency together, and makes up for the deficiency of traditional authentication protocol in identity anonymity and work efficiency while the MT is roaming.

## 2. Anonymous Roaming Mechanism of Mobile Internet

The mobile Internet is mainly composed of MT, home network authentication server (home network authentication server, hs), and remote network authentication server (remote network authentication server, rs). At the same time, it also includes the mobile Internet management center (management center, mc). The frame structure is shown in Figure 1.

The relevant variables and operations used in this paper are defined as follows:

$ID_A$  is the identity or related network label of  $A$ ;  $TID_A$  is the temporary identity generated by the home authentication server HS for  $A$ ;  $Num_A$  is the random secret number selected by  $A$ ;  $S$  is the secret number generated by the calculation;  $\oplus$  is the exclusive OR operation;  $\parallel$  connector;  $KS_A$  is

the private key of  $A$ ;  $KP_A$  is the public key of  $A$ ;  $Cert_A$  is the certificate of  $A$ ; and  $T_A$  is the timestamp generated by  $A$ .

$E(k, m)$  and  $D(k, c)$  are symmetric key encryption/decryption algorithms;  $ENC(KS, m)$  and  $DEC(KP, c)$  are asymmetric key encryption/decryption algorithms;  $H(m)$  is a hash function.

The authentication server of each network registers with the mobile internet management center MC, and the MC is responsible for managing the security and other matters of each authentication server. At the same time, the MC issues identity certificates to each authentication server  $Cert_A = \{ID_A, KP_A, Date_A, LF_A, ENC(KS_{MC}, ID_A \parallel KP_A \parallel Date_A \parallel LF_A)\}$ , where  $Date_A$  is the date the certificate was signed and  $LF_A$  is the validity period of the certificate.

**2.1. MT Registering Home Network.** The MT applies for registration with the local network authentication server HS to complete its identity legality verification, and the HS generates a temporary identity for the legal MT.

- (1) The MT sends a registration application to the HS
- (2) The HS assigns a unique temporary identification number  $TID_{MT}$  to the legally qualified MT

First, the secret number  $S_{MT}$  is generated by the Formula (1), namely,

$$S_{MT} = H(ID_{MT} \parallel Num_{HS}). \quad (1)$$

Then, the temporary identity  $TID_{MT}$  for generating the MT is calculated by using Equation (2), namely,

$$TID_{MT} = S_{MT} ID_{MT} ID_{HS}. \quad (2)$$

The HS establishes the registration information  $\langle ID_{MT}, S_{MT}, TID_{MT}, Num_{HS} \rangle$  for the legally legged MT and hands over the temporary identity  $TID_{MT}$  to the MT for secure storage through the secure channel.

**2.2. MT Anonymous Roaming Mechanism.** After the MT is successfully registered, when the preroaming enters the remote network, the remote network authentication server RS will verify the identity of the MT denial identity based on the anonymous roaming mechanism. The specific application process is shown in Figure 2:

- (1) After the MT generates the random number  $X_0$ ,  $ID_{HS}$ ,  $ID_{RS}$ ,  $TID_{MT}$ , and the time stamp  $T_{MT}$  are encrypted with the public key of HS, that is,  $M = ENC(KP_{HS}, ID_{HS} \parallel ID_{RS} \parallel TID_{MT} \parallel T_{MT})$ , the messages  $ID_{HS}$ ,  $ID_{RS}$ ,  $T_{MT}$ ,  $TID_{MT}$ ,  $M$ , and  $X_0$  and the message signature  $Sig_{MT}$  are encrypted together with the public key of RS, and the public key encryption ensures that only the authentication server of the target network can decrypt it
- (2) RS verifies the integrity of the MT message based on the message signature followed by the verification of the freshness of the message's timestamp to prevent

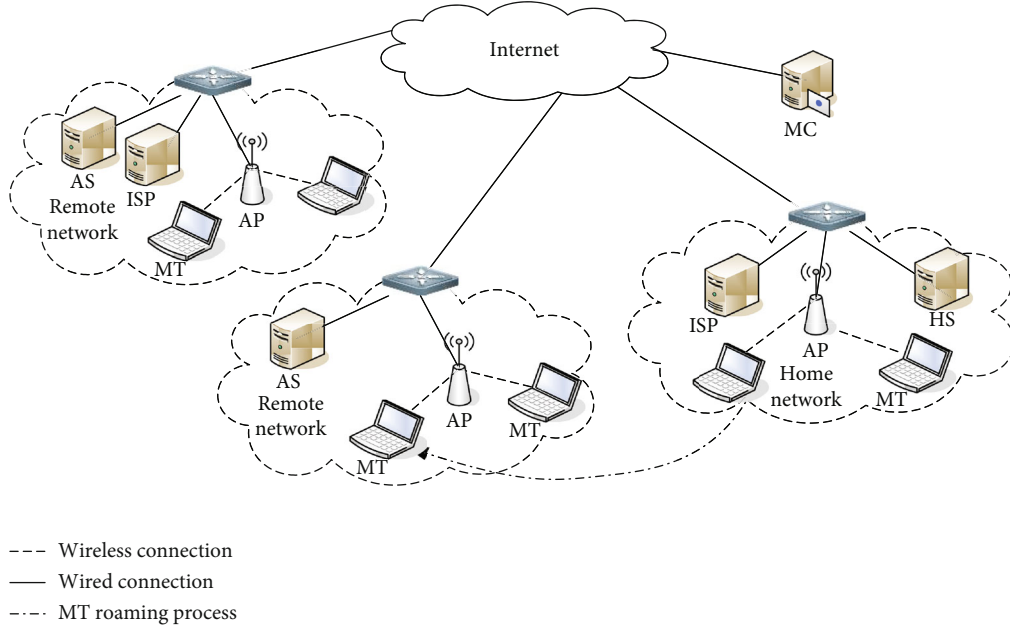


FIGURE 1: Mobile Internet roaming mechanism.

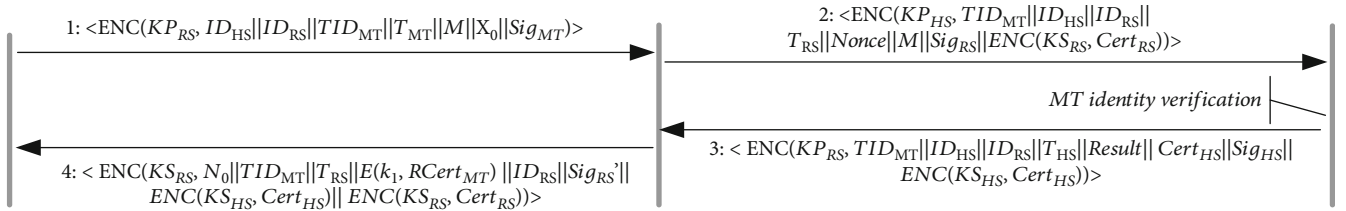


FIGURE 2: Anonymous roaming mechanism of MT.

the replay attack. If the verification is invalid, the roaming request of the MT will be rejected; otherwise, RS will encrypt the message  $TID_{MT}$ ,  $ID_{HS}$ ,  $ID_{RS}$ ,  $T_{RS}$ , Nonce,  $ENC(PS_{RS}, Cert_{RS})$ , and  $M$  and then send them to HS together with the message signature after encrypting them with the public key

- (3) HS verifies the validity of RS ID card and the validity of message integrity and the validity of the timestamp. If the verification fails, the execution will be terminated, and the roaming authentication mechanism will be withdrawn. Otherwise, after the HS decrypts the relevant messages, the identification of MT will be verified by formula (3). That is,

$$ID_{MT}' = TID_{MT}H(ID_{MT} || Num_{HS})ID_{HS}. \quad (3)$$

MT is an illegal user if  $ID_{MT}' \neq ID_{MT}$ , and HS terminates the operation; otherwise, HS encrypts the message  $TID_{MT}$ ,  $ID_{HS}$ ,  $ID_{RS}$ ,  $T_{HS}$ , Result,  $ENC(PS_{HS}, Cert_{HS})$ , and  $Sig_{HS}$  with the public key of RS and sends it to RS.

- (4) RS verifies the authenticity of the HS identity, the validity of the integrity of the message, and the valid-

ity of the timestamp. If not, the execution terminates and the operation exits; otherwise, the RS confirms the identity of the MT according to the relevant information and issues a roaming certificate  $RCert_{MT}$  for the MT whose identity is legal

First, the RS selects the random secret number  $N_0$ ; then, the RS calculates the session key  $k_1 = X_0, N_0$ . Finally, the RS sends the message  $TID_{MT}$ ,  $ID_{RS}$ ,  $T_{RS}$ ,  $Sig_{RS}'$ ,  $N_0$ ,  $E(k_1, RCert_{MT})$ ,  $ENC(KS_{HS}, Cert_{HS})$ , and  $ENC(KS_{RS}, Cert_{RS})$  to the MT after encrypting them with the private key of the RS, where the private key encryption ensures that the roaming response message is sent by the RS.

Through the abovementioned two-round message interaction, the MT identity authentication is completed, and the negotiation between the MT and the RS session key is realized, wherein the session key is determined by the random secret number generated by the MT and the RS. That is,  $k_1 = X_0 \oplus N_0$ . The MT uses the session key to ensure the security of the message in the roaming service process. That is, the MT security obtains the roaming certificate  $RCert_{MT}$ .

**2.3. MT Applying for Service with Certificate.** When MT obtains  $RCert_{MT}$ , it can apply for roaming to remote network authentication server RS many times during its validity



period. The process of MT applying for roaming within the validity period of  $\text{RCert}_{\text{MT}}$  is shown in Figure 3.

- (1) When MT applies for repeated roaming, it generates a random number  $X_i$  and calculates the temporary identity of the roaming application. That is,

$$\begin{aligned} \text{TID}_{\text{MT}_i} &= \text{TID}_{\text{MT}_{i-1}} \oplus N_{i-1} \oplus X_i, \\ \text{TID}_{\text{MT}_0} &= \text{TID}_{\text{MT}}, i = 1, 2, \dots, n. \end{aligned} \quad (4)$$

MT computes the  $E(k_{i-1}, \text{RCert}_{\text{MT}} \| X_i)$  and message signature, reads the message, and then sends the  $\text{Sig}_{\text{MT}_i} \text{TID}_{\text{MT}_i}$ ,  $eE(k_{i-1}, \text{RCert}_{\text{MT}} \| X_i)$ , to RS.

- (2) The RS verifies the freshness of the message and the integrity of the message. If the authentication fails, the roaming service request of the MT will be rejected; otherwise, the message  $E(k_{i-1}, \text{RCert}_{\text{MT}} \| X_i)$  will be decrypted by using the session key  $k_i = X_i \oplus N_i$ . If the MT holds a legal certificate, the RS will generate a random number  $n_i$ , and the session key  $k_i = X_i \oplus N_i$  between the update and the MT will be updated

After the RS reads the timestamp, the message,  $\text{Sig}_{\text{RS}_i}$ ,  $T_{\text{RS}_i}$ , and  $E(k_{i-1}, N_i)$ , will be sent to MT. MT uses Formula (5) to calculate the session key  $k_i$ :

$$k_i = X_i \oplus N_i, \quad (5)$$

in which  $i = 1, 2, \dots, n$ . After the identity authentication between MT and RS has passed, the secure and anonymous communication between MT and RS can be carried out according to the anonymous communication model of mobile Internet in reference [6], which is not discussed in this paper.

#### 2.4. Roaming Structure

**2.4.1. Certificate Structure.** Excessive application for roaming service will not only increase the authentication load of MT but also require that HS must always be online, which makes HS become the authentication bottleneck of the whole roaming mechanism. This paper uses the certificate mechanism to reduce the number of MT verification and to improve the efficiency of the HS. The basic information of the certificate is as follows: (1) validity: the effective time of the certificate, (2) the time of issuance: the time when the certificate is issued, (3) authorization object: the temporary identity  $\text{TID}_{\text{MT}}$  of the certificate holder, and (4) signature: signature information of RS.

**2.4.2. Verification of Legitimacy.** During the validity period of the roaming certificate, MT can apply for roaming service many times with the certificate and verify the authenticity of the roaming certificate through the following steps to judge the identity legitimacy of MT.

- (1) Verify the identity of the issuer through signature information and check whether the contents of the certificate have been tampered simultaneously
- (2) Verify the validity of the certificate based on the validity of the certificate and the time of issue
- (3) Compute  $\text{TID}_{\text{MT}}' = \text{TID}_{\text{MT}_i} \oplus X_i \oplus X_{i-1} \oplus \dots \oplus X_0 \oplus N_i \oplus N_{i-1} \oplus \dots \oplus N_0$ , verify whether  $\text{TID}_{\text{MT}} = \text{TID}_{\text{MT}}'$  holds or not, and whether the person who holds the certificate is the applicant of it. If the above verification is passed, the MT holds true and valid legal certificate

### 3. Security Proof

**3.1. CK Model.** Bellare et al. [15] introduced modularization in 1998 to analyze the security of the protocol, which provides a theoretical basis for constructing a new provable secure key exchange protocol using reusable modules. Then, Canetti and Krawczyk further extended the method [16], which is called CK model.

Two attack models are defined in the ck model, that is, ideal model AM and real model UM. AM is authenticated as link modes. In this model, the attacker is passive and can invoke protocol running, capture protocol participants, query session key, expose, and test the session key. The um model is an unauthenticated link model. Thus, it can only faithfully deliver the same message once and cannot forge, tamper, or replay the message from an uncaptured participant. An UM model is an unverified link model. In addition to performing all attacks in the AM model, attackers can forge, tamper, and replay messages.

**Definition 1.** [16]. Let  $\Pi$  and  $\Pi'$  be  $n$ -side message driven protocols.  $\Pi$  runs in AM, and  $\Pi'$  runs in UM. If for any UM adversary  $U$ , there exists an AM adversary  $A$ , which makes AUTHA,  $\Pi$ , and UNAUTHU,  $\Pi'$  indistinguishable in calculation. Then, the simulation is called in um.

**Definition 2.** [16].

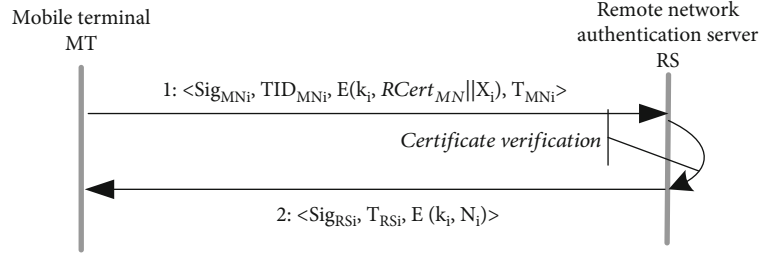
Compiler  $c$  is an algorithm whose input is a protocol description and output a protocol description.

If a compiler  $c$  has a protocol  $c(\pi)$  to emulate in um for any protocol, the editor is called an authenticator. Define 3 [12] in session key security: if for any adversary  $A$ , when and only when the following properties are satisfied, the protocol is session key secure in AM.

**Property 3.** The uncaptured two parties obtain the same session key after participating in the entire agreement.

**Property 4.** If an adversary  $A$  performs a query attack, the possibility that it gets the correct session output value is not more than  $1/2 + \epsilon$ , where  $\epsilon$  is an ignorable decimal in the safety parameter range.

**Theorem 5.** [16]. Assumes that  $\lambda$  is a message transmission authenticator, specifically,  $\lambda$  simulates a simple message

FIGURE 3: MT applies for roaming service with  $RCert_{MT}$ .

transfer protocol in um, assuming that  $c \lambda$  is a compiler defined on the basis of  $\lambda$ , and then  $c \lambda$  is an authenticator. Authenticator is a very important mechanism in modularization method, which can ensure that the security protocol in am can be transformed into security protocol in UM.

The messaging protocol sends a message from one participant to another. A protocol authenticator  $c \lambda$  is a combination of several message transmission authenticators  $\lambda$ . If the protocol in AM has only one message flow, a message transmission authenticator  $\lambda$  can be used as the authenticator  $c \lambda$ . Otherwise, the simulated message transfer authenticator  $\lambda$  of the protocol message flow in AM can be combined together to act as the authenticator  $c \lambda$ . In [16], the basic methods of designing authentication and key agreement protocols based on the CK model are introduced in detail.

3.2. Roaming Protocols in AM. RS relies on the home authentication server HS to authenticate the identity legitimacy and platform credibility of MT. The process of the agreement is as follows:

- (1) The access requests: The MT sends a roaming request message to the RS, which contains the identity of the relevant participant  $\{TID_{MT}, ID_{HS}, ID_{RS}\}$
- (2) The validity verification request: RS after receiving the roaming request of the MT, it is impossible to determine whether the MT is a legally authentic mobile terminal for the reason that the related information of the MT is not grasped, and the assistance authentication of HS is required. RS sends the MT legitimacy verification request  $\{TID_{MT}, ID_{HS}, ID_{RS}, T_{RS}, Nonce_{RS}, Cert_{RS}, M, Sig_{RS}\}$  to the HS
- (3) Legitimacy verification response: after HS receives the request message of RS's legitimacy verification, it first checks whether MT is a legitimate and trusted user and then checks whether RS is a legitimate remote network authentication server. If the above verifications are not passed, the protocol will be terminated and exits; otherwise, the information will be encrypted such as the result of the MT validity verification with the public key of rs, and the cipher text will send ENC x (-  
 $KP_{RS}, TID_{MT} || ID_{HS} || ID_{RS} || T_{HS} || Result || Nonce_{RS} ||$   
 $ENC(KS_{HS}, Cert_{HS}) || Sig_{HS}$ ) all the sighs to RS

- (4) Roaming response: RS decrypts the cipher text message sent by hs using its own private key. RS verifies whether HS is a legitimate home network authentication server. After the verification has passed, it constructs the response message ENC (-  
 $KS_{RS}, N_0 || TID_{MT} || T_{RS} || RCert_{MT} || ID_{RS} || Sig_{RS}' || Nonc$   
 $e_{MT} || ENC(KS_{RS}, Cert_{RS}) || ENC(KS_{HS}, Cert_{HS})$ ) and sends the message to MT

MT verifies that RS is a desired remote network authentication server based on identity certificate, and whether HS is a real home authentication server. Verification fails if the authentication fails.

**Theorem 6.** *The MT roaming protocol is secured in AM when the signature, asymmetric encryption, symmetric encryption, and other algorithms are secured and difficult to solve.*

It is proved that in AM, because the message participant is not captured by the enemy A during the protocol interaction, when the protocol is executed, MT and RS get the untampered  $X_0$  and  $N_0$ . The calculated shared session keys are both  $k_1 = X_0 \oplus N_0$ . Therefore, the protocol satisfies the property of session key security 1.

It is assumed that the enemy  $a$  can distinguish the key agreement parameters from a random number of equal lengths by a probability  $p$ , which cannot be ignored. The probability upper limit of the unsymmetrical encryption algorithm to be breached is  $P_{ENC}$ . The enemy  $a$  can only get  $x_0$  and  $n_0$  by breaking the message encrypted with  $PK_{RS}$  and  $KS_{RS}$  and then perform XOR operation to get the session key. Because the random secret number  $N_0$  generated by rs is transmitted by  $KS_{RS}$  encryption, it is easy to be obtained by the enemy  $a$ . Thereby, the key point of the enemy is that  $X_0$ , which means A can only obtain the random secret number  $X_0$  by breaking the ENC ( $KP_{RS}, ID_{HS} || ID_{RS} || TID_{MT} || T_{MT} || M || X_0 || Nonce_{MT} || Sig_{MT}$ ). Then, the probability of the random secret number  $X_0$  being attacked by the adversary A at least is  $PP_{ENC}$ . If adversary  $a$  can obtain the random number  $X_0$ , then there is  $(1 - PP_{ENC}) \ll PP_{ENC}$  (that is  $PP_{ENC}$  is far greater than  $1 - PP_{ENC}$ ), that is,  $1/2 \ll PP_{ENC}$ ; so, both  $p$  and  $P_{ENC}$  cannot be ignored. This is contrary to the premise that asymmetric encryption algorithm is secure and difficult to solve. So, the

probability of the enemy  $a$  guessing the correct session key  $K_1$  is no more than  $1/2 + \epsilon$ , in which  $\epsilon$  is ignorable. And the protocol satisfies the property of session key security 2. The MT roaming protocol is session key secure in am.

In AM, because the enemy cannot forgery, tamper, and replay the message, they can only transmit the information produced by the legitimate participant. MT and RS get the identity legitimacy and platform credibility verification information without tampering. With the secure session key  $K_1$  negotiated, roaming mechanism is secure in AM.

Proof of completion.

**3.3. Authenticator Construction.** This paper starts with HS authenticating MT, RS authenticating HS, and MT authenticating RS to construct authenticator. For the authentication information flow between RS and HS, and the authentication information flow between mt and RS a signature authenticator  $\lambda_{\text{sig}, T}$  based on timestamp is used, the security proof process of which is detailed in [16]. The specific interactive process of  $\lambda_{\text{sig}, T}$  is as follows: (1)  $A$  obtains timestamp  $t_a$ , computes the signature  $\text{sSig}(m, T_A, B)$ , and sends a message  $\langle m, \text{Sig}(m, T_A, B) \rangle$  to  $B$ . (2) After receiving the message,  $b$  first checks the freshness of the timestamp  $t_a$  and the correctness of the signature. If  $T_a$  is fresh and the signature is correct,  $b$  completes the authentication of  $a$ .

For the authentication message between  $hs$  and  $mt$ , because the message sent by  $mt$  is forwarded by  $rs$ , the message of  $mt$  authentication must be processed accordingly, which cannot directly send relevant real identity information, but also enable  $hs$  to verify the real identity of  $mt$ . Therefore, the identity-based anonymous authenticator  $\lambda_{\text{ENC}, \text{TID}, T}$  is used. Its security and anonymity proof are detailed in reference [9]. The specific interactive process of  $\lambda_{\text{ENC}, \text{TID}, T}$  is described as follows: (1)  $a$  registers to obtain  $t_{id}$ . Use  $b$ 's public key encryption to generate the ENC  $(\text{KP}_B, m \parallel \text{TID}_A \parallel T_A)$  and finally send  $t_{id}$ , ENC  $(\text{KP}_B, m \parallel \text{TID}_A \parallel T_A)$  to  $B$ . (2) After  $B$  receiving the message, the cipher text message is decrypted to verify the validity of the  $\text{TID}_A$ . If the user is illegal, the execution will be terminated. Otherwise, the freshness of the timestamp will be checked. If the verification is passed,  $a$  passes the authentication of  $b$ .

**3.4. Protocol in UM.** First, the above authenticators  $\lambda_{\text{sig}, T}$  and  $\lambda_{\text{ENC}, \text{TID}, T}$  are applied to the am protocol message flow in Section 3.2 of this paper. Then, the authentication of  $mt$  is hidden without affecting the provable security of the protocol. This prevents an attacker from obtaining their true and valid identity information. Finally, the protocol in UM is optimized by using the method in [15], and the  $mt$  roaming protocol in UM shown in Figure 4 is obtained.

**Theorem 7.** *When the signature, asymmetric encryption and symmetric encryption algorithms are secure and difficult to solve, and the  $mt$  roaming mechanism is secure in um.*

*Proof.* It is proved that the  $mt$  roaming mechanism in UM can be automatically compiled according to the ck model because the authenticator used is provable and secure. Then,

the anonymous roaming authentication mechanism is provable under the ck security model.

Completion of proof.  $\square$

Similarly, repeated roaming requests from  $mt$  certificates are also verifiable and secure.

## 4. Model Analysis

### 4.1. Anonymity Analysis

**4.1.1. The Anonymity and Untraceability of Users.** The real identity of the MT does not appear in communication and at the time of registration, it is replaced by the temporary identity  $\text{TID}_{\text{MT}}$ . Since only the HS are in possession of the secret number  $\text{Num}_{\text{HS}}$ , only HS can correctly verify the real identity  $\text{ID}_{\text{MT}}$  of the user through the expression (2) to ensure the anonymity of the MT identity. The different MT corresponds to a different temporary identity  $\text{TID}_{\text{MT}}$  and is generated by a different random number  $\text{Num}_{\text{HS}}$ . Any legal MT cannot calculate the temporary identity of the other MT through its own  $\text{TID}_{\text{MT}}$ .

Each time when a roaming is applied by the same MT, a different temporary identity  $\text{TID}_{\text{MT}}$  is used, which has an untraceable property.

MT encrypts the temporary identity  $\text{TID}_{\text{MT}}$  and passes it to RS, which realizes the anonymity of the user's real identity  $\text{id}_{\text{MT}}$  to  $rs$  and the protection of his temporary identity. Even if the user's temporary identity  $\text{TID}_{\text{MT}}$  is compromised, the attacker can neither know the real identity of the user nor associate the intercepted temporary identity with it nor monitor the communication process of the user and track the message session. To sum up, the use of temporary identity to protect the anonymity of the user identity can effectively prevent attackers from tracking users, eavesdropping, and doing other attacks. This ensures the anonymity of user identity, location, and other privacy information. At the same time, it does not reduce the security of users.

**4.1.2. Anonymity of Certificates.** The roaming certificate can only report whether the identity of its holder is legal. It does not contain the configuration information of MT and its identity information, which means, the roaming certificate is anonymous, and the anonymity depends on the duration of valid authorization. The shorter the authorization time, the stronger the anonymity is. Meanwhile, the anonymity of the certificate is controllable, and controllability depends on the temporary identity information of the certificate holder. It allows only the same user to establish a roaming service. Specifically, it can only prove the identity legitimacy of the same user during the validity period of the certificate.

**4.2. Safety Analysis.** MT uses secret number  $\text{SMT}$  and identity information to calculate temporary identity  $\text{TID}_{\text{MT}}$ , by hash function. The confidentiality of  $\text{SMT}$  and the security of hash function ensure the unforgeability of  $\text{TID}_{\text{MT}}$ .

After the HS checks the freshness of the timestamp, it calculates the validity of the MT to identity its verification, i.e., RS completes the authentication of the MT with the help of HS, in which the message signature guarantees the

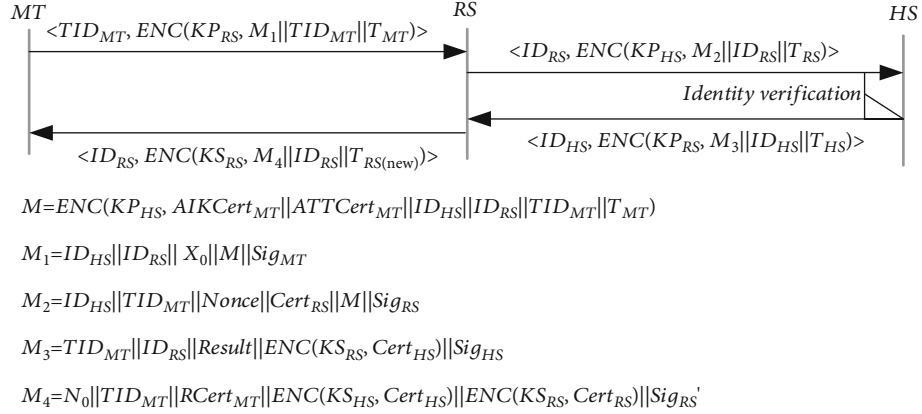


FIGURE 4: MT roaming protocol in UM.

integrity of the message, and the random number can prevent the replay attack.

When the MT repeatedly applies for service, the roaming certificate can prove the validity of the identity. Each time a different session key  $k_i$  is generated, a one-time secret is realized, and the security is enhanced. Since  $X_i$  is selected by the MT and  $N_i$  is selected by the RS,  $k_i$  is a one-time key. Neither party can calculate the generation separately, ensuring the fairness, freshness, and perfect presecracy of the session key. Specially, PGF-JKNN-c got a dramatic 99.9%, which outperforms state-of-the-art methods.

Comparing the methods on the four data sets, we can observe that our proposed methods are better than other methods in three data sets (Indian Pines, KSC, and Salinas) except for University of Pavia data set. University of Pavia data set has the characteristics of few categories and low dimension, which are more suitable for SVM classifier. The SSKNN method performs poorly on KSC data set. That is because SSKNN is more fit able for regular shapes and large-scale shapes. Our proposed FPF-JKNN and GPF-JKNN are more robust to solve complex problems.

#### 4.3. Performance Analysis

**4.3.1. Calculation Efficiency.** In the whole roaming authentication process, MT only needs one hash operation, one symmetric encryption operation, and one asymmetric encryption operation, while HS performs one asymmetric encryption operation and three hash operations. RS performs two hash operations, one symmetric encryption, and two asymmetric decryption operations. The number of communication rounds between MT and RS and between RS and HS is one round. This scheme is comparable with [17, 18] in terms of computational overhead. The results are shown in Table 1.

**4.3.2. Communication Efficiency.** In the roaming mechanism, the first-round interaction verifies the identity legitimacy and platform credibility of MT. If the verification fails, HS and RS will terminate the interaction after the first round, which reduces the execution load of the protocol to a certain extent.

When the identity is legal and the platform is credible, MT applicants with certificates can apply for roaming service on a plurality of times. The use of the certificate mechanism improves the working efficiency of the roaming authentication mechanism, simultaneously effectively reduces the number of authentication times of the identity of the MT, and prevents the RS and HS from becoming a system bottleneck. In the repeated roaming process of the MT certificate holders, the authentication of the identity of the MT can be completed without the assistance of the HS by RS, the MT roaming authentication process for carrying out the 1-round message interaction is realized, and the communication time delay of the MT is reduced.

**4.3.3. Storage Efficiency.** The identity of MT is stored directly by HS. It is unnecessary for special trust center to store and manage it uniformly. MT only needs to store the necessary information, such as temporary identity. The anonymous roaming mechanism in this paper lightens the storage burden of MT.

**4.4. Extensibility.** With the rapid development of network technology, identity legality verification is no longer a necessary and sufficient condition for judging user security. However, it should be concerned with whether the terminal platform is trusted while condition for judging user security. However, it should be the identity is legal. The credibility of the platform is a necessary condition for user security, which promotes the rise and development of trusted computing technology. With the development of trusted computing technology, the mobile trusted module (MTM) [19, 20] specification is intended to be established on mobile terminals. The security mechanism protects the user's private information and sensitive data and builds a secure and reliable mobile trusted terminal. It is only necessary to add the credibility verification information of the user terminal platform in the authentication message of this document, and the HS can verify the credibility of the MT platform according to the platform credibility verification strategy, so that the anonymous roaming requirement of the terminal in the trusted computing environment can be satisfied.



TABLE 1: Comparison of computing overhead by entity.

Operation	Our scheme	Literature [17]	Literature [18]
Hash operation (MT/HS/RS)	1/3/2	7/10/4	5/3/4
Symmetric encryption and decryption(MT/HS/RS)	2/0/2	2/2/2	☆
Asymmetric encryption and decryption(MT/HS/RS)	1/1/2	☆	2/0/2
XOR operation (MT/HS/RS)	0/2/0	5/3/1	☆
Exponent arithmetic (MT/HS/RS)	☆	2/0/2	☆
Chaotoc-maps (MT/HS/RS)	☆	☆	6/2/1
Number of information exchanges(MT-RS/RS-HS)	2/2	2/2	2/2

Note: ☆ indicates that this scheme does not use the operation.

Similarly, the anonymous roaming mechanism of this paper can also be applied to roaming communication and tariff service of mobile user equipment in 3G network environment. RS provides services for MT and charges. If MT pays the fees once on every roaming, it will bring inconvenience. In the MT roaming identity authentication process, the RS charges service fee to the HS, and the HS charges the MT again. Since the HS assists the RS to complete the identity authentication of the MT during the roaming process, the MT cannot deny the cost incurred by the roaming.

## 5. Conclusion

The traditional authentication protocol cannot meet the identity authentication requirements of the mobile terminal roaming service. This paper proposes a mobile Internet anonymous roaming mechanism to improve this disadvantage. When the mobile terminal applies for the roaming service, the remote network authentication server completes the identity verification of the mobile terminal with the assistance of the home network authentication server. The use of temporary identity to achieve user anonymity protection not only makes remote networks and attackers unable to know the user's true identity but also ensures the confidentiality of private information such as user identity and location. The identity is associated with the existing communication information, which can ensure the nontrackability of the private information such as the user identity and location, and effectively prevents attackers from performing attacks such as tracking and eavesdropping on the user. The proposed mechanism does not reduce the process' security during the implementation of the anonymous roaming of mobile terminals. It has the characteristics of security, anonymity, and extensibility.

As the certificate will bring additional storage pressure to the terminal, the next step will be to further study the efficient roaming authentication mechanism under the mobile Internet and design a certificate-free one-round message interactive roaming authentication mechanism.

In the next stage, we will further study the roaming authentication mechanism with better performance based on the conclusions on the sensor distribution [21, 22], performance scheduling [23, 24], etc.

## Data Availability

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

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

The paper was supported by the Science and Technology Project for the Universities of Shandong Province (No. J18KB171), Natural Science Foundation of China under Grant 62102235, Natural Science Foundation of Shandong Province under Grant ZR2020QF029, and Doctoral Fund of Shandong Jianzhu University under Grant XNBS1811. This work was also supported by the Computer Vision and Image Processing Technology Team Construction Project for the Shannxi Institute of International Trade & Commerce.

## References

- [1] M. Gupta and N. S. Chaudhari, "Anonymous two factor authentication protocol for roaming service in global mobility network with security beyond traditional limit," *Ad Hoc Networks*, vol. 84, pp. 56–67, 2019.
- [2] W. Li, S. Zhang, Q. Su, Q. Wen, and Y. Chen, "An anonymous authentication protocol based on cloud for telemedical systems," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 8131367, 2018.
- [3] T. Gao, F. Peng, and N. Guo, "Anonymous authentication scheme based on identity-based proxy group signature for wireless mesh network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, 2016.
- [4] C. Jiang, S. L. Wu, and K. Gu, "New kind of delegation-based anonymous authentication scheme for wireless roaming networks," *International Journal of Network Security*, vol. 20, no. 2, pp. 235–242, 2018.
- [5] K. Park, Y. Park, Y. Park, A. Goutham Reddy, and A. K. Das, "Provably secure and efficient authentication protocol for roaming service in global mobility networks," *IEEE Access*, vol. 5, pp. 25110–25125, 2017.
- [6] S. Aghapour, M. Kaveh, and D. Martín, "An ultra-lightweight and provably secure broadcast authentication protocol for smart grid communications," *IEEE Access*, vol. 8, pp. 125477–125487, 2020.



- [7] V. Kumar, M. Ahmad, A. Kumari, S. Kumari, and M. K. Khan, "SEBAP: a secure and efficient biometric-assisted authentication protocol using ECC for vehicular cloud computing," *International Journal of Communication Systems*, vol. 34, no. 2, 2021.
- [8] U. Chatterjee, D. Mukhopadhyay, and R. S. Chakraborty, "3PAA: a private PUF protocol for anonymous authentication," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 756–769, 2021.
- [9] B. A. Alzahrani, A. Irshad, A. Albeshri, K. Alsubhi, and M. Shafiq, "An improved lightweight authentication protocol for wireless body area networks," *IEEE Access*, vol. 8, pp. 190855–190872, 2020.
- [10] G. Zhang, D. Fan, Y. Zhang, X. Li, and X. Liu, "A privacy preserving authentication scheme for roaming services in global mobility networks," *Security and Communication Networks*, vol. 8, no. 16, 2859 pages, 2015.
- [11] S. A. Chaudhry, A. Albeshri, N. Xiong, C. Lee, and T. Shon, "A privacy preserving authentication scheme for roaming in ubiquitous networks," *Cluster Computing*, vol. 20, no. 2, pp. 1223–1236, 2017.
- [12] X. Li, A. K. Sangaiah, S. Kumari, F. Wu, J. Shen, and M. K. Khan, "An efficient authentication and key agreement scheme with user anonymity for roaming service in smart city," *Personal and Ubiquitous Computing*, vol. 21, no. 5, pp. 791–805, 2017.
- [13] V. Odelu, S. Banerjee, A. K. Das et al., "A secure anonymity preserving authentication scheme for roaming service in global mobility networks," *Wireless Personal Communications*, vol. 96, no. 2, pp. 2351–2387, 2017.
- [14] J. L. Tsai and N. W. Lo, "Provably secure anonymous authentication with batch verification for mobile roaming services," *Ad Hoc Networks*, vol. 44, pp. 19–31, 2016.
- [15] M. Bellare, R. Canetti, and H. Krawczyk, "A modular approach to the design and analysis of authentication and key exchange protocols (extended abstract)," in *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing*, pp. 419–428, Dallas Texas USA, 1998.
- [16] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 453–474, Springer, Berlin, Heidelberg, 2001.
- [17] M. Karuppiah, S. Kumari, X. Li et al., "A dynamic ID-based generic framework for anonymous authentication scheme for roaming service in global mobility networks," *Wireless Personal Communications*, vol. 93, no. 2, pp. 383–407, 2017.
- [18] Q. Xie, H. Bin, X. Tan, and D. S. Wong, "Chaotic maps-based strong anonymous authentication scheme for roaming services in global mobility networks," *Wireless Personal Communications*, vol. 96, no. 4, pp. 5881–5896, 2017.
- [19] M. Kim, H. Ju, Y. Kim, J. Park, and Y. Park, "Design and implementation of mobile trusted module for trusted mobile computing," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 1, pp. 134–140, 2010.
- [20] A. Soltani-Farani, H. R. Rabiee, and S. A. Hosseini, "Spatial-aware dictionary learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 1, pp. 527–541, 2015.
- [21] X. Xue and C. Jiang, "Matching sensor ontologies with multi-context similarity measure and parallel compact differential evolution algorithm," *IEEE Sensors Journal*, vol. 21, no. 21, pp. 24570–24578, 2021.
- [22] X. Xue, X. Wu, C. Jiang, G. Mao, and H. Zhu, "Integrating sensor ontologies with global and local alignment extractions," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6625184, 10 pages, 2021.
- [23] X. Xue and J. Zhang, "Matching large-scale biomedical ontologies with central concept based partitioning algorithm and adaptive compact evolutionary algorithm," *Applied Soft Computing*, vol. 106, article 107343, 2021.
- [24] X. Xue and J. Chen, "Matching biomedical ontologies through compact differential evolution algorithm with compact adaptation schemes on control parameters," *Neurocomputing*, vol. 458, pp. 526–534, 2021.

## Research Article

# Online Missing Data Imputation Using Virtual Temporal Neighbor in Wireless Sensor Networks

Yulong Deng <sup>1,2</sup> Chong Han <sup>1,2</sup> Jian Guo <sup>1,2</sup> Linguo Li <sup>3</sup> and Lijuan Sun <sup>1,2</sup>

<sup>1</sup>College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

<sup>2</sup>Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks,  
Nanjing University of Posts and Telecommunications, Nanjing 210003, China

<sup>3</sup>College of Information Engineering, Fuyang Normal University, Fuyang 236041, China

Correspondence should be addressed to Lijuan Sun; sunlj@njupt.edu.cn

Received 18 December 2021; Accepted 11 January 2022; Published 8 February 2022

Academic Editor: Xingsi Xue

Copyright © 2022 Yulong Deng 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.

A wireless sensor network (WSN) is one of the most typical applications of the Internet of Things (IoT). Missing values exist in the sensor data streams unavoidably because of the way WSNs work and the environments they are deployed in. In most cases, imputing missing values is the universally adopted approach before making further data processing. There are different ways to implement it, among which the exploitation of correlation information hidden in the sensor data interests many researchers, and lots of results have emerged. Researching in the same way, in this paper, we propose VTN imputation, an online missing data imputation algorithm based on virtual temporal neighbors. Firstly, the virtual temporal neighbor (VTN) in the sensor data stream is defined, and the calculation method is given. Next, the VTN imputation algorithm, which applies VTN to make estimates for missing values by regression is presented. Finally, we make experiments to evaluate the performance of imputing accuracy and computation time for our algorithm on three different real sensor datasets. The experiment results show that the VTN imputation algorithm benefited from the fuller exploitation of the correlation in sensor data and obtained better accuracy of imputation and acceptable processing time in the real applications of WSNs.

## 1. Introduction

With the development of the Internet of things (IoT) [1], nowadays more devices and sensors are deployed in the physical environments. Extracting hidden values from the data of IoT becomes challengeable and valuable tasks which make processing ranged from simple tasks such as queries, predictions and classifications to complex processing such as ontology matching [2, 3] and knowledge discovery [4]. As one of the most typical IoT applications, wireless sensor networks (WSNs) [5] are equipped with more nodes with different sensors and are deployed in wider areas with complicated situations. It enables us to obtain a huge amount of physical data in the environment, and these data become the basis for WSN applications. Most of the applications demand complete datasets, i.e., there do not exist missing values in the data obtained from the WSNs because

the missing values degrade the performance of the processing algorithms and even make them inapplicable. For example, in an application that is applied to recognize human activities based on the measurement values obtained from the sensors, such as accelerometer and gyroscopes, where the random forest classifier and support vector machine (SVM) are used for classification, the research work shows that 5% missing rate of values in the dataset makes the performance of HASC recognition decrease to 83% and 84%, respectively, 20% missing rate makes them drop down to 45% and 46%, which is unacceptable for the application [6]. We expect to get complete datasets from WSNs, however, it is a fact that missing values exist commonly and widely in real situations. The way WSNs work and the environment they are deployed in make the data to get lost unavoidably, for example. Factors including the signal strength fading and interferences from the environment bring about 9% to 17%

packet loss over a wireless communication channel approximately [7], which causes some of the measurement values missing in the sensor stream. Previous research showed that 45% of the datasets in the UCI machine learning repository contained missing values [8]. For example, in one of the sensor datasets of this repository, the air quality dataset [9], which is used in our experiments, had approximately 14% of its measurement values missing.

Therefore, it is very important to deal with the missing values in a reasonable way, which can benefit the applications running on the incomplete data of WSNs [10]. Before we go further, there are several points we need to consider.

Firstly, the presentation form of the sensor data decides the way to process it. In the common architecture of WSNs, the way to collect data from the sensor nodes is centralized by the sink node, i.e., the measurement values acquired by the sensors on the nodes are directly forwarded to the sink node using hops or are transferred by the cluster heads till they get to the sink node [11]. Hence, on the sink node, which is usually a base station or a data center, if equipped with more powerful processing resources, the data sent from all nodes is collected and streamed in the form of a time series. In this paper, we focus on this presentation form of data, i.e., sensor data stream on the base station or the data center, and deal with the missing value in it using the online imputation algorithm. Different from the offline imputation that makes processing on the static longtime stored data, online imputation is triggered by the missing value once it occurs in the dynamic data stream.

Secondly, missingness mechanisms decide the design of the imputation algorithms. As a feature to describe the relationship between the missing variables and the underlying values of the variables in the dataset, it comprises of three types of mechanisms: missing completely at random (MCAR), missing at random (MAR), and not missing at random (NMAR) [12]. Among them, MCAR describes the case when the probability of a missing value occurrence for an observed attribute is independent of either the known values or the missing ones, i.e., in WSNs, whether the measurement value is missing does not depend on the values acquired by the sensors, including the missing ones. In most cases, it is a typical situation for the missing values in WSNs caused by communication errors or the faults of sensors. Therefore, most of the research works are based on MCAR so far [13]. In this paper, we carry out our work in the same way. MCAR is applied to generate sensor data streams with simulated missing values that are used to test imputation algorithms.

Thirdly, basic ideas to process missing values direct the way to tackle the incomplete data. There are three basic approaches to cope with missing values: deleting the missing values, continuing data processing with missing values, and imputing missing values to get complete data. The first method is seldom used because it makes the size of the dataset reduced, which brings more information loss. The second method is to make further operations directly on the incomplete dataset. A lattice-based direct mining method is proposed in [14]. It works out the subsets of rules to make classification on the binary dataset with the existence of

missing values. Targeting at classification for incomplete data, a selective neural network ensemble (SNNE) classifying method applies the chosen feature subsets to train neural networks for making classification [15]. Although these methods exhibit good performance in the research work, they are all limited to specified applications, i.e., classifications, and require complex calculations that are not suitable for resource-constrained applications in WSNs. Therefore, the third method, i.e., to make an imputation for the missing values before further processing, is the most effective way to deal with the incomplete data in WSNs. Focusing on the imputing methods to the sensor data stream, lots of algorithms are proposed by researchers. Closed item sets-based association rule mining (CARM) [16] extracts the most recent patterns between the sensor nodes and applies them to impute the missing number of vehicles in a traffic sensor data stream. Besides the association rule mining, more machine learning methods are applied in the imputation in WSNs. Multidirectional recurrent neural network (M-RNN) based on deep learning is applied to make an estimation for the missing values in medical data streams [17]. A deep imputation network (Deep IN) utilizes deep learning to find a continuous missing pattern that contributes to imputation for sensor data streams in a smart space [18]. These methods exploit machine learning to improve the accuracy of imputation but increase the computational complexity as well. Compared with them, statistical technique-based methods are simpler for calculation and more explainable when they are applied in imputation. Mean imputation [19] is the simplest method, however, it presents the worst performance in most cases because it ignores the relationship between the adjacent values, whereas another simple method, i.e., the linear interpolation model (LIN) [20], applies the temporal relationship to interpolate the missing values and gets improved in accuracy. Lots of data analysis results prove that there exist correlations between measurement values in sensor data streams. Data estimation using a statistical model (DESM) [21] utilizes both spatial and temporal correlations between the sensor nodes and applies previous values to make estimates for the missing values. It is weighed by the Pearson correlation coefficient. In [22], the tensor-based description of the multiple attribute sensor data is used to exploit the correlation between the attributes to make imputation with higher accuracy. Moreover, regression tools are widely used by many researchers to design imputation algorithms. In the spatial correlation-based adaptive missing data estimation algorithm (AMR) [23], spatial correlations between the nodes in WSNs are exploited and adjustable regression models are applied to calculate the estimations for the missing values. For improving the accuracy of imputation, in a new estimation model based on a spatial-temporal correlation analysis (STCAM) [24], four subalgorithms based on regression are combined to deal with the missing values in the sensor datasets. Temporal and spatial nearest neighbor value-based missing data imputation (TSNN) [25] is proposed to make imputation in WSNs by the combination of four spatial and temporal nearest neighbor values, which can exploit the correlation information more effectively.

In this paper, we focus on the way that combines the correlation information with regression tools because it can obtain high imputation accuracy with relatively simple calculations. The aforementioned research works utilize correlations in different ways, however, there still exists room for improvement. Besides, when we deal with missing values in the sensor data stream with limited information, in other words, when the measurement values from spatial neighbor nodes are unavailable or partly available, some of these previous methods present reduced performance, which requires further research work to design a more suitable imputation algorithm. In addition, the computational time is seldom discussed in the previous research work, however, it is very important, especially for online imputation. It is analyzed in this paper based on experiment results.

The rest of this paper is organized as follows: firstly, in Section 2, we make a review of representative imputation algorithms in WSNs and summarize the main contributions of our work. Then, in Section 3, we give the definition of VTN, and based on it, we propose our VTN imputation algorithm. Next, the experiments for evaluating the VTN imputation algorithm and the results are elaborated in Section 4. Finally, in Section 5 and 6, we make a discussion and conclude our research work.

## 2. Related Work

In this section, we present the typical algorithms in previous research works. Firstly, for discussing the related work conveniently, we propose the description of the sensor data stream in WSNs. Then, the representative algorithms can be redescribed briefly by math expressions. Finally, based on the discussion of the deficiencies of previous works, we present our contributions in this paper.

The imputation for missing values in the sensor data stream is one of the attractive research areas in WSNs, and many researchers have proposed a variety of imputing algorithms in recent decades. They make the imputation for missing values from the perspective of time, space, or both dimensions. Before introducing our VTN algorithms, several representative algorithms are summarized as follows,

and they are also used as comparative algorithms in this paper.

The sensor data stream in WSNs is the common object that all imputation algorithms deal with for redescribing the implementation of these typical algorithms, and to present our algorithm later easily, we give the subsequent description.

In general, the nodes in a wireless sensor network are working continuously. Let  $N$  be a set of  $m$  sensor nodes,  $N = \{n_1, n_2, \dots, n_{m-1}, n_m\}$ ,  $n_i \in N$ , and let  $T$  be a time point series  $T = \{t_1, t_2, \dots, t_{m-1}, t_m, \dots\}$ ,  $t_j \in T$ . For each node  $n_i$ ,  $G(i, j)$  represents its spatial nearest neighbors at the time point  $t_j$ , and the number of the spatial nearest neighbors is  $k_s$ .  $U(i, j)$  represents its temporal nearest neighbors at the time point  $t_j$ , and the number of the temporal nearest neighbors is  $k_t$ . Let  $S$  be a set of  $p$  sensors equipped on each of the node,  $S = \{s_1, s_2, \dots, s_{p-1}, s_p\}$ ,  $s_k \in S$ , and let  $V$  be a set of measurement values acquired by  $p$  sensors, respectively, on a node,  $V = \{v_1, v_2, \dots, v_{p-1}, v_p\}$ ,  $v_k \in V$ , and  $V(n_i, t_j)$  is the vector of the sensor values on the node  $n_i$  at time point  $t_j$ .

The measurement data obtained by the nodes can be transferred to the base station or data center, where it is collected in the form of sensor data stream, denoted by  $S$   $DS$ , which can be described as follows:

$$SDS = \begin{bmatrix} V(n_1, t_1) & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ \cdots & \cdots & V(n_m, t_i) & \cdots \end{bmatrix}. \quad (1)$$

The measurement value on the sensor  $s_k$  of node  $n_i$  at the time point  $t_j$  can be denoted by  $v(n_i, t_j, s_k)$ , and if there is a missing value on the sensor  $s_k$  of node  $n_i$  at the time point  $t_{\text{miss}}$ , the imputation algorithm calculates an estimated value for it, denoted by  $v(n_i, \widehat{t_{\text{miss}}}, s_k)$ .

For example, suppose there are only two sensors  $s_t$  and  $s_h$  on each node for acquiring temperature and humidity values, respectively. In this case,  $S = \{s_t, s_h\}$ , and hence, the measurement values are  $v(n_i, t_j, s_t)$  and  $v(n_i, t_j, s_h)$ , and the sensor data stream can be reduced as follows:

$$SDS = \begin{bmatrix} (v(n_1, t_1, s_t), v(n_1, t_1, s_h)) & \cdots & (v(n_1, t_m, s_t), v(n_1, t_m, s_h)) & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ (v(n_m, t_1, s_t), v(n_m, t_1, s_h)) & \cdots & (v(n_m, t_m, s_t), v(n_m, t_m, s_h)) & \cdots \end{bmatrix}. \quad (2)$$

We assume that there is a missing value on the sensor  $s_k$  of the node  $n_i$  at the time point  $t_{\text{miss}}$ , and it can be imputed by following algorithms using  $v(n_i, \widehat{t_{\text{miss}}}, s_k)$ .

**2.1. Linear Interpolation Model (LIN) Algorithm.** The linear interpolation is utilized to calculate the estimated value  $v(n_i, \widehat{t_{\text{miss}}}, s_k)$ . It applies two previous values that have been

read in from the data stream at the time points  $t_a$  and  $t_b$ , which are the nearest to the time point  $t_{\text{miss}}$ .

$$v(n_i, \widehat{t_{\text{miss}}}, s_k) = v(n_i, t_b, s_k) + \frac{(v(n_i, t_a, s_k) - v(n_i, t_b, s_k))(t_{\text{miss}} - t_b)}{t_a - t_b}, \quad (3)$$

where  $t_b < t_a < t_{\text{miss}}$ .

**2.2. Mean of Temporal Neighbors (MEAN) Algorithm.** MEAN is a simple imputation algorithm, in which the estimated value for imputation is the average of the  $p$  values of temporal neighbors at the nearest time points before the occurrence of the missing value.

$$v(n_i, \widehat{t_{\text{miss}}}, s_k) = \frac{1}{|U|} \sum_U v(n_i, t_{\text{non-miss}}, s_k), \quad (4)$$

where  $t_{\text{non-miss}} \in U$ ,  $t_{\text{non-miss}} < t_{\text{miss}}$ ,  $|U| = p$ .

**2.3. Temporal and Spatial Nearest Neighbor Value-Based Missing Data Imputation (TSNN) Algorithm.** Assume that there exist spatial nearest neighbors  $G(i, j)$  and temporal

nearest neighbors  $U(i, j)$  for the missing value of sensor  $s_k$  on node  $n_i$  at the time point  $t_{\text{miss}}$ . It has four different nearest neighbor values:  $N_{sgm}(s_i, t_j)$ ,  $N_{sdn}(s_i, t_j)$ , which come from spatial nearest neighbors in the geometrical distance and in data distance,  $N_{ttm}(s_i, t_j)$  and  $N_{tdn}(s_i, t_j)$ , which come from the temporal nearest neighbors in time distance and in data distance. Then, the correlations among the measurement value of the node  $n_i$  and its four nearest neighbors can be applied to calculate the estimated value for imputation by regression.

$$v(n_i, \widehat{t_{\text{miss}}}, s_k) = \lambda v(n_i, \widehat{t_{\text{miss}}}, s_k)(S) + (1 - \lambda)v(n_i, \widehat{t_{\text{miss}}}, s_k)(T), \quad (5)$$

where,

$$\begin{cases} v(n_i, \widehat{t_{\text{miss}}}, s_k)(S) = \frac{R_{s_k}^2(1)}{R_{s_k}^2(1) + R_{s_k}^2(2)} v(n_i, \widehat{t_{\text{miss}}}, s_k)(1) + \frac{R_{s_k}^2(2)}{R_{s_k}^2(1) + R_{s_k}^2(2)} v(n_i, \widehat{t_{\text{miss}}}, s_k)(2), \\ v(n_i, \widehat{t_{\text{miss}}}, s_k)(T) = \frac{R_{s_k}^2(3)}{R_{s_k}^2(3) + R_{s_k}^2(4)} v(n_i, \widehat{t_{\text{miss}}}, s_k)(3) + \frac{R_{s_k}^2(4)}{R_{s_k}^2(3) + R_{s_k}^2(4)} v(n_i, \widehat{t_{\text{miss}}}, s_k)(4), \end{cases} \quad (6)$$

$$\begin{cases} v(n_i, \widehat{t_{\text{miss}}}, s_k)(1) = \alpha_{s_k}(1) + \beta_{s_k}(1)N_{sgm}(n_i, t_{\text{miss}}), \\ v(n_i, \widehat{t_{\text{miss}}}, s_k)(2) = \alpha_{s_k}(2) + \beta_{s_k}(2)N_{sdn}(n_i, t_{\text{miss}}), \\ v(n_i, \widehat{t_{\text{miss}}}, s_k)(3) = \alpha_{s_k}(3) + \beta_{s_k}(3)N_{ttm}(n_i, t_{\text{miss}}), \\ v(n_i, \widehat{t_{\text{miss}}}, s_k)(4) = \alpha_{s_k}(4) + \beta_{s_k}(4)N_{tdn}(n_i, t_{\text{miss}}), \end{cases} \quad (7)$$

where  $\alpha_{s_k}(1)$  to  $\alpha_{s_k}(4)$ ,  $\beta_{s_k}(1)$  to  $\beta_{s_k}(4)$  are regression coefficients.  $R_{s_k}^2(1)$  to  $R_{s_k}^2(4)$  are the measures of the fit of the regression models.  $\lambda$  is the spatial-temporal coefficient, which can be calculated based on the contribution ratio of the spatial-temporal neighbors.

Particularly, when the spatial neighbors and the values of the other sensors on the same node are not available, TSNN algorithm degrades to,

$$v(n_i, \widehat{t_{\text{miss}}}, s_k) = v(n_i, \widehat{t_{\text{miss}}}, s_k)(3) = \alpha_{s_k}(3) + \beta_{s_k}(3)N_{ttm}(n_i, t_{\text{miss}}). \quad (8)$$

**2.4. Data Estimation Using Statistical Model (DESM) Algorithm.** The estimated value  $v(n_i, \widehat{t_{\text{miss}}}, s_k)$  can be calculated based on the temporal nearest neighbors and spatial nearest neighbors of the missing value of sensor  $s_i$  on node  $n_i$  at time point  $t_{\text{miss}}$ . A correlation coefficient is obtained from the node  $n_i$ , and its spatial neighbors inside the sensor streams can be used as the weight coefficient.

$$v(n_i, \widehat{t_{\text{miss}}}, s_k) = (1 - \alpha)v(n_i, t_a, s_k) + \alpha v(n_i, t_a, s_k) \cdot \left( 1 + \frac{v(g(i, t_{\text{miss}}), t_{\text{miss}}, s_k) - v(g(i, t_{\text{miss}}), t_a, s_k)}{v(g(i, t_{\text{miss}}), t_a, s_k)} \right), \quad (9)$$



where  $t_a < t_{\text{miss}}$ ,  $g(i, j) \in G(i, j)$ ,  $g(i) = g(i, j)_{j=1}^{j=a}$ , and the weight parameter  $\alpha$  can be computed as follows:

$$\alpha = \frac{\text{Cov}(RV(n_i, s_k), RV(g(i), s_k))}{\sigma RV(n_i, s_k) RV(g(i), s_k)}, \quad (10)$$

$$\begin{cases} RV(n_i, s_k) = \{v(n_i, t_1, s_k), v(n_i, t_2, s_k), \dots, v(n_i, t_{a-1}, s_k), v(n_i, t_a, s_k)\}, \\ RV(g(i), s_k) = \{v(g(i), t_1, s_k), v(g(i), t_2, s_k), \dots, v(g(i), t_{a-1}, s_k), v(g(i), t_a, s_k)\}, \end{cases} \quad (11)$$

Particularly, when the spatial neighbors are not available, the estimated value alone can be calculated based on temporal neighbors as follows:

$$v(n_i, \widehat{t_{\text{miss}}}, s_k) = v(n_i, t_a, s_k). \quad (12)$$

The above algorithms can be applied to make an imputation for the missing values in sensor data streams in WSNs, and their effectiveness is verified in experiments or real situations. However, there are still some deficiencies that should be corrected in further research work.

The temporal neighbors, spatial neighbors, or both are utilized to calculate the estimated value in some ways for imputation. As we present above, the sensor data stream (SDS) is a data stream combined with spatial and temporal data. If they are all available to make an imputation, the performance of the estimation algorithm can be improved because of more information extracted from more nonmissing values of the spatial and temporal neighbors. However, in WSNs, affected by the following problems occurring in the network, the spatial data may be unavailable or be difficult to be applied by the imputation algorithm. One situation is that there are also existing missing values in the spatial neighbors, which makes it impossible to obtain data from them at some time points of SDS. Another common situation is that the neighbors can be changing in the self-organized network, which makes it hard to obtain continuous measurement data from a fixed neighbor of a node at all time points of SDS. Moreover, when the node is out of sync with its spatial neighbors when transferring data to the base station or data center, it causes the rows in the matrix of SDS to be asynchronous. In other words, it is unable to obtain the data from neighbors at these time points in the data stream.

LIN and MEAN algorithms are not affected by the above problem because they only utilize temporal neighbors of the missing value. However, the performances of DESM and TSNN algorithms are reduced without the support of the data from spatial neighbors.

LIN and TSNN algorithms apply temporal neighbors to calculate the estimated value by linear interpolation. It makes them more accurate than MEAN, which only gets the estimated value by the arithmetic average of temporal neighbor values. When they two are applied

in offline data, which are already stored on the base station or data center, the nonmissing values at all time points in the observed data window can be applied to calculate the estimated value, however, when they are applied in the online stream, the nonmissing values at the time points after the missing point are not available to them, which causes the calculation to be less precise. TSNN has better performance than LIN, DESM, and MEAN when they are working on offline data because it utilizes the correlation between the nearest neighbor values and raw values, and the regression is applied to make the final estimated value. In the regression step, the training data are selected from the nonmissing values alone based on the distance between the two values. If we can find a new way to select more suitable training data by more reasonable rules for regression, it can get better accuracy for imputation.

In this paper, addressing the above problems, we propose a new imputation algorithm VTN that works for online sensor data stream in WSNs. The main contributions of our work are described as follows:

VTN works based on the temporal neighbors in SDS and only requires the measurement data from one sensor on the node. Hence, it is not affected by the availability of spatial neighbors, and the algorithm can be deployed in WSNs with multiple sensor nodes or with a single sensor node.

Similar to LIN and TSNN, the technology of linear interpolation based on temporal neighbors is also applied in the VTN algorithm. However, different from them, in VTN, the nonmissing value at the next time point of the current missing point is required to read in to obtain the extra data, and therefore, based on the past information provided by the previous nonmissing values and the future information provided by the nonmissing value at the next time point, the VTN algorithm can calculate the virtual temporal neighbor with relatively higher accuracy, and it builds up the foundation to make a precise estimation for the missing value.

Compared with TSNN, VTN makes an improvement in the rule of selecting nonmissing temporal neighbors. The temporal neighbors that are eligible for training data not only have the value nearer to the temporal neighbor at the missing time point but also have a

similar change rate with it. This new way to choose training data helps the VTN algorithm get higher accuracy because the change rate extracted extra information hidden in the data, and it is beneficial to the estimation of missing values.

Besides the accuracy of imputation, the computational time is another evaluation standard for imputation algorithms, especially for the online imputation of the sensor data stream. We make a detailed analysis for it based on the experiment results and give the final evaluation of the VTN algorithm in the real application in WSNs.

### 3. Materials and Methods

In this section, we present our VTN algorithm in detail. Firstly, we define the virtual temporal neighbor (VTN) and give the algorithm to calculate it. Then, the correlation between the VTNs and their raw values is studied as the basis

of our algorithm designing. Next, we describe the VTN imputation algorithm based on VTNs. Finally, the method to set the parameter  $\theta$  for the algorithm is suggested.

#### 3.1. Virtual Temporal Neighbor (VTN)

**3.1.1. Definition and Calculation Method of Virtual Temporal Neighbor.** VTN is the basis of VTN algorithm. Based on our previous description of sensor data stream in Section 2, we give the definition of VTN as follows:

*Definition.* Virtual temporal neighbor (VTN): in the sensor data stream of a WSN, the VTN of the sensor  $s_k$  of the node  $n_i$  at the time point  $t_j$  exists and is denoted by  $vtn(n_i, t_j, s_k)$  if and only if there exist two measurement values acquired by the same sensor of the same node at the time point  $t_a$  and  $t_b$ , and they satisfy the following conditions:

$$\left\{ \begin{array}{l} t_a = \left\{ t_x \mid \begin{array}{l} \forall t_y < t_j, t_y \in T, v(n_i, t_y, s_k) \neq NA; \\ \exists t_x, t_x < t_j, t_x \in T, v(n_i, t_x, s_k) \neq NA \text{ that makes } |t_j - t_x| \leq |t_j - t_y| \end{array} \right\}, \\ t_b = \left\{ t_x \mid \begin{array}{l} \forall t_y > t_j, t_y \in T, v(n_i, t_y, s_k) \neq NA; \\ \exists t_x, t_x > t_j, t_x \in T, v(n_i, t_x, s_k) \neq NA \text{ that makes } |t_x - t_j| \leq |t_y - t_j| \end{array} \right\}, \end{array} \right. \quad (13)$$

where  $NA$  denotes the missing value at the time point.

If the VTN at the time point  $t_j$  exists, it can be calculated by linear interpolation as follows:

$$vtn(n_i, t_j, s_k) = \frac{v(n_i, t_a, s_k)(t_b - t_j) + v(n_i, t_b, s_k)(t_j - t_a)}{t_b - t_a}, \quad (14)$$

and the change rate of VTN denoted by  $\Delta vtn(n_i, t_j, s_k)$  can be computed as follows:

$$\Delta vtn(n_i, t_j, s_k) = \frac{v(n_i, t_b, s_k) - v(n_i, t_a, s_k)}{t_b - t_a}. \quad (15)$$

For calculating VTN, we need to implement the data structures on the base station or data center in the WSN, shown as Figure 1. The input data matrix (IDM) is used to store the data read from the sensor data stream SDS by the stream reader program, and it is also the data source for the imputation program, for a specified sensor of a node, the measurement value at the time point  $t_j$  is reduced as  $v(j)$ . The values and the time points are stored in different rows of the matrix and are referred to by the  $s\_index$  for the stream reader. In addition, three indices, namely the  $u\_index$ ,  $c\_index$ , and  $v\_index$ , are applied to access the matrix in the VTN algorithm. The VTN matrix (VTNM) is another matrix for storing the calculated VTN and its change rate and is accessed by the  $vtn\_index$ . Waiting indices vector (WIV)

stores the  $c\_index$  referring to the time point at which the VTN cannot be calculated temporarily.

The calculation of VTN outputs the VTN and its change rate of the value at the previous time point when it gets a new value from the stream. Moreover, it scans the waiting indices vector to calculate the VTN for those previous values that are waiting for getting their VTN calculated. The flow chart of VTN calculation is shown as Figure 2.

The pseudocode for calculating VTN is shown in Algorithm 1.

**3.1.2. Correlations between Raw Values and Temporal Neighbor Values.** As discussion in Section 2, similar to TSNN, in the VTN algorithm, we also utilize the correlation between the temporal neighbors and their raw values. However, we make further improvements by changing the method to calculate the temporal neighbors. To verify the effectiveness of our methods, we make exploratory experiments on the real WSNs dataset.

Firstly, we get a test dataset by reading the streamed data in a random time frame on the Intel lab dataset [26]. Then, we calculate the temporal neighbors of the raw values according to the methods of TSNN and VTN, respectively. Next, the correlation between the temporal neighbors and their raw values is plotted along with the Pearson correlation coefficient between them. The results are shown in Figure 3.

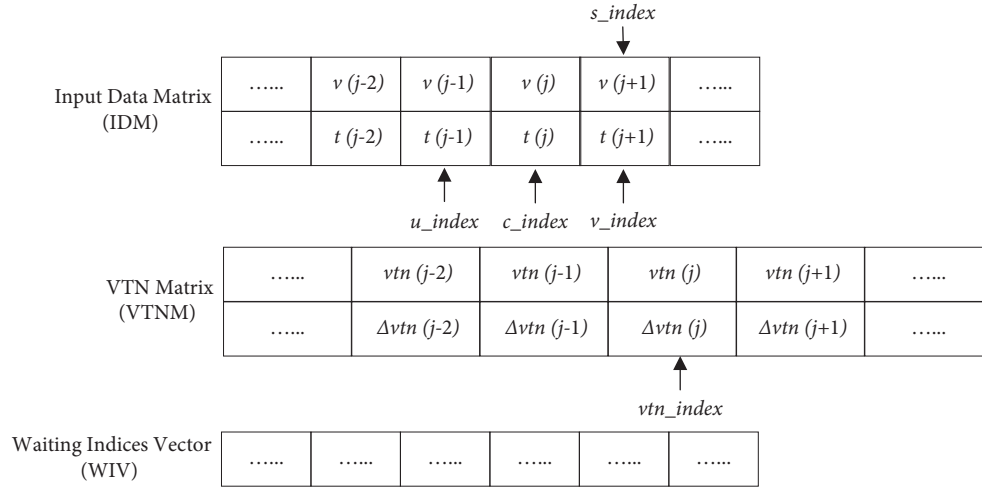


FIGURE 1: Data structures used in the calculation of VTN.

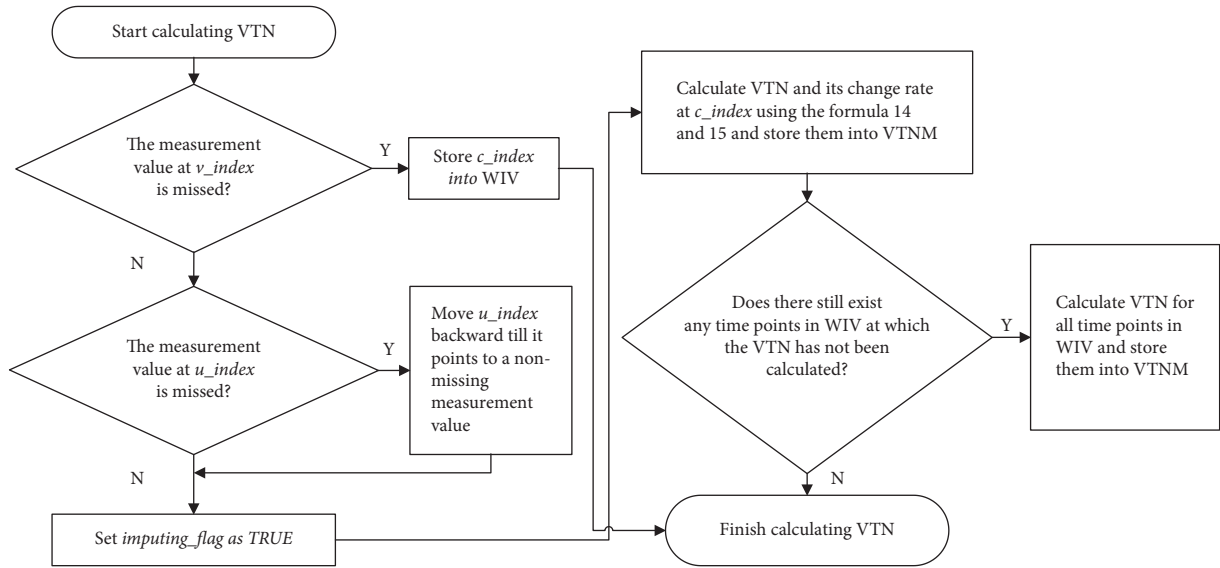


FIGURE 2: Flow chart of VTN calculation.

Figure 3 shows that the correlation in temperature data is stronger than that in humidity data. It conforms to the real situation because the humidity has more fluctuation than temperature because of more affecting physical factors in the environment. Compared with TSNN, we can find that the data points are less scattered along the diagonal and that the correlation is much stronger in VTN. Upon repeating the same experiments on different time frames or on other real datasets, we get similar results. It indicates that the method to calculate the temporal neighbors in VTN is better than the one in TSNN, which helps compute a more accurate estimation value by regression in the next step.

### 3.2. VTN Imputation Algorithm

#### 3.2.1. The Calculation of the Estimated Value Based on VTN.

Once we have VTNs for all nonmissing values in the sensor data stream, the estimated value for the missing time point can be calculated by regression. The estimation equation is as follows:

$$v(n_i, \widehat{t}_{\text{miss}}, s_k) = \alpha_{s_k} + \beta_{s_k} vtn(n_i, t_{\text{miss}}, s_k), \quad (16)$$

$\alpha_{s_k}$  and  $\beta_{s_k}$  can be calculated by minimized residual sum of squares as follows:

```

Input: input data matrix (IDM), VTN matrix (VTNM), u_index, v_index, c_index, vtn_index, waiting indices vector (WIV),
      imputing_flag.
Output: VTN matrix (VTNM), imputing_flag. 1.
if IDM[1, v_index] = NA then
(2)   Add c_index into WIV
(3)   else if IDM[1, v_index] ≠ NA and IDM[1, u_index] ≠ NA then
(4)   imputing_flag ← TRUE
      VTNM [1, vtn_index] ← (IDM[1, v_index] - IDM[1, u_index]) *
      (IDM[2, c_index] - IDM[2, u_index]) / (IDM[2, v_index] - IDM[2, u_index] + IDM[1, u_index])
(5)   VTNM[2, vtn_index] ← (IDM[1, v_index] - IDM[1, u_index]) / (IDM[2, v_index] - IDM[2, u_index])
(6)   if WIV ≠ NULL then
(7)     Reverse WIV
(8)     for each element i in WIV do
(9)       c_index_temp ← i
(10)      u_index_temp ← u_index
(11)      v_index_temp ← v_index
(12)      if c_index_temp = u_index_temp then
(13)        u_index_temp ← u_index_temp - 1
(14)        while VTNM[u_index_temp, 1] = NA do
(15)          u_index_temp ← u_index_temp - 1
(16)        end while
(17)      end if
(18)      VTNM[1, c_index_temp] ← (IDM[1, v_index_temp] - IDM[1, u_index_temp]) *
      (IDM[2, c_index_temp] - IDM[2, u_index_temp]) / (IDM[2, v_index_temp] - IDM[2, u_index_temp]) +
      IDM[1, u_index_temp]
(19)      VTNM[2, c_index_temp] ← (IDM[1, v_index_temp] - IDM[1, u_index_temp]) / (IDM[2, v_index_temp]
      - IDM[2, u_index_temp])
(20)    end for
(21)  end if
(22) end if
(23) return VTN matrix (VTNM), imputing_flag.

```

ALGORITHM 1: Calculation of VTN.

$$\alpha_{s_k}(1), \beta_{s_k}(1) = \underset{\alpha'_{s_k}(1), \beta'_{s_k}(1)}{\operatorname{argmin}} \sum_{T_{\text{sample}}} \left( v(n_i, t_{\text{sample}}, s_k) - \alpha'_{s_k}(1) - \beta'_{s_k}(1) \operatorname{vtn}(n_i, t_{\text{sample}}, s_k) \right)^2, \quad (17)$$

where  $t_{\text{sample}} \in T_{\text{sample}}$ .  $T_{\text{sample}}$  is the selected timepoint set of nonmissing data in the input data matrix IDM.

To have the closest values and change rates in the same direction to the VTN of the missing value, the VTNs in the  $T_{\text{sample}}$  can be obtained as follows:

$$T_{\text{sample}} = \left\{ T' \mid \forall T'' \subset T, |T''| = \theta, t' \in T''; \exists \Delta \operatorname{vtn}(n_i, t', s_k) * \Delta \operatorname{vtn}(n_i, t_{\text{miss}}, s_k) \geq 0 \text{ and } T' = \underset{T''}{\operatorname{argmin}} \sum_{T''} |\Delta \operatorname{vtn}(n_i, t', s_k) - \Delta \operatorname{vtn}(n_i, t_{\text{miss}}, s_k)| \right\}. \quad (18)$$

$\theta$  is the preset number of VTNs of nonmissing values that are used as training data of regression in the VTN imputation algorithm. As the result,  $v(n_i, t_{\text{miss}}, s_k)$  can be used to impute the missing value in the input data matrix IDM. The flow chart of VTN imputation is shown in Figure 4.

The implementation of VTN imputation is shown in Algorithm 2.

**3.2.2. Discussion about Parameter  $\theta$  in VTN Imputation Algorithm.** In the VTN imputation algorithm, the number

of VTNs of nonmissing values is preset by  $\theta$ . Firstly, all VTNs in the past time points of the input data matrix, including the VTN of the missing value are sorted by values. Next, centered on the VTN of the missing value, twice the number of  $\theta$  VTNs are selected to be sorted again by their change rates if they have the same changing direction as that of the missing value. Finally,  $\theta$  VTNs are picked out and are applied to the regression in the next step. To find the relationship between the two performance indicators, the imputing accuracy and the computational time and the parameter  $\theta$ , we make experiments as follows:

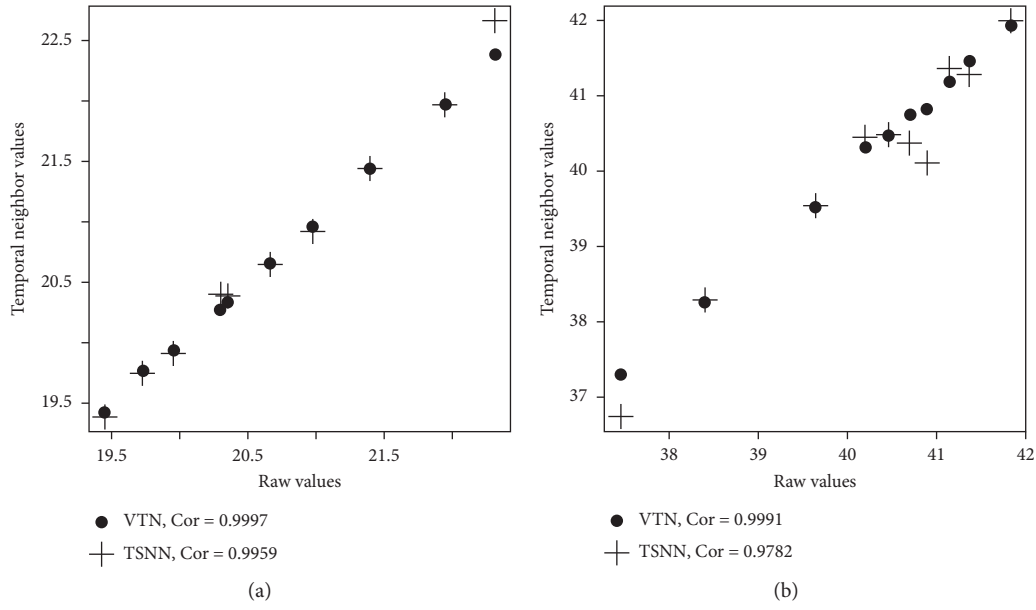


FIGURE 3: Correlation between raw values and their temporal neighbor values on Intel Lab dataset. (a) Temperature. (b) Humidity.

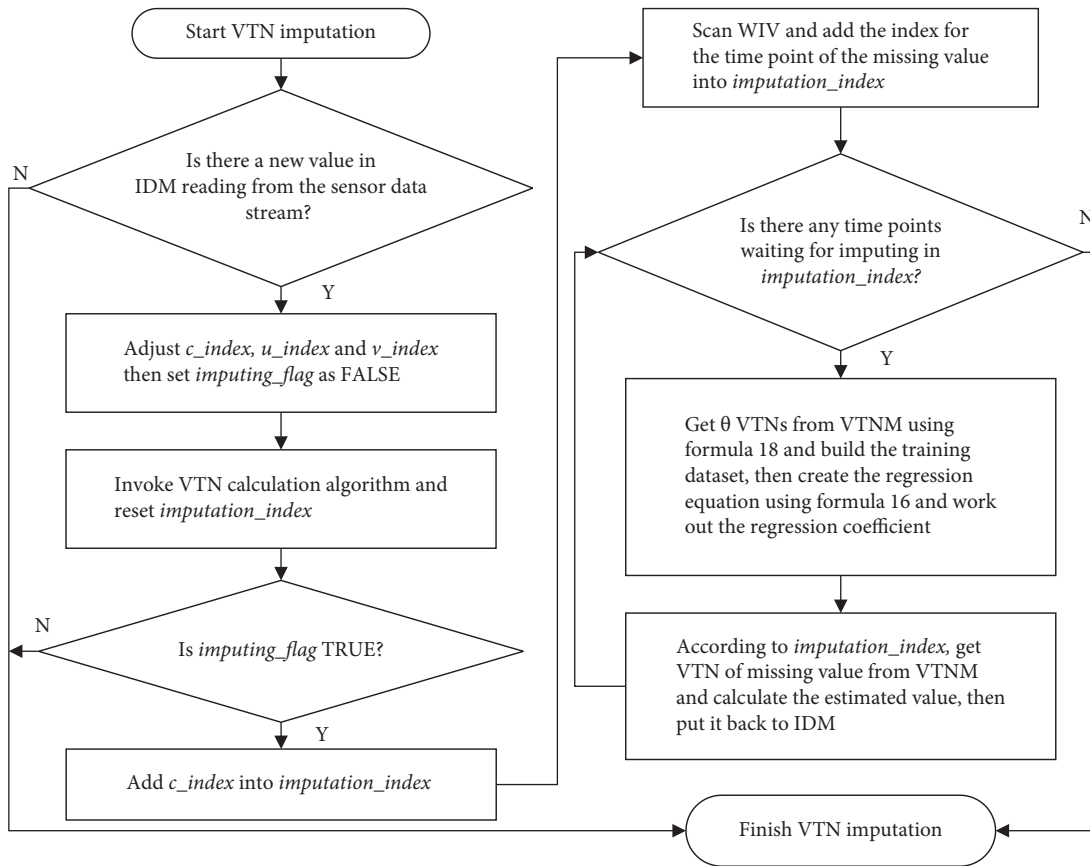


FIGURE 4: Flow chart of VTN imputation.

Firstly, the parameter  $\theta$  is set as 5, and we randomly select one of the subsets from the Intel lab dataset. Apply the MCAR method to temperature values to get the subsets with

missing values at each of the missing percentages ranging from 5% to 50%. Then, we stream the dataset by a rate, such as one data per minute, and apply the VTN imputation



```

Input: input data matrix (IDM), node  $n_i$ , sensor  $s_j$ ,  $\theta$ ,
Output: input data matrix (IDM)
if there exists a new data in IDM then
(27)  $c\_index \leftarrow c\_index + 1$ 
(28)  $U\_index \leftarrow c\_index - 1$ 
(29)  $V\_index \leftarrow c\_index + 1$ 
(30)  $vtn\_index \leftarrow vtn\_index + 1$ 
(31)  $imputing\_flag \leftarrow FALSE$  Get the new data from IDM by  $v\_index$ 
(32) Call Calculation of VTN
(33)  $imputation\_index \leftarrow NULL$ 
(34) if  $imputing\_flag = TRUE$  then
(35)   Add  $c\_index$  into  $imputation\_index$ 
(36) end if
(37) if  $WIV \neq NULL$  then
(38)   for each element  $i$  in  $WIV$  do
(39)     if  $IDM[1, i] = NA$  then
(40)       Add  $IDM[2, i]$  into  $imputation\_index$ 
(41)     end if
(42)   end for
(43) end if
(44) if  $imputation\_index \neq NULL$  then
(45)   Sort  $imputation\_index$  by increasing index
(46)   for each element  $j$  in  $imputation\_index$  do
(47)     for each  $k$  in  $1: j$  do
(48)       if  $VTNM[2, k] * VTNM[2, j] \geq 0$  then
(49)         Add  $VTNM[, k]$  into  $PAST\_VTNM$ 
(50)       end if
(51)     end for
(52)     sort  $PAST\_VTNM$  by increasing order of  $PAST\_VTNM[1, ]$ 
(53)      $new\_imputation\_index \leftarrow new\ index\ for\ imputing\ value\ in\ PAST\_VTNM$ 
(54)      $lower\_bound \leftarrow new\_imputation\_index - \theta$ 
(55)      $high\_er\_bound \leftarrow new\_imputation\_index + \theta$ 
(56)     if  $lower\_bound < 1$  then
(57)        $lower\_bound \leftarrow 1$ 
(58)     else if  $high\_er\_bound > |PAST\_VTNM| - 1$ 
(59)        $high\_er\_bound \leftarrow |PAST\_VTNM| - 1$ 
(60)     end if
(61)      $PAST\_VTNM\_CAN\_DI\_DATE \leftarrow PAST\_VTNM[, lower\_bound: higher\_bound]$ 
(62)      $new\_imputation\_index \leftarrow new\ index\ for\ imputing\ value\ in\ PAST\_VTNM\_CAN\_DI\_DATE$ 
(63)     for each element  $m$  in  $PAST\_VTNM\_CANDADATE$  do
(64)        $CHANGE\_RATE\_DIST$ 
(65)        $\leftarrow |PAST\_VTNM\_CAN\_DI\_DATE[2, new\_imputation\_index] - PAST\_VTNM\_CAN\_DI\_DATE[2, m]|$ 
(66)     end for
(67)     sort  $PAST\_VTNM\_CAN\_DI\_DATE$  by increasing order of  $CHANGE\_RATE\_DIST$ 
(68)     Remove imputing value from  $PAST\_VTNM\_CAN\_DI\_DATE$ 
(69)      $PAST\_VTNM\_CAN\_DI\_DATE \leftarrow PAST\_VTNM\_CAN\_DI\_DATE[, 1: \theta]$ 
(70)      $IDM\_CAN\_DI\_DATE \leftarrow raw\ values\ of\ PAST\_VTNM\_CAN\_DI\_DATE\ in\ IDM$ 
(71)     Construct the estimation equation using  $PAST\_VTNM\_CANDADATE$  and  $IDM\_CANDIDATE$  to
(72)     regress the coefficients  $\alpha_{s_k}, \beta_{s_k}$ 
(73)     Compute  $v(n_i, \widehat{t}_{miss}, s_k)$  using  $\alpha_{s_k}, \beta_{s_k}$  and  $VTNM[1, j]$ 
(74)      $IDM[1, j] \leftarrow v(n_i, \widehat{t}_{miss}, s_k)$ 
(75)   end for
(76) end if
(77) end if
(78) return IDM

```

ALGORITHM 2: VTN imputation algorithm.

algorithm to process the stream. After repeating the experiment 10 times for different subsets, we get the results of the average RMSE and the computational time for the imputation. Increasing the  $\theta$  by step size 5 till 35, we repeat the experiments, and the results are shown in Figure 5.

From Figure 5, we can find that larger  $\theta$  can bring lower RMSE, i.e., the higher accuracy of imputation, however, it increases the time costs as well. Moreover, the performance of accuracy is improved slowly when the  $\theta$  is great than 10. We repeat the same experiments on the different sensor data for humidity on the Intel lab dataset and repeat all experiments on other datasets: the GreenORB dataset [27] and the Air Quality dataset. All of them give similar results.

Therefore, in the VTN algorithm, it is reasonable that the parameter  $\theta$  is preset as the number that is a bit larger than 10 so that we can get relatively higher accuracy and shorter computational time for imputation.

## 4. Experiment Results

In this section, we make evaluation of the performance of the proposed VTN algorithm. Firstly, we introduce evaluation methods and datasets applied in experiments. Then, the experiment steps are given, and results for imputing accuracy and computational time on three different real datasets are described in detail.

*4.1. Evaluation Methods.* Generally, the accuracy of imputation can be evaluated with the root mean square error (RMSE), which can be computed by,

$$\text{RMSE} = \sqrt{\frac{\sum(\text{real value} - \text{estimated value})^2}{\text{the number of estimations}}}. \quad (19)$$

The smaller RMSE indicates the higher accuracy of the imputation algorithm. In addition, the computational time should be taken into consideration for online imputation, and the microbenchmark tool [28] is applied to evaluate them in experiments. We write codes in *R* language, and all experiments are running on a computer with Intel Core i7 2.9 Ghz CPU and 16 GB RAM.

*4.2. Datasets and Their Preprocessing.* Three real world sensor datasets are applied in experiments: the Intel Lab dataset, the GreenOrbs dataset, and the Air Quality dataset. They represent different application scenarios, which ensure the sufficient tests of our imputation algorithm.

The Intel lab dataset contains data from an indoor wireless sensor network deployed in the Intel Berkeley Research lab, which is composed of 54 nodes. Each node is equipped with multiple sensors and is working continuously to obtain different physical quantities on its location every 31 s, including temperature, humidity, light, and voltage values. The GreenOrbs dataset provides us with the outdoor data collected from 120 wireless sensor nodes scattered in a forest. Each node obtains the data, including temperature, humidity, and light values every 80 to 85 s. The Air Quality dataset contains the data of a single multisensory node

network, which gets the air quality data hourly, including temperature, humidity, the concentration of CO, NO<sub>2</sub> concentration, Non Metanic HydroCarbons concentration, and benzene concentration.

Before applying these three datasets to the experiments, we preprocess them as follows: firstly, the missing data contained in these raw datasets are deleted to get complete datasets. Then, the data values, except the temperature and humidity, are removed because only temperature and humidity values are applied in our experiments. Secondly, for every dataset, we randomly divide it into subsets in which the measurement values are continuous, i.e., the raw dataset and the number of values in the subsets are the same for the experiments. It is worth noting that for the Intel Lab dataset and the GreenOrbs dataset, the measurement values can be obtained from more than one note in the WSNs. For testing our algorithm reasonably, we get subsets from 20% of all nodes chosen randomly. Thirdly, in this paper, we only consider the situation that the value obtained by the sensor of the node is missing at the time points randomly. Therefore, some values of the subsets have been marked as dummy missing values (not available, denoted by *NA*) using the Missing Completely at Random (MCAR). Then, we get two versions for each subset, one is with missing values and is applied to test algorithms, and the other is without the missing values. After the missing values in the subset are imputed in experiments, RMSE can be calculated based on the raw values in the latter one. In experiments, the missing percentage can be changed to observe the different results. The upper bound of the missing percentage is set as 50% in our experiments because it is risky that over 50% of data is missing for the observed values and imputation should not be used in this situation [29]. In addition, in the experiments, the measurement values can be extracted in different intervals to make the data density of the dataset changeable. The dataset with the shorter sampling interval gets the higher data density. Then, we can observe the results and evaluate the algorithms. Finally, to simulate the actual working of the sensor network, the base station or data center collects the data from the sensor nodes in the form of a data stream, and the imputation algorithm deployed on it checks the data and makes imputation for the missing values. In each experiment, the imputation algorithm reads the measurement value one by one at the original rate from the subsets and makes an imputation when there is a missing value. After finishing the imputation for all missing values in the subset, the imputation result can be used to evaluate the performance of algorithms.

### 4.3. Evaluation of RMSE

*4.3.1. Experiments and Results on Intel Lab Dataset.* Firstly, the sampling interval is set to 1 min, and the missing percentages are from 5% to 50%. Next, we randomly select one of the subsets of the Intel lab dataset. Apply the MCAR method to the temperature values to get the subsets with missing values at each of the missing percentages. Then, we apply different algorithms to make the imputation for the

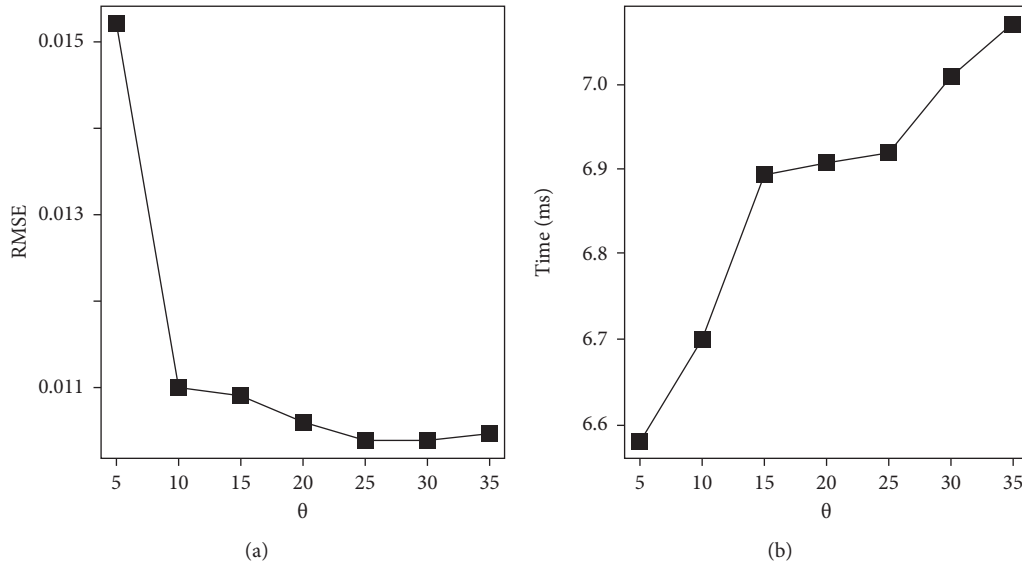


FIGURE 5: The relationship between parameter  $\theta$  and performance of VTN algorithm. (a) RMSE and ?. (b) Computational time and ?.

missing values in the stream from the subset, each one as a test case. The experiment is repeated 10 times for different subsets. Finally, the average RMSE for each case at the same missing percentage is calculated based on the test results of all cases. Similarly, we apply the MCAR method to humidity values and repeat the same experiments.

The parameter  $\theta$  is set as 15 in the VTN algorithm. To evaluate all imputation algorithms under the same condition, the spatial data from the neighbor nodes and the temporal data from other sensors on the same node are not applied in the TSNN and DESM algorithms. Therefore, in the TSNN algorithm, the spatial-temporal coefficient  $\lambda$  is set to 0, and the best  $k_t$  for the temporal nearest neighbors will be calculated and applied by the algorithm. Similarly, in the DESM algorithms, the weight parameter  $\alpha$  is set to 0.

The experiment results are shown in Figure 6.

Figure 6 shows that for all imputation algorithms, the RMSE rises as the percentage of the missing values increases. The reason is that the reliable information from the non-missing values decreases as the number of missing values increases, and the imputed values are applied in the calculation of the estimated value of the next missing value when the algorithms make the online imputation. Therefore, for imputation algorithms that depend on more than one previous value, such as LIN, MEAN, and TSNN, the performance of accuracy is dropping down and is lower than DSEM, which only requires one previous value. Among the algorithms based on the calculated neighbor value, i.e., LIN, TSNN, and VTN, LIN has a lower performance than the latter two algorithms because it does not utilize the correlation between the measurement value and its neighbors. It is worth noting that although TSNN and VTN are the algorithms applying regression to estimate the missing values, the TSNN algorithm has poorer performance because the temporal neighbor is calculated merely based on the previous values. Besides, it cannot get benefits from the spatial neighbor and the data from other sensors on the same node

because they are unavailable in the experiment. However, the VTN algorithm has the smallest RMSE in the entire range of percentages of missing values and shows its best performance. It benefits from the new algorithm design: the method to calculate the virtual temporal neighbors exploits the information from the value next to the current value, and the information of previous values is fully exploited than other algorithms. In addition, compared with temperature, the RMSE for humidity data is bigger. It is caused by the more dramatic variety of the humidity data than the temperature data, which is because of more physical factors affecting humidity than the temperature in real environments.

To evaluate the performance of algorithms in different data densities, we set the sampling intervals from 1 min to 30 mins. Before making the MCAR operation, we extract measurement values based on them to make the subsets with data density from high to low. Then, for different imputation algorithms, we follow the same steps as the previous examples and calculate the average RMSE of cases within the entire missing percentage range. The experiment results are shown in Figure 7.

Figure 7 displays that, for all imputation algorithms, with the longer intervals, the data density is getting lower, and the RMSE is fluctuating toward the bigger direction. Among all algorithms, MEAN has the poorest performance because precise information obtained from the previous values is becoming much lesser when the data density is decreased, and it has the greatest impact on the MEAN algorithm. However, similar to the previous analysis, the VTN algorithm is slightly affected and maintains its best performance among all algorithms.

*4.3.2. Experiments and Results on GreenOrbs Dataset.* Similar to the experiments on the Intel lab dataset, we make the same experiments on the GreenOrbs dataset. Because of

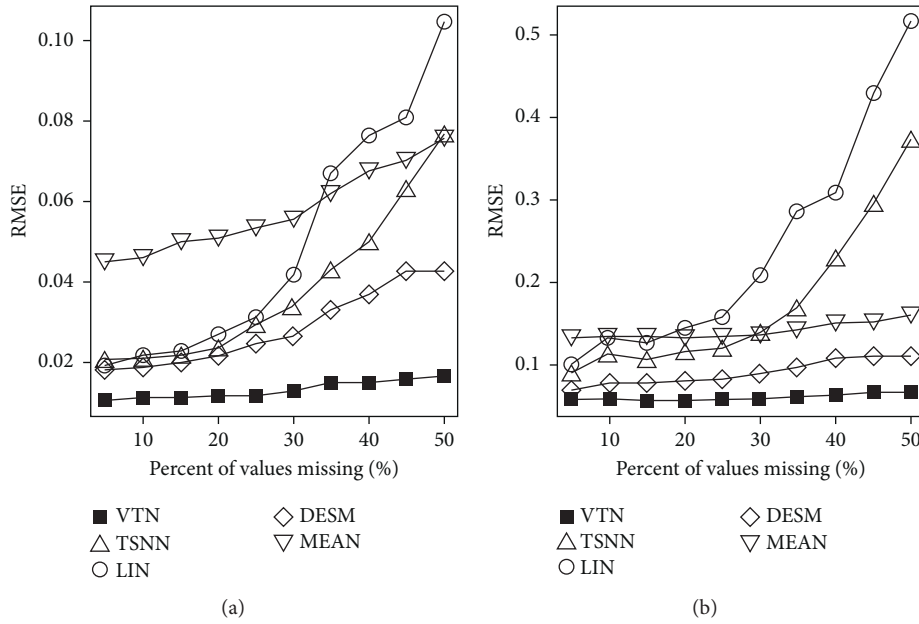


FIGURE 6: RMSE of imputation algorithms on Intel lab dataset. (a) Temperature data. (b) Humidity data.

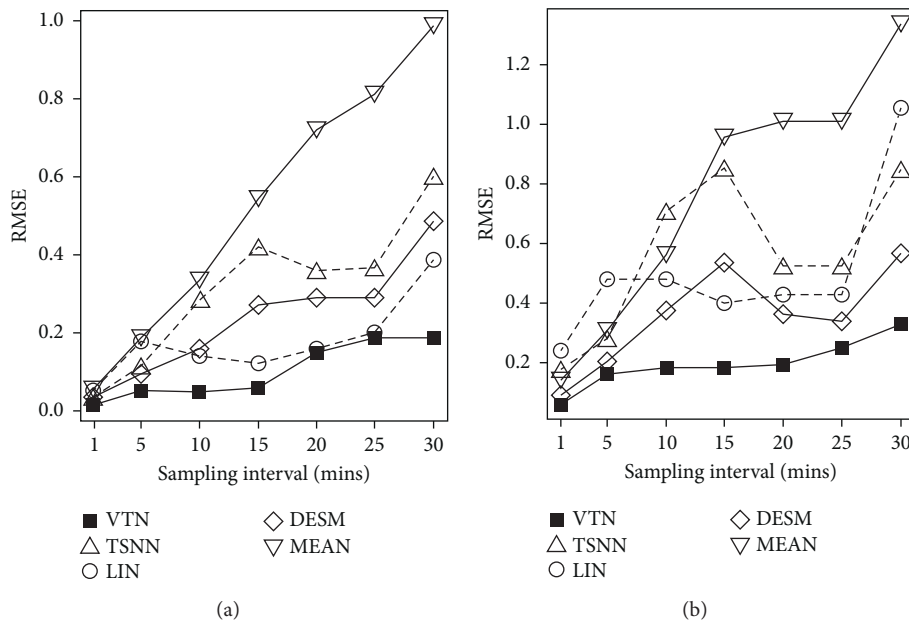


FIGURE 7: RMSE of imputation algorithms in different data density on Intel lab dataset. (a) Temperature data. (b) Humidity data.

the different data rate in the raw dataset, in experiments on the GreenOrbs dataset, the sampling interval is set to 3 mins, and the missing percentages are still from 5% to 50%.

The experiment results are shown in Figure 8.

Figure 8 demonstrates that the RMSE of all imputation algorithms becomes relatively bigger in the GreenOrbs dataset than in the Intel lab dataset. It is because of the more intensive change of the temperature and humidity data in the outdoor forest than indoors, and it is also the reason that MEAN gets relatively lower performance. However, VTN still obtains the highest performance because the neighbors

chosen based on the change rate contribute to the accurate estimated values, which is affected less severely in this situation.

Next, for evaluating the performance of algorithms in the different data density, we set the sampling intervals from 3 mins to 21 mins and make the same experiments. The results are shown in Figure 9.

Similarly, from Figure 9, we find that MEAN is still the worst one among all algorithms. The performance of VTN is getting degraded as the sampling interval is getting longer. However, it still maintains the best accuracy in RMSE.

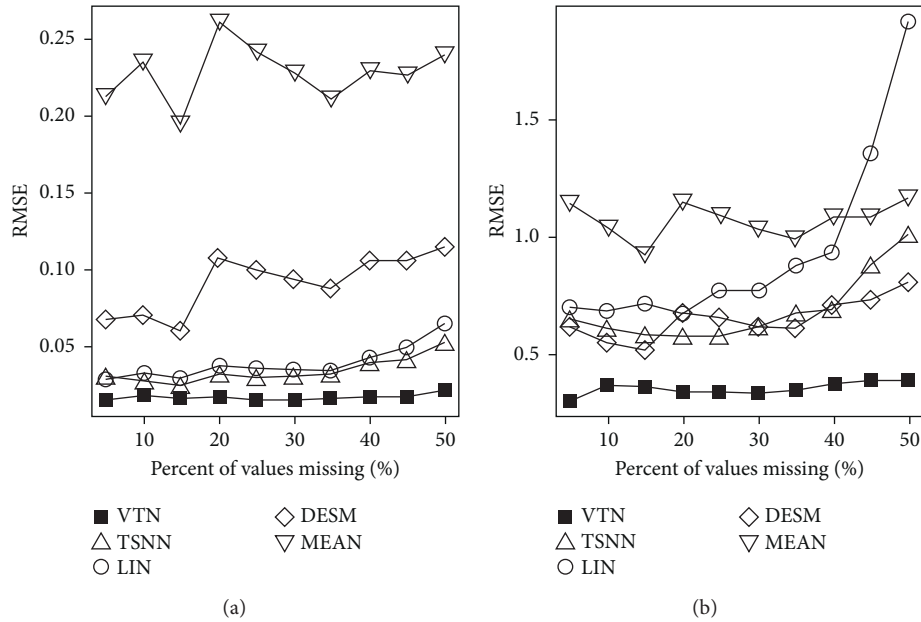


FIGURE 8: RMSE of imputation algorithms on GreenOrbs dataset. (a) Temperature data. (b) Humidity data.

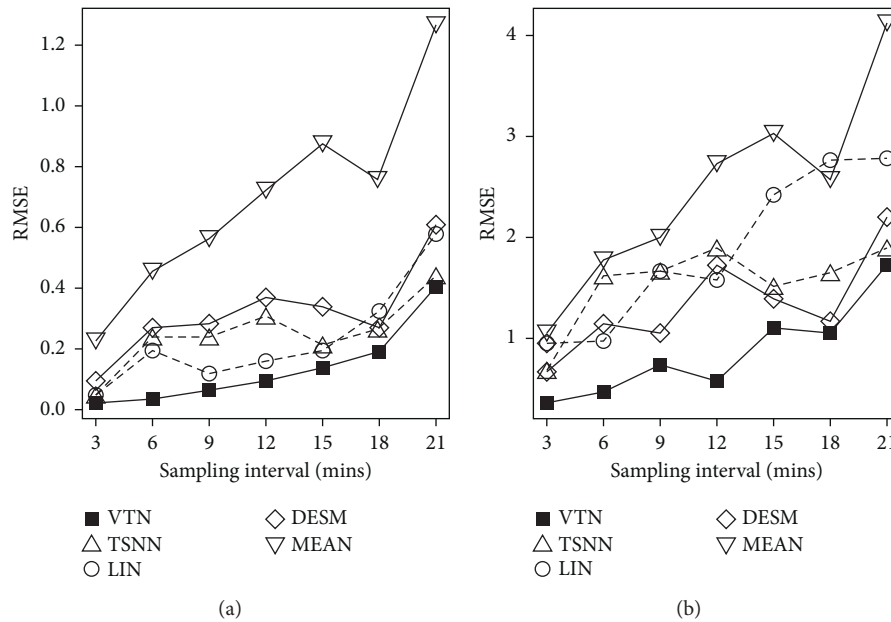


FIGURE 9: RMSE of imputation algorithms in different data density on GreenOrbs dataset. (a) Temperature data. (b) Humidity data.

**4.3.3. Experiments and Results Air Quality Dataset.** In experiments on the Air Quality dataset, the sampling interval is set to 1 hour according to the raw dataset. Same as before, the missing percentages are from 5% to 50%. We make the same experiments on the Air Quality dataset and get results as shown in Figure 10.

As shown in Figure 10, for all algorithms, the RMSE is getting bigger with the increasing percentage of missing values. Moreover, for all algorithms, the average RMSE is the biggest in the Air Quality dataset, because in the Intel lab

dataset and the GreenOrbs dataset, the intervals in which the sensor obtains the measurement are less than two minutes. However, in the Air Quality dataset, the less interval of the measurement values is one hour, which causes the difference between the two adjacent measurement values to be bigger than the one in the other two datasets.

Then, we set the sampling intervals from 1 hour to 7 hours and make the same experiments to evaluate the performance of algorithms in different data densities. The results are shown in Figure 11.



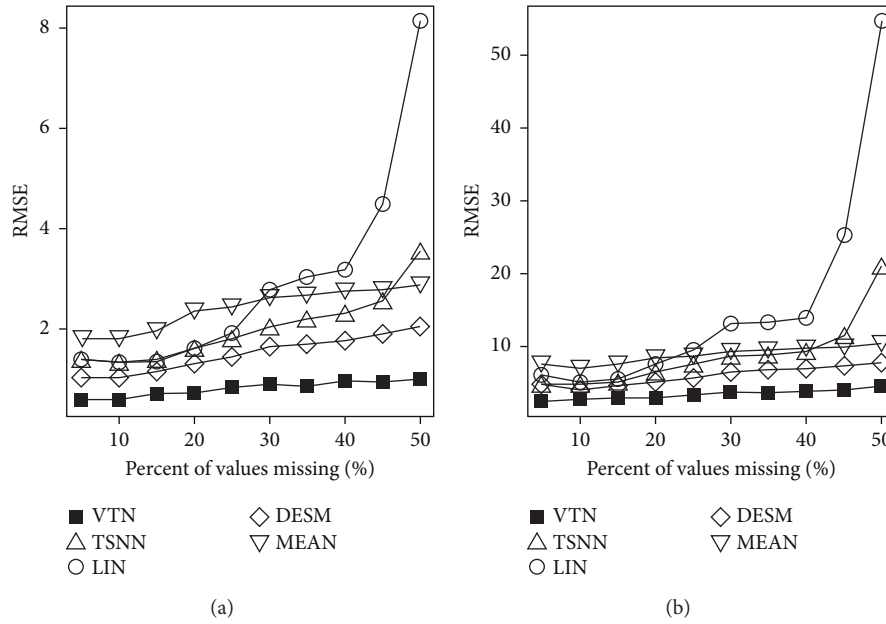


FIGURE 10: RMSE of imputation algorithms on air quality dataset. (a) Temperature data. (b) Humidity data.

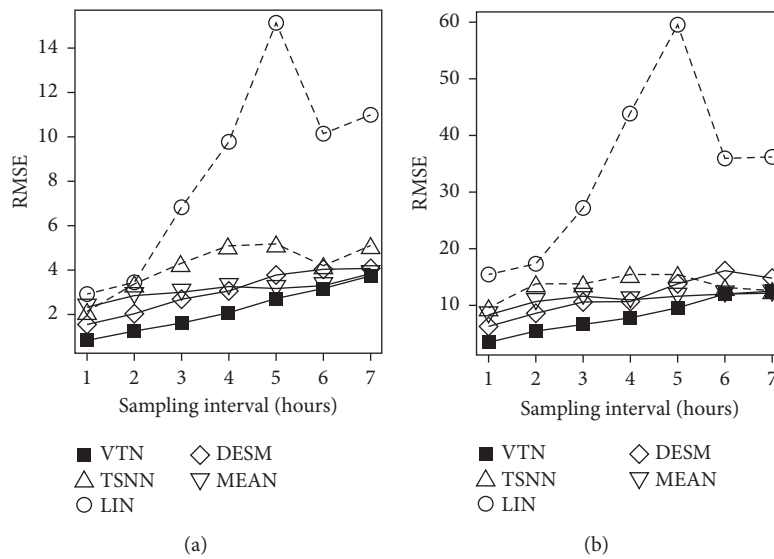


FIGURE 11: RMSE of imputation algorithms in different data densities in the Air Quality dataset. (a) Temperature data. (b) Humidity data.

It is interesting that different from the previous experiments, we can find that LIN replaces MEAN as the worst algorithm, especially when the percentage of the missing values is getting higher or the sampling interval is getting longer as shown in Figures 10 and 11, because LIN gets the greatest impact among all algorithms because of the drastically variational neighbor values, which makes the linear interpolation produce the poorest result. However, VTN is still the best algorithm with the highest imputation accuracy.

In sum, evaluated by RMSE, the above experimental results on three different datasets show that VTN gets the best accuracy in imputation.

**4.4. Evaluation of Computational Time.** In this paper, the computational time is denoted by  $T_C$ , which is the elapsed time from the imputation algorithm is triggered by a missing data to the end of the imputation. The running time of an imputation algorithm for outputting an estimated value for a missing value read in is denoted by  $T_R$ . Generally,  $T_C$  is longer than  $T_R$  because the base station must spend some time for other required system processing tasks, for example, time for data I/O, and it is denoted by  $T_S$ . Hence, the computational time  $T_C$  is the sum of  $T_R$  and  $T_S$ . Moreover, there is extra time required by some imputation algorithms, which is used to obtain the data they need. For example, to

the VTN algorithm, it needs to get the next value in the sensor data stream to make imputation for the current missing value. Hence, it must wait for the next coming data, and the delay time is denoted by  $T_D$ . Therefore, the actual total processing time for completing an imputation for a missing value, denoted by  $T_P$ , can be computed as follows:

$$T_P = T_C + T_D = T_R + T_S + T_D. \quad (20)$$

In our experiments, we do not need to know about the actual running time  $T_R$  and system time  $T_S$ . Hence, the computational time  $T_C$  is enough for us to evaluate the performance of algorithms, and it can be measured by the microbenchmark tool on the  $R$  platform in our experiments.

Firstly, we randomly select one of the subsets of the Intel lab dataset and set the sampling interval as 1 min. The interval does not affect the computational time of imputation algorithms, however, it must ensure that the number of values inside the dataset is kept the same for all experiments so we can compare the test results of the different algorithms. Next, the missing percentages are set from 5% to 50%. We apply the MCAR method to temperature values to get the subsets with the missing values at each of the missing percentages. Then, we apply different algorithms to make the imputation for the missing values in the stream, each one as a test case. The experiment is repeated 10 times for different subsets, and finally, based on the test results of all cases, the average computational time for each case at the same missing percentage is measured. As the data type is unrelated to the computational time, we do not need to repeat the same experiments on humidity data.

Similarly, we set the sampling interval as 3 mins for the GreenOrbs dataset and 1 hour for the Air Quality dataset and ensure these subsets have the same number of values. Then, we repeat the same experiments, and the results are shown in Figure 12.

In Figure 12, we can find that each of the imputation algorithms shows similar performance on three different datasets, which indicates there is no relationship between the computational time of imputation algorithm and the dataset. The time consumed by all algorithms is increasing with the percentage of values missing because of more missing values required to be imputed, while VTN and TSNN are rising sharply among them.

The bar charts describe the average computational time for a missing value with different imputation algorithms. DESM has the shortest computational time, and MEAN and LIN have relatively shorter computational times. By contrast, VTN and TSNN still have prominently longer computational times, and VTN requires the longest time for imputing a missing value.

The reason that VTN and TSNN have poorer performances in computational time is that they apply regression to calculate the estimated value for missing data, which is a time-consuming operation. In addition, computing virtual neighbors for each value in the streaming data brings more cost of computational time for the VTN algorithm and makes it the top time-consuming algorithm. Actually, the experiment results only demonstrate the computational time  $T_C$ , and we also need to consider the delay time  $T_D$  to get the

final imputation processing time  $T_P$ . Compared with VTN, the other four algorithms, namely TSNN, LIN, MEAN, and DESM, are not required to wait for getting the next value in the sensor data stream.  $T_D$  can be ignored for them, however, VTN does need it, which depends on the data stream rates on the sensor network, and it makes the final processing time  $T_P$  for the VTN algorithm longer.

## 5. Discussion

It is an essentiality of the imputation algorithms that they calculate the estimation for the missing value in some way based on the nonmissing values.

Firstly, the accuracy of imputation, which is evaluated by RMSE, is determined by the level of similarity between the estimated values and raw values if they are not missing. The only available information for estimating the operations comes from the nonmissing values. The best precise imputation algorithm should fully exploit the information hidden in the nonmissing values and build the best connection between the valuable information and the missing value. It includes two meanings: one is to effectively exploit the current information to extract the useful one for estimation. The other is to obtain extra information that helps improve the performance of estimation if possible. For the first one meaning, the MEAN algorithm utilizes the average number of previous nonmissing values, which has relatively less accuracy because of its inherent statistical deficiency, as the results shown in experiments. DESM applies the nonmissing value next to the missing value directly. Depending on the deviation level between the adjacent values, it represents different performance as the results shown in experiments. LIN applies more calculation to get neighbor values for making linear interpolation. Hence, its performance depends on the fluctuant of the temporal neighbors and becomes worse when the data density gets lower or the numbers of nonmissing values get lesser. TSNN makes further utilization on the relationship between the missing value and its temporal neighbors by regression. Compared with LIN, its performance is improved in most cases. LIN and TSNN extract more useful information from the previous nonmissing values, but because the temporal neighbors they relied on are calculated based on the nonmissing values on the time points before the time when there is a missing value, and it makes the calculated temporal neighbors less accurate, which leads to the poor preciseness of the regression in the next step. Eventually, LIN and TSNN cannot get ideal performance. They even become worse, as shown in the experiments results. However, by doing the way in the second meaning, in other words, to get extra information from the next measurement value improves the accurate of imputation in VTN algorithm. There are two crucial points in VTN, one is that the virtual temporal neighbor of a value is calculated based on the two values before and after the time point of the value, which boosts the accuracy of the virtual temporal neighbor. The other is that only the virtual temporal neighbors, which are the closest to the temporal neighbor of the missing value in values and change rates are used in the regression to calculate the

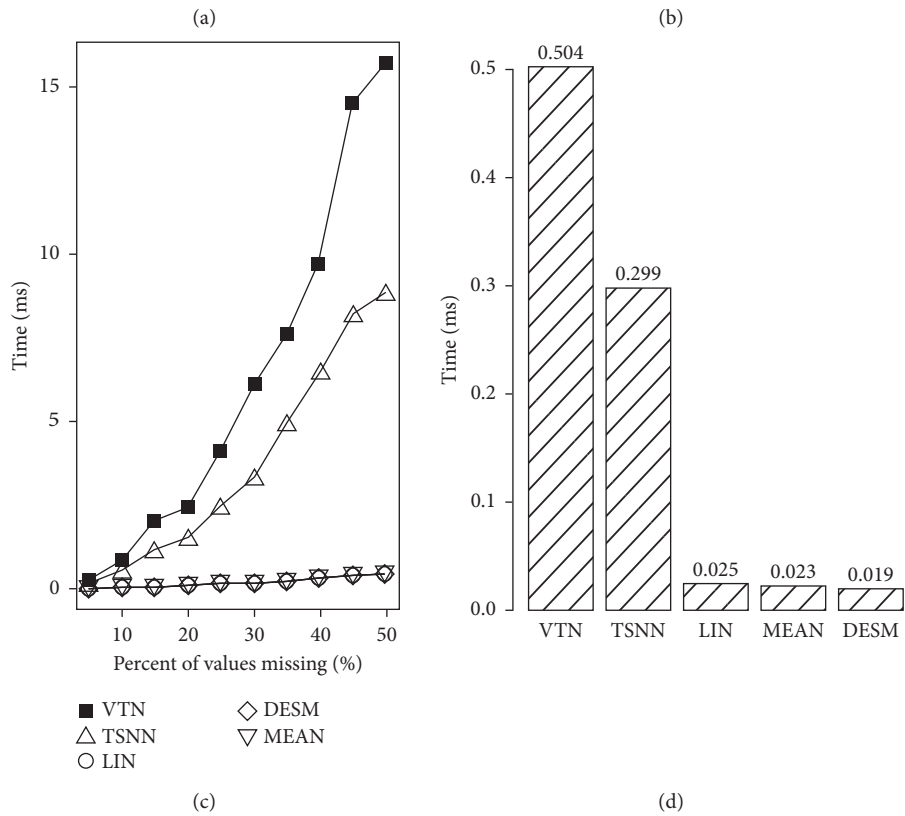
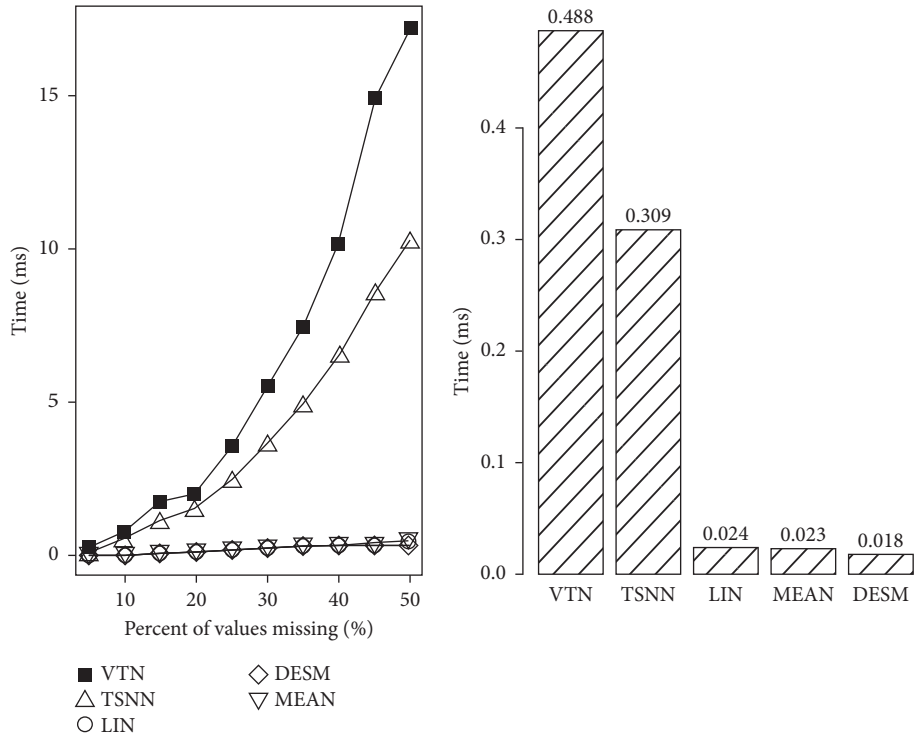


FIGURE 12: Continued.

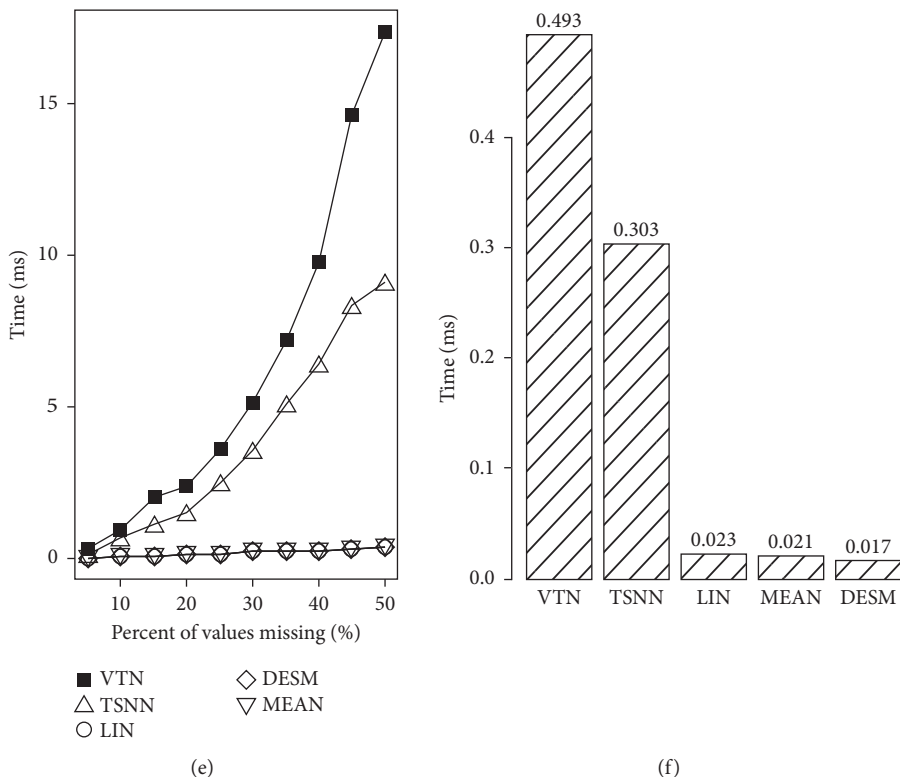


FIGURE 12: Computational time (a) (c) (e) and the average computational time for a missing value (b) (d) (f). (a) Intel lab dataset. (b) Intel lab dataset. (c) GreenOrbs dataset. (d) GreenOrbs dataset. (e) Air quality dataset. (f) Air quality dataset.

estimated value for imputing the missing value. The first point obtains extra information from the next measurement value read in, and the second point makes further exploitation of all information, which can explain the reason that VTN obtains the best performance in the accuracy of imputation in our experiments.

Secondly, the computational time is another important evaluation strategy for the imputation algorithm, especially for the online imputation of the sensor data stream. Because of more time expenses for more complex calculations than other algorithms, VTN has the longest computational time as shown in the experiment results. If considering about the delay time for waiting the next coming measurement values, the actual total processing time for VTN is quite longer than other algorithms. However, it does not mean VTN is not applicable in real circumstances. On the one side, in sensor networks, because of the limited computing power, storage, and power supply of sensor nodes, the imputation algorithm usually is deployed on the base station or data center, which has enough processing resources to support the running of the VTN imputation algorithm at a reasonable time cost. On the other hand, based on the results of experiments, the final total processing time for VTN is the sum of the computational time and the delay time. For example, on the Intel lab dataset, the original rate of the sensor data stream is 1/31 value per second, i.e., the interval between the two values is 31 s. Hence, the processing time for a missing value is approximately 31.0005 s. It indicates the delay time for obtaining the next value accounts for the most part of

processing time. Generally, in the applications of the sensor network, the acceptable response time depends on the type of application. The query application expects the shortest response time, whereas the prediction can accept a little longer response time. For simplicity's sake, we use the query application as an example. As the imputation algorithm is working continuously to impute the possible missing value and output the stream without missing values, the query application can immediately get all complete data without the missing values unless the missing value happens to be in the end point of the data that has been querying currently, which adds the extra delay time of reading one value or more than one values (if the value read in is also a missing value) into the final response time. However, the likelihood of this situation is quite low when the percentage of value missing is not too big. In fact, in real applications, the percentage of value missing is usually small in most cases. Therefore, the extra time cost brought by the delay time in VTN has relatively minor impacts on the application. Compared with minor extra processing time, the improved accuracy of imputation makes the gains outweigh the loss for applications in WSNs, which makes VTN a valuable imputation algorithm.

Finally, as a new imputation algorithm, there is still plenty of room to improve the performance of VTN. The first point is that compared with other algorithms, its computational time is relatively long, and it can be shorter if we make further optimization in the implementation of the algorithm. The next point is that we are continuing to work is

the accuracy of imputation. By designing better methods to choose more suitable virtual temporal neighbors used in the regression step, we expect to improve the imputing preciseness of the VTN algorithm in the future. In addition, in this paper, the method to choose the parameter  $\theta$  is based on experiments on limited datasets. More experiments on a larger range of datasets are required to be made for finding a better method to preset the parameter in the further study. The last point that is that we have not considered a lot about is the memory space used by the algorithm. In the current VTN, we apply extra space for storing the virtual temporal neighbors, and how to make it less without affecting the computational time should be studied in the next stage. In our research objective to make VTN an online imputation algorithm with faster running time, more imputing accuracy, and lesser memory space, we still have a lot of work to do in our further research work.

## 6. Conclusions

In this paper, we propose the VTN imputation algorithm to cope with the missing values in the sensor data stream in WSNs. Spatial information is not required for the algorithm, and only the previous temporal values and a next value reading from the stream are used to create the VTN. Next, the missing value can be estimated by regression based on the VTNs, which are closed to the VTN at the missing time point in values and change rates. The RMSE and the computational time are evaluated for the VTN imputation algorithm on three different sensor datasets. Compared with other representative algorithms, the VTN imputation algorithm presents higher imputing accuracy and acceptable time costs when it is applied to online imputation for the sensor data stream.

## Data Availability

The system parameters used to support the findings of this study are included within the article. The experiment data used to support the study is available through the web links in the references.

## Conflicts of Interest

The authors declare no conflicts of interest.

## Acknowledgments

The authors would like to thank the reviewers for their comments, which helped to improve the paper. This work is partly supported by the National Natural Science Foundation of China under Grants nos. 61872194 and 61902237, the Anhui Science and Technology Department Foundation under Grant 1908085MF207, the Postdoctoral Found of Jiangsu Province under Grant no. 2018K009B<sub>2</sub>, and the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant no. KYLX15\_0837.

## References

- [1] S. H. Shah and I. Yaqoob, "A survey: internet of things (IoT) technologies, applications and challenges," in *Proceedings of the 2016 IEEE Smart Energy Grid Engineering (SEGE)*, pp. 381–385, Oshawa, ON, Canada, August 2016.
- [2] X. Xue, J. Lu, and J. Chen, "Using NSGA-III for optimising biomedical ontology alignment," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 3, pp. 135–141, 2019.
- [3] X. Xue, P.-W. Tsai, and Y. Zhuang, "Matching biomedical ontologies through adaptive multi-modal multi-objective evolutionary algorithm," *Biology*, vol. 10, no. 12, p. 1287, 2021.
- [4] A. R. Ganguly, O. A. Omiaom, and R. M. Walker, "Knowledge discovery from sensor data for security applications," in *Learning from Data Streams* Springer, Berlin, Heidelberg, 2007.
- [5] I. F. Akyildiz, W. Weilian Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.
- [6] T. Hossain and S. Inoue, "A comparative study on missing data handling using machine learning for human activity recognition," in *Proceedings of the 2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, pp. 124–129, Spokane, WA, USA, June 2019.
- [7] J. Zhao, R. Govindan, and D. Estrin, "Computing aggregates for monitoring wireless sensor networks," in *Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications*, pp. 139–148, Anchorage, AK, USA, June 2003.
- [8] C. T. Tran, M. Zhang, P. Andraee, and B. Xue, "Multiple imputation and genetic programming for classification with incomplete data," in *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 521–528, Spokane, WA, USA, 2017.
- [9] S. De Vito, E. Massera, M. Piga, L. Martinotto, and G. Di Francia, "On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario," *Sensors and Actuators B: Chemical*, vol. 129, no. 2, pp. 750–757, 2008.
- [10] E. Elmahrawy, "Research directions in sensor data streams: Solutions and challenges," Tech. Rep. DCIS-TR-527, Rutgers University, New Brunswick, Canada, 2003.
- [11] S. Chavhan and N. A. Chavhan, "A review on data collection method with sink node in wireless sensor network," *International Journal of Distributed and Parallel systems*, vol. 4, no. 1, pp. 67–74, 2013.
- [12] R. J. A. Little and D. B. Rubin, *Statistical Analysis with Missing Data*, John Wiley & Sons, Hoboken, NJ, USA, 3rd ed. edition, 2019.
- [13] W.-C. Lin and C.-F. Tsai, "Missing value imputation: a review and analysis of the literature (2006-2017)," *Artificial Intelligence Review*, vol. 53, no. 2, pp. 1487–1509, 2020.
- [14] V. Gorodetsky, O. Karsaev, and V. Samoilov, "Direct mining of rules from data with missing values," in *Foundations of Data Mining and Knowledge Discovery, Studies in Computational Intelligence* Springer, Berlin, Germany, 2005.
- [15] Y.-T. Yan, Y.-P. Zhang, Y.-W. Zhang, and X.-Q. Du, "A selective neural network ensemble classification for incomplete data," *International Journal of Machine Learning and Cybernetics*, vol. 8, no. 5, pp. 1513–1524, 2017.



- [16] N. Jiang, "A data imputation model in sensor databases," in *Proceedings of the High Performance Computing and Communications Third International Conference*, pp. 86–96, Houston, USA, 2007.
- [17] J. Yoon, W. R. Zame, and V. Mihaela, "Estimating missing data in temporal data streams using multi-directional recurrent neural networks," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 5, pp. 1477–1490, 2018.
- [18] M. Lee, J. An, and Y. Lee, "Missing-value imputation of continuous missing based on deep imputation network using correlations among multiple IoT data streams in a smart space," *IEICE - Transactions on Info and Systems*, vol. 102, no. 2, pp. 289–298, 2019.
- [19] P. J. García-Laencina, J. L. Sancho-Gómez, and A. R. Figueiras-Vidal, "Pattern classification with missing data: a review," *Neural Computing & Applications*, vol. 19, no. 2, pp. 263–282, 2010.
- [20] L. Pan and J. Li, "K-nearest neighbor based missing data estimation algorithm in wireless sensor networks," *Wireless Sensor Network*, vol. 2, no. 2, pp. 115–122, 2010.
- [21] Y. Li, C. Ai, W. P. Deshmukh, and Y. Wu, "Data estimation in sensor networks using physical and statistical methodologies," in *Proceedings of the 28th International Conference on Distributed Computing Systems*, pp. 538–545, Beijing, China, 2008.
- [22] Y. Shao, Z. Chen, F. Li, and C. Fu, "Reconstruction of big sensor data," in *Proceedings of the 2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, pp. 1–6, New Brunswick, Canada, 2016.
- [23] L. Pan, H. Gao, H. Gao, and Y. Liu, "A spatial correlation based adaptive missing data estimation algorithm in wireless sensor networks," *International Journal of Wireless Information Networks*, vol. 21, no. 4, pp. 280–289, 2014.
- [24] X. Ren, H. Sug, and H. Lee, "A new estimation model for wireless sensor networks based on the spatial-temporal correlation analysis," *Journal of information and communication convergence engineering*, vol. 13, no. 2, pp. 105–112, 2015.
- [25] Y. Deng, C. Han, J. Guo, and L. Sun, "Temporal and spatial nearest neighbor values based missing data imputation in wireless sensor networks," *Sensors*, vol. 21, no. 5, p. 1782, 2021.
- [26] S. Madden, "Intel lab data," 2013, <http://db.csail.mit.edu/labdata/labdata.html>.
- [27] GreenOrbs. Available online: <http://www.greenorbs.org/>.
- [28] Microbenchmark. Available online: <https://cran.r-project.org/package=microbenchmark>.
- [29] G. D. Garson, *Missing Values Analysis and Data Imputation*, Statistical Associates Publishers, Asheboro, NC, USA, 2015.

## Research Article

# Basic Research on Ancient Bai Character Recognition Based on Mobile APP

Zeqing Zhang <sup>1,2</sup>, Cuihua Lee,<sup>1</sup> Zuodong Gao <sup>1</sup>, and Xiaofan Li <sup>1</sup>

<sup>1</sup>Xiamen University, Amoy, China

<sup>2</sup>West Yunnan University of Applied Sciences, Dali, China

Correspondence should be addressed to Zeqing Zhang; 313460472@qq.com

Received 8 September 2021; Revised 9 November 2021; Accepted 6 December 2021; Published 31 December 2021

Academic Editor: Xingsi Xue

Copyright © 2021 Zeqing Zhang 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.

Bai nationality has a long history and has its own language. Limited by the fact that there are fewer and fewer people who know the Bai language, the literature and culture of the Bai nationality begin to lose rapidly. In order to make the people who do not understand Bai characters can also read the ancient books of Bai nationality, this paper is based on the research of high-precision single character recognition model of Bai characters. First, with the help of Bai culture lovers and related scholars, we have constructed a data set of Bai characters, but limited by the need of expert knowledge, so the data set is limited in size. As a result, deep learning models with the nature of data hunger cannot get an ideal accuracy. In order to solve this issue, we propose to use the Chinese data set which also belongs to Sino-Tibetan language family to improve the recognition accuracy of Bai characters through transfer learning. In addition, we propose four transfer learning approaches: Direct Knowledge Transfer (DKT), Indirect Knowledge Transfer (IKT), Self-coding Knowledge Transfer (SCKT), and Self-supervised Knowledge Transfer (SSKT). Experiments show that our approaches greatly improve the recognition accuracy of Bai characters.

## 1. Introduction

Bai nationality has a long history, splendid culture, and a population of more than one million. Most of them live in Dali Bai Autonomous Prefecture of Yunnan, and the rest are distributed in all parts of Yunnan, Bijie Prefecture of Guizhou, Liangshan Prefecture of Sichuan, Sangzhi County of Hunan, etc. They have their own unique language. Bai language is not only the common communication language of Bai people, but also an important link to condense national emotion and an important carrier of Bai culture development. As the vocabulary, pronunciation and grammar of Bai characters are all Chinese and Tibeto Burmese. The language structure of Bai characters has very important academic value and has been widely concerned by Chinese and national language circles at home and abroad for a long time. For the Bai nationality, whose literature is extremely scarce, its historical and cultural value is self-evident.

In order to promote Bai culture, it is important that people who do not understand the Bai characters can also read the historical documents of Bai nationality or Bai characters

on stone steles. It is urgent to study and proposed an automatic model that can recognize the single Bai characters. In this way, once we meet an unknown Bai character, we can take a picture, then use the model recognition, and finally give the explanation of the word.

With the renaissance of neural networks and deep learning, tremendous breakthroughs have been achieved on various recognition tasks [1–6]. Therefore, we consider using deep learning models to do the word recognition of Bai characters. First, we construct a single word data set of Bai characters, which is a handwritten data set by Bai people and culture researches. Due to the requirement for specialized knowledge, the cost of constructing and labeling this data set is high.

Given the data set, we directly train traditional and recent deep learning classification models [1, 7, 8] on this data set, but we find that their performance is not ideal. This is because the success of depth models needs a lot of data support, and the data-hunger nature of depth models leads to their poor performance when there is less data. Because of the need of a lot of expert knowledge, our data set cannot be constructed as large as the traditional classification data set [9–11].

In order to solve this problem, we find that Chinese and Bai language have a high degree of similarity, both belong to the Sino-Tibetan language family (see Figure 1). Therefore, we propose that we can use the way of knowledge transfer [12–15] to transfer a large amount of Chinese knowledge to Bai language, so as to obtain a better accuracy in Bai language. We have designed four methods of knowledge transfer: Direct Knowledge Transfer (DKT), Indirect Knowledge Transfer (IKT), Self-coding Knowledge Transfer (SCKT), and Self-supervised Knowledge Transfer (SSKT).

DKT is a classic idea of knowledge transfer. First, the model is pretrained on the Chinese character data set, then the feature extraction module of the model is used as the parameter initialization of the Bai character recognition network, and finally, the Bai character recognition network is fine-tuned on the Bai character data set. The advantage of this method is that the idea is direct and the implementation is simple, but the disadvantage of this method is also very obvious, that is, the Chinese character label does not contain any semantic information, and it is difficult to guarantee how much knowledge extracted by this hard label can be transferred to Bai characters.

IKT is a method to ensure the quality of knowledge transfer. It is noted that both Chinese and Bai language are composed of 32 basic strokes, just like English is composed of 26 letters. Therefore, the number of basic strokes of each Chinese character is counted, and the number of basic strokes is used as a soft label to train the network. In this way, the network can directly mine the common knowledge of Chinese and Bai language, instead of mining the knowledge through a classification task, so that the knowledge mined by the network can be better transferred to the Bai language.

SCKT and SSKT are two unsupervised knowledge transfer methods. The unlabeled data set is easier to obtain and has lower cost, so the unsupervised knowledge transfer method will be more practical. SCKT is to train a self-encoder [16–18] with Chinese data set and then use the encoded part as the feature extraction part of Bai character recognition network. SSKT uses the method of comparative learning [19–21] to let the data automatically mine the potential knowledge. The biggest advantage of these two methods is that no tags are needed for Chinese data sets, but the disadvantage is that it is difficult to guarantee how much knowledge acquired by these unsupervised methods can be used for knowledge transfer. It is also found that the accuracy of unsupervised knowledge transfer is lower than that of supervised knowledge transfer.

Our main contributions are fourfold: (1) We build a Bai character data set. (2) We propose four methods of knowledge transfer, DTK, ITK, SCKT, and SSKT, to transfer the knowledge of Chinese characters to Bai characters. (3) The four methods proposed in this paper have greatly improved the recognition accuracy of Bai characters. (4) The research could benefit the development of a mobile APP for recognition of Bai characters.

## 2. Materials and Methods

*2.1. Notations.* In order to improve the recognition performance of the model, we consider using transfer learning

[12–15, 22]. Transfer learning is an ability of a system to recognize and apply knowledge and skills learned in previous domains/tasks to novel domains/tasks. Specifically, let the domain be denoted as  $D = \{\mathcal{X}, P(X)\}$ , where  $\mathcal{X}$  represents the feature space,  $P(X)$  represents the marginal probability distribution, and  $X \in \mathcal{X}$ . And we can define the task as  $T = \{\mathcal{Y}, f(\cdot)\}$ , where  $\mathcal{Y}$  represents the label space and  $f(\cdot)$  represents the target prediction function.

The main problem of this paper is how to use the knowledge of Chinese characters to improve the recognition ability of Bai characters. Obviously, the novel domain is composed of Bai characters, which can be defined as  $D^b = \{\mathcal{X}^b, P(X^b)\}$ . The novel task is also certain, that is, to predict the label of Bai characters. We define the novel task as  $T^b = \{\mathcal{Y}^b, f^b(\cdot)\}$  where  $f^b : \mathcal{X}^b \rightarrow \mathcal{Y}^b$ . Similarly, the previous domain is composed of Chinese characters, which is defined as  $D^c = \{\mathcal{X}^c, P(X^c)\}$ . And the design of the previous task  $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$  directly determines the quality of the knowledge extracted from Chinese characters. The design of the previous task is also the focus of this paper. Finally, in this paper, we give the design of four kinds of previous tasks.

*2.2. Transfer Learning Approaches.* We divide transfer learning approaches into supervised and unsupervised.

Two approaches were designed as supervised, Direct Knowledge Transfer (DKT) and Indirect Knowledge Transfer (IKT); DKT directly uses Chinese character label as the training task while IKT uses common Chinese and Bai character attributes as a knowledge transfer bridge. In addition, we also designed two unsupervised transfer learning approaches. One is Self-coding Knowledge Transfer (SCKT). As the name suggests, this method uses self-encoder [16–18] to extract low-frequency information of characters, that is, commonness. The other is Self-supervised Knowledge Transfer (SSKT). Thanks to the prosperity of self-supervised [23–26], self-supervised proposes a method that allows data to monitor themselves to extract features. Contrastive learning [19–21] is a promising way effectively extracts features used to distinguish different categories from data.

The details of these four approaches are as follows.

*Direct Knowledge Transfer.* The idea and implementation of this approach is very direct. Because the target task  $T^b$  is a classification, the most intuitive way is to design the source task also as a classification problem. That is,  $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$  where  $\mathcal{Y}^c$  represents the label space of Chinese characters and  $f^c : \mathcal{X}^c \rightarrow \mathcal{Y}^c$ . Suppose  $P = f(X^c)$ , where  $P = [p_1, p_2, \dots, p_n]$  represents the probability that an example is classified into each of the  $n$  possible Chinese characters. Then, the training loss of this approach using source domain  $T^c$  using previous domain  $D^c$  is as follows:

$$\mathcal{L} = -\log \frac{\exp(p_k)}{\sum_{i=1}^{i=n} \exp(p_i)}, \quad (1)$$

where  $p_k$  represents the the probability of the correct label.

The advantage of this approach is that it is straightforward with a simple implementation. The fundamental purpose of this approach is to separate Chinese characters,

Bai characters	:	廳	鷺	吐	斲	梲	哧	奪
Chinese	:	春	雀	上	也	打	不	不得

FIGURE 1: Comparison between Chinese characters and Bai characters.

so there is a part of the method focused on learning the differences between Chinese characters. Although the characteristics between Bai and Chinese characters are different, knowledge learned in a task can be transferred to the target task.

*Indirect Knowledge Transfer.* Since Chinese character label does not contain any semantic information, we design a label containing semantic information. It is observed that Chinese and Bai characters are composed of 32 basic strokes, as shown in Figure 2. Intuitively, we can use these 32 strokes as soft labels to better transfer the knowledge of Chinese characters to Bai characters. That is,  $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$  where  $\mathcal{Y}^c$  represents the label space of 32 basic strokes. If  $Y = [y_1, y_2, \dots, y_{32}] \in \mathcal{Y}^c$ , then  $Y$  is a vector with a length of 32, and  $y_i$  represents the number of the  $i$ -th basic strokes. Then, the loss of training this previous task  $T^c$  using previous domain  $D^c$  is as follows:

$$\mathcal{L} = \|Y - f(X)\|_2^2. \quad (2)$$

The advantage of this approach is that all labels contain rich semantic information, which is shared by Chinese and Bai characters. In this way, the knowledge of 32 basic strokes extracted from Chinese characters can be transferred to Bai characters. The disadvantage is that we need to label each Chinese character with 32 basic strokes, which requires a certain amount of extra work.

*Self-coding Knowledge Transfer.* In fact, the above two approaches need annotated Chinese characters, where annotation inevitably brings a lot of work. Since a lot of unlabeled examples of Chinese characters are available, a natural idea is knowledge from unlabeled Chinese characters. First, we consider the classical unsupervised learning method: self-encoder, which can learn to compress high-dimensional data into low dimensional without losing information as much as possible. That is,  $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$  where  $\mathcal{Y}^c = \mathcal{X}^c$  and  $f^c(\cdot)$  is a structure that first encodes and compresses the data and then decodes and restores the data. Then, the loss of training of this approach  $T^c$  using previous domain  $D^c$  is as follows:

$$\mathcal{L} = \|X - f(X)\|_2^2. \quad (3)$$

However, there is no guarantee that low-frequency information is the effective knowledge to aid Bai character recognition. This task is similar to word2vec [27]; in low latitude space, similar words will still be close together, so it is still an effective method.

*Self-supervised Knowledge Transfer.* Self-supervised learning [23–26] has gained great attention in recent years

because it can automatically extract knowledge in data. Among them, contrastive learning [19–21] has made surprising progress. Therefore, using the recent comparative learning method MoCo [28] to automatically extract the knowledge of Chinese characters has become a natural choice. That is,  $T^c = \{\mathcal{Y}^c, f^c(\cdot)\}$  where  $\mathcal{Y}^c = [k^+, k_1^-, k_2^-, \dots, k_n^-]$ .  $k^+$  represents that the positive sample used in contrastive learning is usually obtained from the same picture using different data expansion methods, and the others represent negative samples, which are obtained from other pictures. Then, the loss of training approach  $T^c$  using source domain  $D^c$  is as follows:

$$\mathcal{L} = -\log \frac{\exp(X \cdot k^+)}{\exp(X \cdot k^+) + \sum_{i=1}^{i=n} \exp(X \cdot k_i^-)}. \quad (4)$$

The features extracted by MoCo [28] have achieved encouraging results in many tasks, such as image classification [9] and target detection [29]. The advantage of this approach is that it can inherit this powerful feature extraction ability. However, the training of comparative learning needs larger batch size, which has higher requirements for hardware cost. At the same time, it is difficult to guarantee the quality of the knowledge extracted by comparative learning, which can only be verified by downstream tasks.

*2.3. Model Training.* After learning a model in the source domain, the known can be transferred to the target domain 2.3.

## 3. Experimental Results

### 3.1. Experimental Setup

*3.1.1. Data Sets.* We build a large data set with 400 Bai characters. Because there is a certain overlap between Bai characters and Chinese characters, we only select Bai characters which are quite different from Chinese ones to compose and build this data set. There are about 2,000 samples for each word and character, all written by Bai people and Bai culture lovers. We split the data set into 50. In addition, we also collected a large data set of Chinese characters. The data set consists of 509 Chinese characters, each of which has about 50 samples. In order to Indirect Knowledge Transfer, we also label each Chinese character with 32 strokes.

*3.1.2. Evaluation Protocol.* We evaluate the proposed method in terms of the average per-class Top-1 accuracy (ACC).



FIGURE 2: 32 basic strokes of Chinese characters and Bai characters.

```

Require: Chinese character data set  $D^c$  and Bai character data set  $D^b$ .
Ensure: Function  $f^b(\cdot)$  for Bai character classification.
Initialize the parameters of both  $f^c(\cdot)$  and  $f^b(\cdot)$ .
1: while the  $f^c(\cdot)$  does not converge do
2:   for samples in  $D^c$  do
3:     Optimize  $f^c(\cdot)$  by Eq.(1) or Eq.(2) or Eq.(3) or Eq.(4).
4:   end for
5: end while
6: The parameters of feature extraction part in  $f^b(\cdot)$  are replaced by those in  $f^c(\cdot)$ .
7: while the  $f^b(\cdot)$  does not converge do
8:   for samples in  $D^b$ 
9:     Optimize  $f^b(\cdot)$  by Cross entropy loss.
10:  end for
11: end while

```

ALGORITHM 1: Proposed approach.

**3.1.3. Classification Model.** We use three classical classification models: AlexNet [8], VGG19 [7], and ResNet101 [1]. Comparing multiple models, we can analyze that our method is effective and has strong generalization ability.

**3.1.4. Implementation Details.** We use SGD optimizer with learning rate (lr) = 0.01 and a batch size of 64 to train DKT, IKT, and SCKT. On the other hand, we use SGD optimizer with lr = 0.001 and a batch size of 512 to train SSKT. Finally, we use SGD optimizer with lr = 0.01 and a batch size of 64 to train  $f^b(\cdot)$ . We use StepLR learning rate adjustment strategy, where the learning rate every 20 epochs becomes 0.8 of the original.

**3.2. Accuracy Analysis.** The accuracy comparison of the transfer learning approaches is shown in Table 1.

We observe that the proposed IDK achieves significant improvements over the other approaches. On the three CNN models, the accuracy is 12.01%, 9.56%, and 10.00% higher than that without transfer learning. At the same time, this training strategy of transfer learning is the most accurate

TABLE 1: Accuracy comparison of different CNNs and transfer learning approaches. No means using Bai characters to train the model directly, and transfer learning is not used.

CNN	No	DKT	IKT	SCKT	SSKT
AlexNet	73.16	83.42	85.17	74.82	77.94
VGG19	78.28	82.92	87.84	78.63	80.21
ResNet101	78.54	87.82	88.54	80.41	82.09

of the four approaches we proposed. This fully shows that our design of 32 basic strokes as soft labels can transfer the knowledge learned from Chinese characters to Bai characters. Although the 32 basic stroke label does not consider the structure and position, it is still a very effective means of transferring learning based on results.

The second high accuracy was obtained by DKT. It uses hard labels directly, that is, labels for each word, to pretrain the models. Although this label does not contain any semantic information, the model still extracts relevant features that



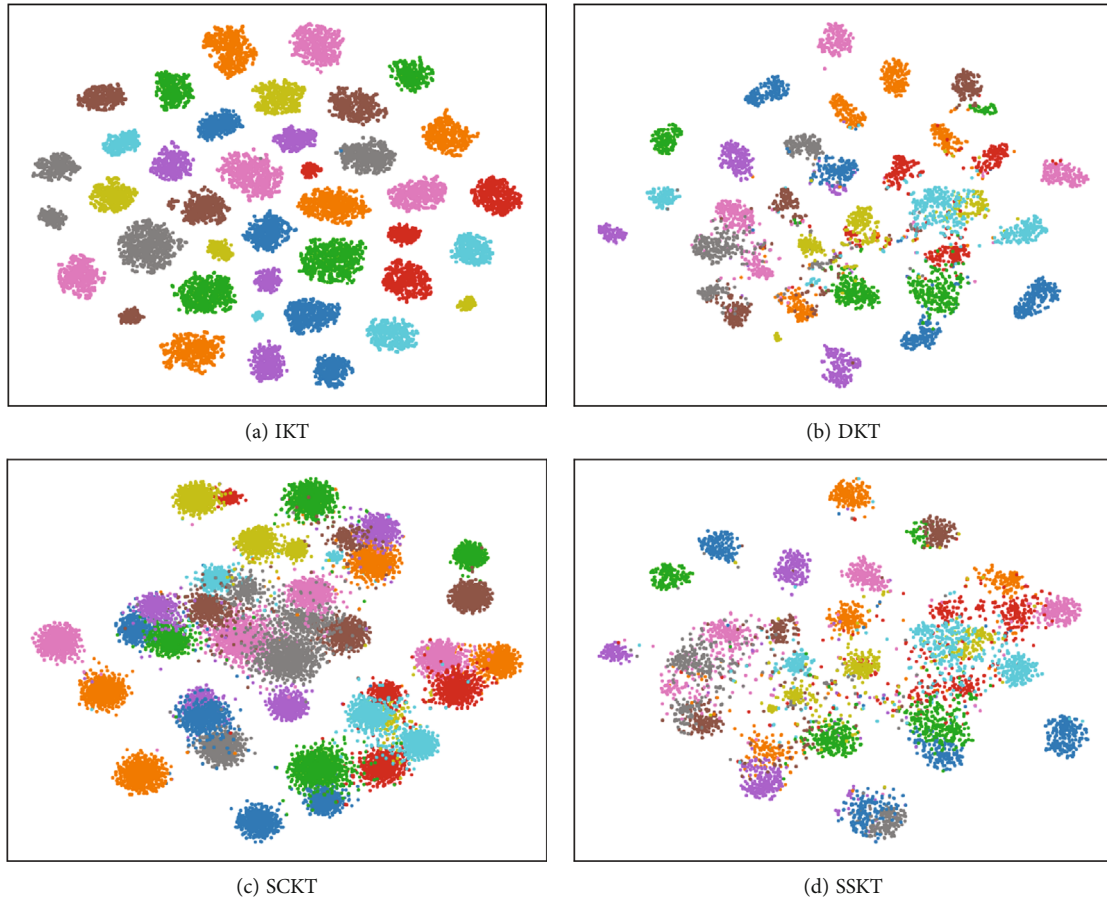


FIGURE 3: Different methods of feature visualization results.

can be used for knowledge transfer in the pretraining. On the three models, the accuracy is 10.26%, 4.64%, and 9.28% higher than that without transfer learning. The results showed that this approach can also bring a good improvement in the accuracy, but there is a certain lack of interpretability of the knowledge transferred to the target task.

The accuracy of unsupervised transfer learning approaches SCKT and SSKT is lower than that of supervised approaches. However, it is also an excellent solution if the data set is difficult to get annotation. The essence of SCKT is to seek a low latitude compression of data. Obviously, most of the knowledge used for compression is not directly transferable to another task, so the improvement in accuracy is not obvious. On the three models, the accuracy is 1.66%, 0.35%, and 1.87% higher than that without transfer learning. In fact, compared with the Bai character training model directly, the accuracy is not greatly improved.

Although SSKT cannot be compared to supervised approaches, it was significantly better than SCKT. On the three models, the accuracy is 4.78%, 1.93%, and 3.55% higher than that without transfer learning. Although the latest comparative learning method has been able to compare with the full supervision method, it needs a huge data set to bring. This is also the reason why our SSKT method is inferior to the full supervision method. In many cases, it is difficult for us to obtain a large number of unlabeled data. At that time, this method is the most suitable. Of course, this

method requires additional calculation cost for hardware, and it is also a disadvantage that cannot be ignored.

**3.3. Feature Visualization Analysis.** In order to further illustrate the effect of these four approaches, we directly use the pretrained network to extract the features of 40 Bai characters. Then, t-SNE [30] algorithm is used to visualize these features, as shown in Figure 3. It can be seen that, after pretraining with the IKT, the extracted features can be well distinguished even if there is no fine-tuning in the Bai character data set. Although the features extracted by DKT method have a good degree of aggregation within classes, there is some overlap between classes. Although some of the features extracted by SCKT and SSKT can be well distinguished, most of them will overlap each other. Through the visualization results, we have a more intuitive understanding of the differences between the four methods, and also explain why IKT method is better than other methods.

**3.4. Convergence Speed Analysis.** In Figure 4, we show the difference of convergence speed of different CNNs. It can be observed that the convergence speed of the CNNs is greatly improved after the application of the knowledge transfer methods. Without the use of knowledge transfer, the CNNs will go through multiple epochs before it starts to converge. However, after the use of the knowledge transfer method, the CNNs will converge from the beginning of

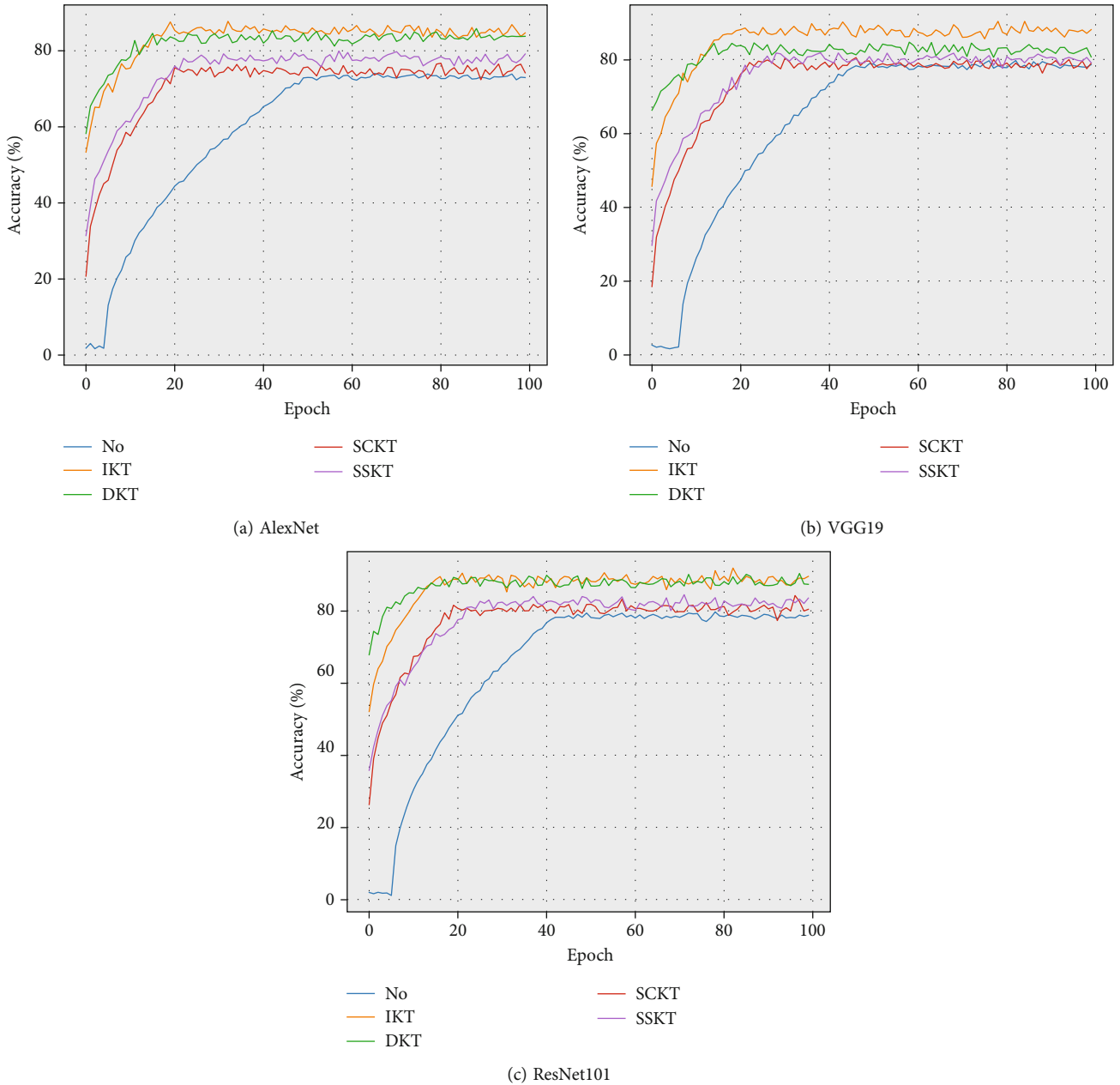


FIGURE 4: Convergence speed of different methods.

training. This is because the CNN has recognize Bai characters. In other words, the CNN has a good initial parameter, so it can converge quickly.

#### 4. Conclusion

Bai nationality, as a nation with a long history in China, not only has its own language but also has created brilliant culture. However, with the development of the times, because fewer and fewer people know Bai characters, Bai culture is dying out. In order to make people who love Bai culture and related researchers can read Bai literature smoothly, this paper mainly studies how to train a high-precision Bai character recognition CNNs. First, we build a data set of Bai characters, but limited by the need of

expert knowledge, so the data set is limited in size. As a result, those depth models that need a lot of data-driven cannot achieve satisfactory results on this data set. In order to solve this problem, we propose to use the Chinese data set which also belongs to Sino-Tibetan language family to help improve the recognition accuracy of Bai characters through knowledge transfer. In addition, we propose four methods of knowledge transfer: Direct Knowledge Transfer (DKT), Indirect Knowledge Transfer (IKT), Self-coding Knowledge Transfer (SCKT), and Self-supervised Knowledge Transfer (SSKT). Sufficient experiments not only show that our method can greatly improve the recognition accuracy of Bai characters but also show the advantages of our method from the visualization and convergence speed.

## Data Availability

We build a large data set of Bai characters. There are a total of 400 Bai characters. Because there is a certain overlap between Bai characters and Chinese characters, we only select Bai characters which are quite different from Chinese characters to build this data set. There are about 2,000 samples for each word, all written by Bai people and Bai culture lovers. The training set and the test set are about one to one. At present, the data set is still private and will be made public later.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, <https://arxiv.org/abs/1512.03385>.
- [2] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: deep learning on point sets for 3d classification and segmentation," 2016, <https://arxiv.org/abs/1612.00593>.
- [3] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," 2015, <https://arxiv.org/abs/1506.01497>.
- [4] X. Xingsi, W. Xiaojing, J. Chao, M. Guojun, and Z. Hai, "Integrating sensor ontologies with global and local alignment extractions," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6625184, 10 pages, 2021.
- [5] X. Xue and J. Zhang, "Matching large-scale biomedical ontologies with central concept based partitioning algorithm and adaptive compact evolutionary algorithm," *Applied Soft Computing*, vol. 106, article 107343, 2021.
- [6] X. Xue and J. Chen, "Matching biomedical ontologies through compact differential evolution algorithm with compact adaptation schemes on control parameters," *Neurocomputing*, vol. 458, pp. 526–534, 2021.
- [7] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for natural language processing," 2016, <https://arxiv.org/abs/1606.01781>.
- [8] S. Targ, D. Almeida, and K. Lyman, "Resnet in resnet: Generalizing residual architectures.," 2016, CoRR abs/1603.08029.
- [9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems 25*, pp. 1106–1114, 2012.
- [10] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "Emnist: an extension of mnist to handwritten letters," 2017, <https://arxiv.org/abs/1702.05373>.
- [11] M. Swofford, "Image completion on CIFAR-10," 2018, <https://arxiv.org/abs/1810.03213>.
- [12] M. Chen, Z. E. Xu, K. Q. Weinberger, and F. Sha, "Marginalized denoising autoencoders for domain adaptation," 2012, <https://arxiv.org/abs/1206.4683>.
- [13] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: a deep learning approach," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 513–520, Bellevue, Washington, USA, 2011.
- [14] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *CVPR*, pp. 1717–1724, IEEE Computer Society, 2014.
- [15] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 3320–3328, 2014.
- [16] Y. J. Fan, "Autoencoder node saliency: selecting relevant latent representations," 2017, <https://arxiv.org/abs/1711.07871>.
- [17] H. Ishfaq, A. Hoogi, and D. Rubin, "TVAE: triplet-based variational autoencoder using metric learning," 2018, <https://arxiv.org/abs/1802.04403>.
- [18] Q. Li, X. Zheng, and X. Wu, "Collaborative autoencoder for recommender systems," 2017, <https://arxiv.org/abs/1712.09043>.
- [19] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, pp. 1597–1607, 2020.
- [20] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, <https://arxiv.org/abs/1807.03748>.
- [21] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *ECCV (11). Lecture Notes in Computer Science, vol. 12356*, A. Vedaldi, H. Bischof, T. Brox, and J. M. Frahm, Eds., pp. 776–794, Springer, 2020.
- [22] S. Mirsamadi and J. H. L. Hansen, "Multi-domain adversarial training of neural network acoustic models for distant speech recognition," *Speech Communication*, vol. 106, pp. 21–30, 2019.
- [23] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," 2018, <https://arxiv.org/abs/1808.06670>.
- [24] I. Misra, C. L. Zitnick, and M. Hebert, "Shuffle and learn: unsupervised learning using temporal order verification," in *ECCV (1). Lecture Notes in Computer Science, vol. 9905*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., pp. 527–544, Springer, 2016.
- [25] J. Wu, X. Wang, and W. Y. Wang, "Self-supervised dialogue learning," in *ACL (1)*, A. Korhonen, D. R. Traum, and L. Márquez, Eds., pp. 3857–3867, Association for Computational Linguistics, 2019.
- [26] Z. Wu, Y. Xiong, S. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance-level discrimination," 2018, <https://arxiv.org/abs/1805.01978>.
- [27] X. Rong, "word2vec parameter learning explained," 2014, <https://arxiv.org/abs/1411.2738>.
- [28] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9726–9735, 2020.
- [29] P. R. P. A. P. S. T. Vinod, "Object detection an overview," *International Journal of Trend in Scienti\_c Research and Development*, vol. 3, no. 3, pp. 1663–1665, 2019.
- [30] N. Rogovschi, J. Kitazono, N. Grozavu, T. Omori, and S. Ozawa, "T-distributed stochastic neighbor embedding spectral clustering," in *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 1628–1632, Anchorage, AK, USA, 2017.

## Research Article

# Improved Joint Optimization Design for Wireless Sensor and Actuator Networks with Time Delay

Lihan Liu <sup>1,2</sup>, Yuehui Guo <sup>1</sup>, Yang Sun <sup>1</sup>, Zhuwei Wang <sup>1</sup>, Enchang Sun <sup>1</sup>,  
and Yanhua Sun<sup>1</sup>

<sup>1</sup>Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China

<sup>2</sup>School of Information, Beijing Wuzi University, Beijing, Beijing 101149, China

Correspondence should be addressed to Yang Sun; sunyang@bjut.edu.cn

Received 24 September 2021; Accepted 9 December 2021; Published 30 December 2021

Academic Editor: Kingsi Xue

Copyright © 2021 Lihan Liu 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 wireless communication technology, the newest development of wireless sensor and actuator networks (WSANs) provides significant potential applications for various real-time scenarios. Currently, extensive research activities have been carried out in the field of efficient resource management and control design. However, the stability of the controlled plant and the efficiency of network resources are rarely considered collaboratively in existing works. In this paper, in order to enhance the control stability and improve the power consumption efficiency for the WSAN, a novel three-step optimization algorithm jointly designing the control strategy and transmission path routing is proposed when the time delay is considered. First, the minimum hop routing algorithm is used to obtain the set of candidate transmission paths. Then, the optimal control signals for each candidate transmission path can be iteratively derived with a backward recursion method. Finally, the best transmission path is determined under the optimal control strategy to achieve the joint optimization design. The effectiveness of the proposed joint optimization algorithm is verified by the simulations of the application in the power grid system.

## 1. Introduction

Wireless communication technology plays a vital role in various communication networks to promote the progress of modern science and technology [1–3]. It is believed that the emergence and development of the wireless technology are revolutionizing the traditional wired communication. Wireless sensor and actuator networks (WSANs), typically consisting of sensors, controllers, and actuators, is one of the most critical wireless communication applications [4]. With the characteristic of spatially distributed nodes, WSAN has the capability of information perception, transmission, analysis, and processing to meet the demands for both high reliability and low latency. Efficient information sharing and energy consumption management can be realized in the closed-loop feedback control network with proper resource allocation and transmission path routing. Currently, WSAN has already become an attractive

research topic in many application areas, such as Internet of Thing (IoT), intelligent transportation, automotive industry, and smart healthcare [5–10].

WSAN takes advantage of the wireless network to provide information sharing, resource utilization, and plant control. However, there are still some challenges introduced especially with the increasing number of connected devices and sensor nodes [11–14]. One of the problems is the time delay caused by wireless communication which may significantly degrade the system performance and even cause instability [15]. Many works have been done to alleviate the influences of the delays. In [16], the network-induced short delay is analyzed for addressing the real-time system control problem. It uses the stochastic control theory to analyze the optimal state feedback for stabilization in discrete-time domain. The authors in [17] study an optimal controller for network control systems to maintain stability under the long time delay caused by wireless communication. An



overview makes a deep analysis of stability of linear systems with time-varying delays in [18]. Fog computing is introduced to minimize the delay for IoT applications in [19–20]. Moreover, in [21], a packet-based control law is proposed in networked control systems that explicitly compensates losses of the delay, data packet disorder, and data packet dropout based on Markov chain. Currently, the joint optimization design in wireless sensor networks has begun to attract more and more attention. A two-step algorithm is proposed in [22] when the real-time control and resource management are collaboratively considered.

In addition, the power consumption has gradually become another challenging problem [23–25]. In [26], two distributed local algorithms dynamically adjust the transmission power level per node and are proposed to take advantages in improving energy efficiency. The medium access control protocol in [27] is provided for efficiently reducing energy consumption with a two-radio architecture. In [28], an optimization method of controller and communication systems is proposed to minimize the power consumption for wireless networked control systems considering the imperfections of time delay and packet error. Joint resource allocation and power control are investigated to maximize the energy efficiency of device-to-device (D2D) communications in [29]. In [30], a power-based vehicle longitudinal control optimal algorithm is proposed to minimize energy consumption of the connected eco-driving system. In [31], the latency optimization for resource allocation is proposed in mobile edge computation offloading. Furthermore, the power consumption is considered a key indicator in many actual applications. Currently, in multihop wireless sensor network, a distributed power control and data scheduling algorithm based on a differential game framework is proposed in [32] to achieve effective use of the available harvested energy and balance the buffers of all sensor nodes.

Unfortunately, most of the existing works focus on either control strategy design or power consumption management in the WSA. The overviews in [33–34] and our previous work [35] reveal potential benefits of jointly optimizing control strategy and power consumption. However, few studies have taken into account these two aspects simultaneously. In this paper, a novel joint optimization algorithm is proposed, which meets the requirements of the real-time control and power consumption reduction. The main contributions of this paper can be summarized as follows:

- (i) In discrete-time domain, based on the control dynamics modeling and power consumption analysis, the optimization problem jointly considering the control stability and power consumption efficiency is formulated in the presence of the time delay
- (ii) A novel three-step joint optimization algorithm is proposed. First, the set of candidate paths is obtained by using the minimum hop routing algorithm. Next, the optimal control strategy for each candidate path can be iteratively derived with a backward recursion method. Finally, the joint optimization design is real-

ized through the best transmission path selection under the determined optimal control strategy

- (iii) In particular, the minimum hop routing algorithm is obtained based on the strong correlation between the utility function of the WSA and the network-induced time delay, which is totally determined by the number of hops in the transmission path. In addition, the optimal control signal can be derived as a linear function of the current state information and previous control signals

The rest of this article is organized as follows. Section 2 gives the system model and problem description. Then, in Section 3, the joint optimal control design under the influence of time delay is derived. The simulation experiment and conclusion are presented in Section 4 and Section 5, respectively.

## 2. System Model and Problem Formulation

In this section, a typical WSA model consists of the plant, sensor, actuator, controller, and spatially distributed network nodes as shown in Figure 1. The sensors adjacent to the controlled plant sample the state information at periodic intervals. The controller acts as a decision maker to arrange an optimal transmission path and generate control strategies to realize the closed-loop feedback control. Then, the control signals are sent along this transmission path to the actuators to ensure the desirable dynamic and steady-state response. However, the time delay induced by the shared wireless communication among the WSA components has serious effects on the stability of real-time control application [35]. Considering time delay, the joint optimization design of WSA is investigated in this paper to achieve both plant control stability and energy efficiency of the wireless sensor network.

*2.1. Power Consumption Analysis.* In general, there are several alternative transmission paths available in wireless sensor networks because of the inherent nature of distributed network structure. A simple dual-path case is shown in Figure 1 that the solid transmission path (1 → 2 → controller → 7 → 8) and the dotted transmission path (1 → 3 → 4 → controller → 5 → 6 → 8) are two candidate paths. In particular, a given transmission path is depicted in Figure 2, where there are  $m_k$  network nodes with the controller located at the  $m_k^c$ th node.

Considering a given transmission path  $k$ , the transmission power consumption from the  $j$ th network node to the  $(j + 1)$ th network node can be given by [36]

$$P_{j,j+1}^k = \mu \|x\|^2 + \kappa_d \|x\|^2 d_{j,j+1}^r, \quad (1)$$

where  $\|x\|$  denotes the amplitude of the signal  $x$ ,  $r$  is the signal attenuation factor,  $d_{j,j+1}$  represents the transmission distance, and  $\mu$  and  $\kappa_d$  are system determined constants.

Then, in the  $i$ th sampling interval ( $iT, (i + 1)T$ ), the sensor-to-controller and the controller-to-actuator power



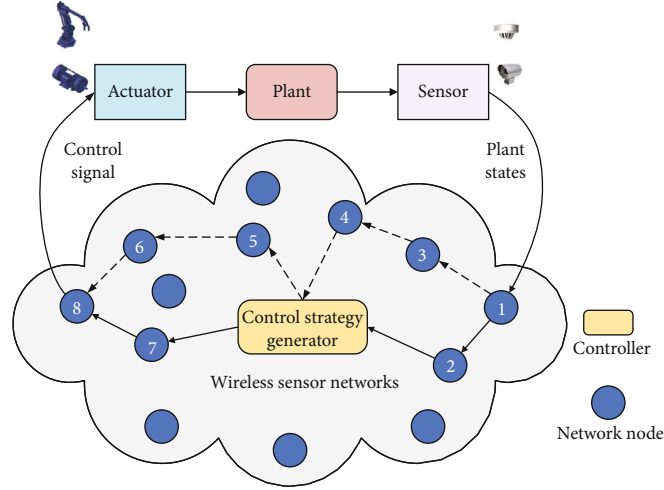
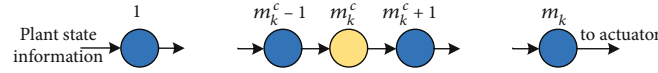


FIGURE 1: Structure of WSANs.


 FIGURE 2: The  $k$ th transmission path.

consumptions can be, respectively, expressed as [22].

$$P_{sc,i}^k = \sum_{j=1}^{m_k^c-1} \left( \mu \|s_i\|^2 + \kappa_d \|s_i\|^2 d_{j,j+1}^r \right), \quad (2)$$

$$P_{ca,i}^k = \sum_{j=m_k^c}^{m_k} \left( \mu \|u_{k,i}\|^2 + \kappa_d \|u_{k,i}\|^2 d_{j,j+1}^r \right), \quad (3)$$

where  $s_i$  is the sampled plant state and  $u_{k,i}$  is the control signal generated based on the received  $s_i$ .

Thus, the total transmission power consumption of the entire control process is

$$P_{\text{net}}^k = \sum_{i=0}^{J-1} \left( P_{sc,i}^k + P_{ca,i}^k \right), \quad (4)$$

where  $J$  is the number of sampling intervals in the control process.

**2.2. Control System Modeling.** In WSANs, the system dynamics in  $i$ th sampling interval can be formulated as [16]

$$s_{i+1} = V_i s_i + W_{i1} u_{k,i} + W_{i2} u_{k,i-1}, \quad (5)$$

where

$$\begin{aligned} V_i &= e^{V^T}, \\ W_{i1} &= \int_0^{T-\tau} e^{V^t} dt W, \\ W_{i2} &= \int_{T-\tau}^T e^{V^t} dt W, \end{aligned} \quad (6)$$

and here  $V$  and  $W$  are determined system parameters and  $\tau$  is the network-induced time delay, which is influenced by many factors such as the sensor distribution, node size, network topology, and even signal transmission, processing, and reception.

In order to ensure the WSAN stability, the objective of the control strategy design is to minimize the system cost function as [37]

$$P_{\text{cont}}^k = s_J^T B_J s_J + \sum_{i=0}^{J-1} \left( s_i^T B_0 s_i + u_{k,i}^T C_0 u_{k,i} \right), \quad (7)$$

where  $B_J$ ,  $B_0$ , and  $C_0$  are determined system matrices.

**2.3. Optimization Problem Formulation.** Considering both power consumption efficiency and control stability of WSANs, the utility function of the joint optimization problem can be expressed as

$$P_J^k = P_{\text{cont}}^k + \beta P_{\text{net}}^k, \quad (8)$$

where  $\beta$  is a weight coefficient.

Then, the utility function can be rewritten as

$$P_J^k = s_J^T B_J s_J + \sum_{i=0}^{J-1} (s_i^T B s_i + u_{k,i}^T C u_{k,i}), \quad (9)$$

where

$$B = B_0 + \beta \left[ \mu(m_k^c - 1) + \kappa_d \sum_{j=1}^{m_k^c - 1} d_{j,j+1}^r \right] I_B, \quad (10)$$

$$C = C_0 + \beta \left[ \mu(m_k - m_k^c + 1) + \kappa_d \sum_{j=m_k^c}^{m_k} d_{j,j+1}^r \right] I_C,$$

and  $I_i$  is an identity matrix with the same size as  $i$ .

Therefore, the objective of the joint optimization problem is to minimize the utility function through the transmission path routing and control strategy design, which is formulated as

$$\min_{\{u_{k,i}, k\}} P_J^k = s_J^T B_J s_J + \sum_{i=0}^{J-1} (s_i^T B s_i + u_{k,i}^T C u_{k,i}), \quad (11)$$

$$s.t. s_{i+1} = V_i s_i + W_{i1} u_{k,i} + W_{i2} u_{k,i-1}.$$

### 3. Joint Optimization Algorithm Design

In this section, a novel three-step algorithm is proposed to solve the joint optimization problem in (9). First, the candidate set of transmission paths is obtained by using the minimum hop routing algorithm. Then, for a given candidate transmission path, the control strategy can be derived in a backward recursion manner. Finally, the best transmission path selection under the optimal control strategy is determined.

In fact, it is difficult to directly solve the joint optimization problem (9). According to the principle of decoupling, the joint optimization problem can be decomposed into two subproblems: (S1) When the control strategy is designed, the joint optimization problem can be simplified to be an optimal path routing problem. (S2) When the transmission path is selected, the joint optimization problem can be converted to an individual control design problem. That is,

$$\text{S1 :}$$

$$\min_{\{u_{k,i}, k\}} P_J^k = s_J^T B_J s_J + \sum_{i=0}^{J-1} (s_i^T B s_i + (u_{k,i}^*)^T C u_{k,i}^*), \quad (12)$$

$$s.t. s_{i+1} = V_i s_i + W_{i1} u_{k,i}^* + W_{i2} u_{k,i-1}^*,$$

$$\text{S2 :}$$

$$\min_{\{u_{k,i}, k\}} P_J^{k^*} = s_J^T B_J s_J + \sum_{i=0}^{J-1} (s_i^T B s_i + u_{k,i}^{*T} C u_{k,i}^*), \quad (13)$$

$$s.t. s_{i+1} = V_i s_i + W_{i1} u_{k,i}^* + W_{i2} u_{k,i-1}^*,$$

where  $u_{k,i}^*$  and  $k^*$  denote the optimal control strategy and the optimal transmission path routing, respectively.

In general, the transmission path routing and control strategy design can be addressed base on subproblems S1 and S2, respectively, and then iteratively converge to the joint optimal design. However, the iteration process usually has uncertain convergence and large computational complexity. Below, the further analysis is presented to simplify the iteration process.

**3.1. Optimization Problem Transformation.** In order to solve S1 in (11), a typical approach is the exhaustive search method to derive the best transmission path. However, it requires lots of computations, especially in a large-scale network. Therefore, a set of candidate paths needs to be determined first to reduce the computation burden.

**Theorem 1.** *The subproblem S1 can be converted to be the transmission path selection problem with minimum hop count.*

*Proof.* Based on the assumption in subproblem S1, the optimization problem can be simplified as the following optimal transmission path selection problem

$$\min_{\{u_{k,i}^*, k\}} P_J^k(u_{k,i}^*) = s_J^T B_J s_J + \sum_{i=0}^{J-1} (s_i^T B s_i + (u_{k,i}^*)^T C u_{k,i}^*). \quad (14)$$

Then, the set of candidate transmission paths  $\eta$  subject to the minimum utility function can be obtained as

$$\eta = \arg \min_{\{k\}} P_J^k(u_{k,i}^*). \quad (15)$$

In the WSAAN, wireless communication may introduce time delays resulting in the system instability. Existing works [38–40] demonstrate the strong correlation between the delay and the utility function that a larger delay leads to an increase in the utility function, and vice versa. Thus, the minimum utility function can be transformed into the minimum delay problem as follows:

$$\eta = \arg \min_{\{k\}} \{\tau_k\}, \quad (16)$$

where  $\tau_k$  is the time delay of  $k$ th transmission path.

Theoretically, the time delay mainly includes transmission and access delays, which is typically proportional to the hop count [39, 41]. Therefore, the optimization problem in (15) can be equivalent to the minimum hop problem as

$$\eta = \arg \min_{\{k\}} \text{HC}_k, \quad (17)$$

where  $\text{HC}_k$  denotes the number of hops of a given transmission path.  $\square$

**3.2. Minimum Hop Routing Algorithm.** In this subsection, the minimum hop routing algorithm is provided to efficiently reduce the computational complexity. First, the

dynamic programming approach, as an optimization method of multistage decision-making, is used to transform the nonstandardized network into a standardized one to provide a coherent and regular architecture. Then, the set of candidate transmission paths is derived based on the minimum hop routing algorithm.

In general, as shown in Figure 3, the sensor network without obvious decision-making stages is called the nonstandardized network. It is difficult to find the minimum hop routing directly with the increase in the number of sensor nodes. In order to search the path quickly, it is necessary to add virtual nodes to convert the network into a standardized sensor network. It can be seen from the Figure 3 that the number of hops from node A to node E can be 2 hops ( $A \rightarrow C \rightarrow E$ ) or 3 hops ( $A \rightarrow B \rightarrow C \rightarrow E$ ), so a virtual child node E1 equivalent to node E can be added in the second decision-making stage as shown in Figure 4.

Based on the step above, the nonstandardized network problem can be successfully addressed. It is assumed that there are  $i$  sensor nodes during the  $l$ th decision-making stage. Then, we can model the minimum hop count function in the  $l$ th decision-making stage as follows:

$$\min HC_{(l,i)} = \min HC_{(l-1,i)} + \min Hop_{(l,i)}, \quad (18)$$

where

$$\text{Hop}_{(l,i)} = \begin{cases} 0, & \text{virtuallink,} \\ 1, & \text{otherwise.} \end{cases} \quad (19)$$

(1)  $l = 0$ : when  $l = 0$ , the minimum hop count function in the source node is

$$\min HC_{(0,i)} = 0 \quad (20)$$

(2)  $l = 1$ : identify and store the adjacent nodes with the smallest hop count in terms of (18), and the minimum hop count function when  $l = 1$  has

$$\min HC_{(1,i)} = \min Hop_{(1,i)} \quad (21)$$

(3)  $l = 2, \dots, n - 1, n$ : here,  $n$  is the total decision stages. We can obtain the set of candidate transmission paths with the minimum hop count as in (17). Finally, the set of candidate transmission paths can be derived as

$$\eta = \arg \min_{\{k\}} HC_{(n,i)}, \quad (22)$$

where  $\{k\}$  is the set of available transmission paths in the standardized sensor network.

By using the proposed minimum hop routing algorithm summarized as in Algorithm 1, the set of candidate transmission paths to the optimization problem in (16) can be obtained. For example, as in Figure 4, there are three candidate transmission paths, namely,  $A \rightarrow C \rightarrow D \rightarrow F$ ,  $A \rightarrow B \rightarrow D \rightarrow F$ , and  $A \rightarrow C \rightarrow E \rightarrow F$ .

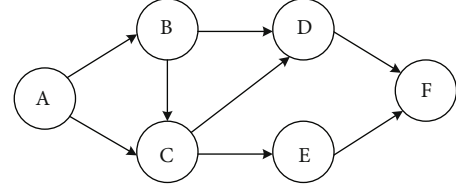


FIGURE 3: Nonstandardized sensor networks.

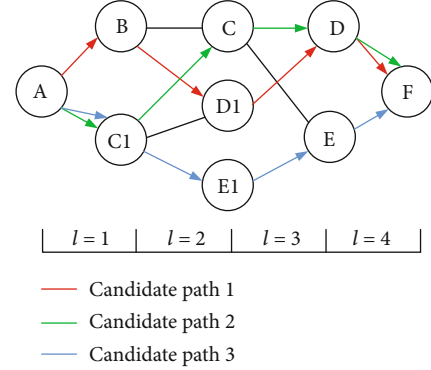


FIGURE 4: Standardized sensor networks.

**1 Step 1:**  
**2 Input:** Source node and destination node.  
**3 Step 2:**  
**4 Initialize**  $l = 0$ ,  $\min HC_{(0,i)} = 0$ .  
**5 for**  $l = 0 : 1 : n$  **do**  
**6 Calculate**  
 $\min HC_{(l,i)} = \min HC_{(l-1,i)} + \min Hop_{(l,i)}$ .  
**7 end**  
**8 Step 3:**  
**9 Output:**  $\eta = \arg \min_{\{k\}} HC_{(n,i)}$ .

ALGORITHM 1: Minimum hop routing algorithm.

3.3. *Optimal Control Design.* For a given transmission path  $\hat{k} \in \eta$ , the optimization problem in (13) can be equivalent to the following problem:

$$\begin{aligned} \min_{\{u_{\hat{k},i}\}} P_J^{\hat{k}} &= s_J^T B_J s_J + \sum_{i=0}^{J-1} \left( s_i^T B s_i + u_{\hat{k},i}^T C u_{\hat{k},i} \right), \\ \text{s.t. } s_{i+1} &= V_i s_i + W_{i1} u_{\hat{k},i} + W_{i2} u_{\hat{k},i-1}. \end{aligned} \quad (23)$$

Define a new state vector  $f_{\hat{k},i} = [s_i^T, u_{k \wedge, i-1}]^T$ , and then, the optimization problem in (22) can be rewritten as

$$\begin{aligned} \min_{\{u_{\hat{k},i}\}} P_J^{\hat{k}} &= f_{\hat{k},J}^T \bar{B}_J f_{\hat{k},J} + \sum_{i=0}^{J-1} \begin{bmatrix} f_{k \wedge, i} \\ u_{k \wedge, i} \end{bmatrix}^T \begin{bmatrix} \bar{B} & 0 \\ 0 & C \end{bmatrix} \begin{bmatrix} f_{\hat{k},i} \\ u_{\hat{k},i} \end{bmatrix}, \\ \text{s.t. } f_{\hat{k},i+1} &= E_i f_{\hat{k},i} + F_i u_{\hat{k},i}, \end{aligned} \quad (24)$$

**1 Step 1:**  
 2 Use minimum hop routing algorithm.  
 3 Derive the set of candidate transmission paths as in (16) that  $\eta = \arg \min_{\{k\}} HC_k$ .

**4 Step 2:**  
 5 For a given candidate transmission path  $\hat{k}$   
 6 Initialize  $Q_J = \bar{B}_J$ .  
 7 **for**  $i = J - 1 : -1 : 0$  **do**  
 8     Calculate  $H_{\hat{k},i}$  by using (26) that  

$$H_{\hat{k},i} = [F_i^T Q_{i+1} F_i + C]^{-1} F_i^T Q_{i+1} E_i.$$
  
 9     Calculate  $Q_i$  by using (26) that  

$$Q_i = E_i^T Q_{i+1} E_i + \bar{B} - H_{\hat{k},i}^T F_i^T Q_{i+1} E_i.$$
  
 10 **end**  
 11 Initialize  $s(0), u_{\hat{k},i} = 0, i \leq 0$ .  
 12 **for**  $i = 0 : 1 : J - 1$  **do**  
 13     Obtain  $f_{\hat{k},i} = [s_i^T, u_{k \wedge i-1}]^T$ .  
 14     Calculate  $u_{\hat{k},i}^*$  by using (24) that  

$$u_{\hat{k},i}^* = -H_{\hat{k},i} f_{\hat{k},i}.$$
  
 15 **end**  
 16 Step 3  
 17 **for**  $\hat{k} \in \eta$   
 18     Calculate  $P_J^{\hat{k}}(u_{\hat{k},i}^*) =$   

$$f_{\hat{k},J}^T \bar{B}_J f_{\hat{k},J} + \sum_{i=0}^{J-1} \{f_{\hat{k},i}^T \bar{B} f_{\hat{k},i} + (u_{k \wedge i}^*)^T C u_{\hat{k},i}^*\}.$$
  
 19 **end**  
 20 Derive the best transmission path  $k^*$  by using (28)  

$$k^* = \arg \min_{\{\hat{k}\}} \{P_J^{\hat{k}}(u_{\hat{k},i}^*)\}.$$
  
 21 The corresponding optimal control design is  

$$u_{k^*,i}^* = -H_{k^*,i} f_{k^*,i}.$$

ALGORITHM 2: Three-step joint optimization algorithm.

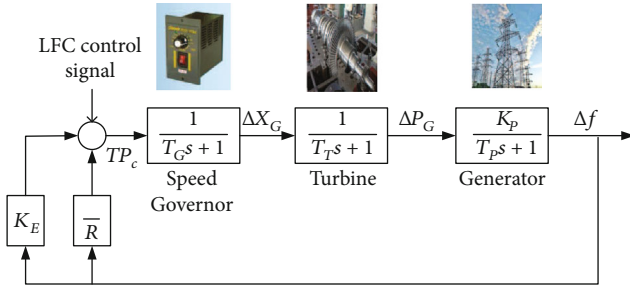
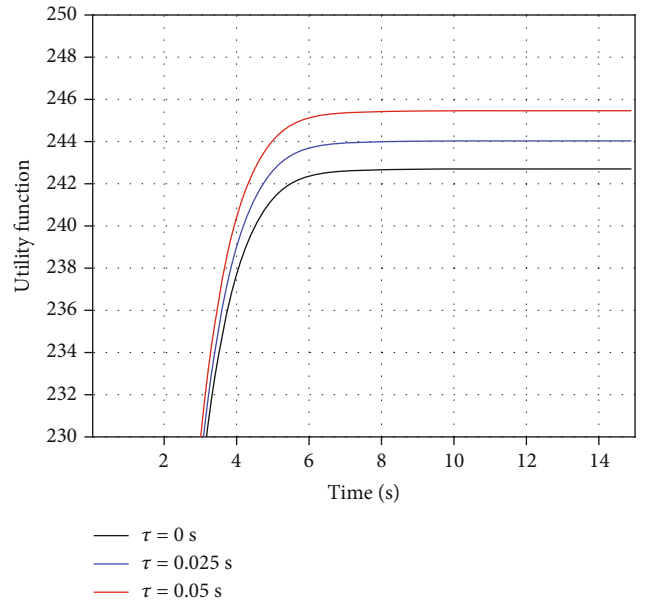


FIGURE 5: Block diagram of an LFC system for power grid.

where

$$\bar{B}_J = \begin{bmatrix} B_J & 0 \\ 0 & 0 \end{bmatrix}, \bar{B} = \begin{bmatrix} B & 0 \\ 0 & 0 \end{bmatrix}, \quad (25)$$

$$E_i = \begin{bmatrix} V_i & W_{i2} \\ 0 & 0 \end{bmatrix}, F_i = \begin{bmatrix} W_{i1} \\ 1 \end{bmatrix}.$$

FIGURE 6: Performance comparison of different time delays with the sampling period  $T = 0.1$  s.

**Theorem 2.** The optimal control strategy to the optimization

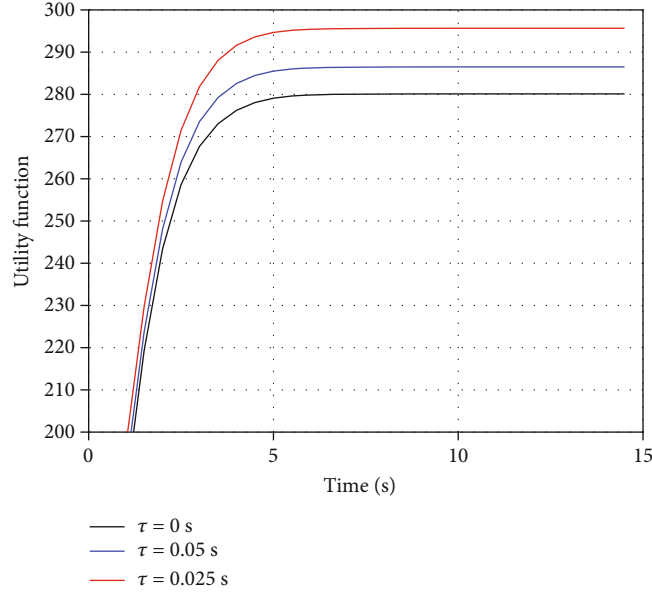


FIGURE 7: Performance comparison of different time delays with the sampling period  $T = 0.5$  s.

problem (23) is given by

$$u_{\hat{k},i}^* = -H_{\hat{k},i}^T f_{\hat{k},i}, \quad i = 1, 2, \dots, J-1, \quad (26)$$

where

$$\begin{aligned} H_{\hat{k},i} &= [F_i^T Q_{i+1} F_i + C]^{-1} F_i^T Q_{i+1} E_i, \\ Q_i &= E_i^T Q_{i+1} E_i + \bar{B} - H_{\hat{k},i}^T F_i^T Q_{i+1} E_i, \\ Q_J &= \bar{B}_J. \end{aligned} \quad (27)$$

The proof can be achieved similar to the derivation process of optimal control strategy in [37].

**3.4. Joint Optimal Path Determination.** Once the set of candidate transmission paths and the corresponding optimal control design are determined, the utility function can be expressed as

$$P_J^{\hat{k}}(u_{\hat{k},i}^*) = f_{\hat{k},J}^T \bar{B}_J f_{\hat{k},J} + \sum_{i=0}^{J-1} \left\{ f_{\hat{k},i}^T \bar{B} f_{\hat{k},i} + (u_{\hat{k},i}^*)^T C u_{\hat{k},i}^* \right\}. \quad (28)$$

Then, the best transmission path is determined by

$$k^* = \arg \min_{\{\hat{k}\}} \left\{ P_J^{\hat{k}}(u_{\hat{k},i}^*) \right\}, \quad (29)$$

and the corresponding optimal control design is given by (24) that

$$u_{k^*,i}^* = -H_{k^*,i}^T f_{k^*,i}. \quad (30)$$

Therefore, in order to meet the requirements of real-time

control and efficient power consumption, the three-step joint optimization algorithm can be summarized as in Algorithm 2. First, the set of candidate transmission paths is obtained by using the minimum hop routing algorithm, and then, the optimal control design for each candidate transmission path can be derived in a backward recursion manner. Finally, the best transmission path and the corresponding control design are determined by the minimum utility function.

## 4. Simulation Results

In this section, a case study of the load frequency control (LFC) system in power grid [37] is used to verify the performance of the provided joint optimal design for WSANs. In the simulation, 15 sensor nodes are considered in the shared wireless network, and the controller is placed in a determined location. The distance between sensor nodes is uniform in [1m, 5m].

The typical LFC system consists of generator, turbine, speed governor, and LFC controllers as shown in Figure 5. The objective is to design the control signal, namely, the speed, to maintain the frequency deviation  $\Delta f$  within the specified range. The plant state is  $s(t) = [\Delta P_c \quad \Delta f \quad \Delta P_G \quad \Delta X_G]^T$ , and  $\Delta X_G$ ,  $\Delta P_G$ , and  $\Delta P_c$  represent the valve position, the derivation of generator mechanical output, and the generator output, respectively [37]. The system parameters are given by

$$\begin{aligned} W &= [0 \quad 1 \quad 0 \quad 0], \\ V &= \begin{bmatrix} 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0.95 & -1.2 \end{bmatrix}. \end{aligned} \quad (31)$$



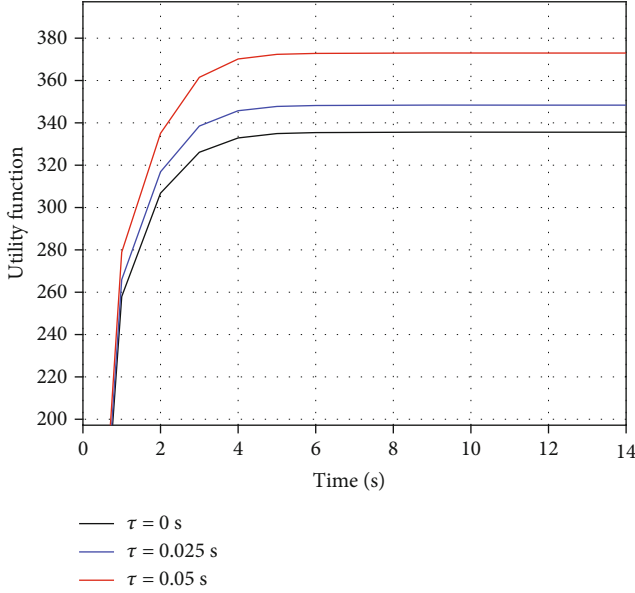


FIGURE 8: Performance comparison of different time delays with the sampling period  $T = 1.0$  s.

In the simulation, we set the weight coefficient  $\beta = 1$ , and

$$B_j = B_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, C_0 = 1. \quad (32)$$

First, the utility functions of the proposed joint optimization algorithm in the presence of different time delays are shown in Figures 6–8. The sampling periods are set as  $T = 0.1$  s,  $0.5$  s,  $1.0$  s with different time delays  $\tau = 0, 0.25T, 0.5T$  in different scenarios. It can be seen from figures that the utility function can gradually converge to a constant value, which indicates that the proposed joint optimization algorithm is efficient and stable under various time delays. Furthermore, the comparison results show that the time delay will influence the utility function. That is, as the time delay increases, the utility function becomes larger, which means that more serious the system performance degradation is caused.

In order to further verify the effectiveness of the proposed joint optimization algorithm, the convergence utility function is shown in Figure 9 when the ratio of the time delay to the sampling interval is set from 0 to 0.9. It can be seen that the system stability can always be guaranteed. In addition, the lower utility function can be obtained either the sampling interval or the time delay becomes larger. This stems from the fact that increasing the sampling interval and time delay will delay the execution of the control signal and also slow down the acquisition frequency of the plant state, which makes it much more difficult for the plant to converge.

Finally, in Figures 10–12, we show performance comparison in three different cases: individual optimal transmission path routing, individual optimal control strategy design, and

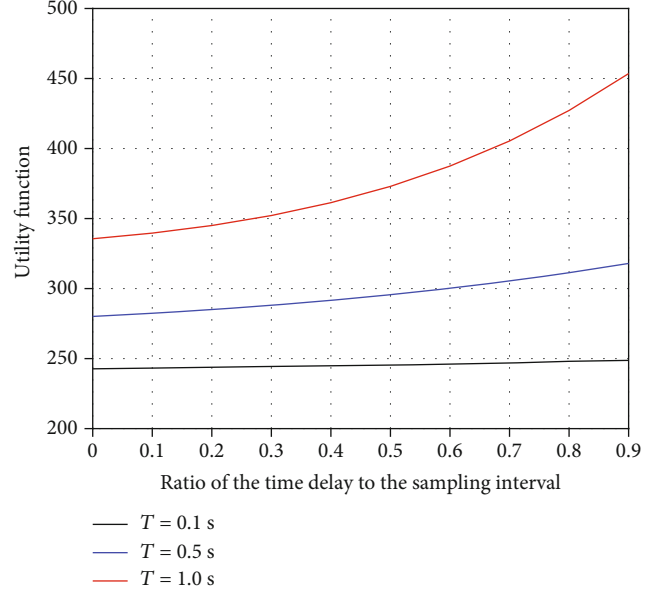


FIGURE 9: Performances of the proposed algorithm with different time delays.

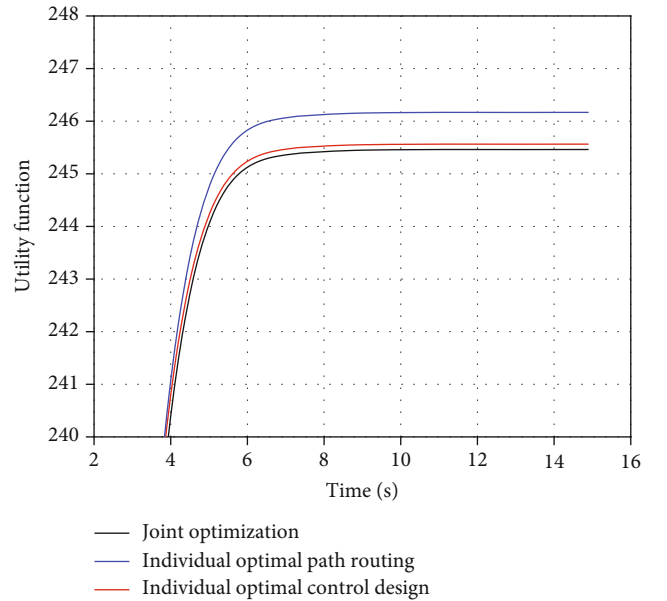


FIGURE 10: Utility function comparison of three algorithms with the sampling period  $T = 0.1$  s.

the proposed joint optimization. The sampling periods are set as  $T = 0.1$  s,  $0.5$  s,  $1.0$  s and the time delay is  $\tau = 0.5T$ . The results indicate that the proposed joint optimization design is superior to the independent design. The joint optimization scheme improves the stability of the control system and the power consumption efficiency. We can also observe that, when the delay is relatively small, the performance of the individual transmission path routing is very close to the performance of the joint optimization algorithm. While its utility function will significantly increase, even fail to converge, when the time delay becomes larger.

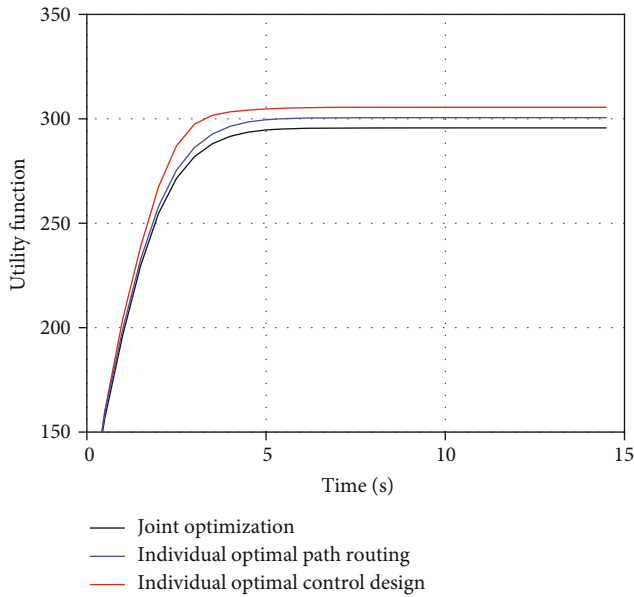


FIGURE 11: Utility function comparison of three algorithms with the sampling period  $T = 0.5$  s.

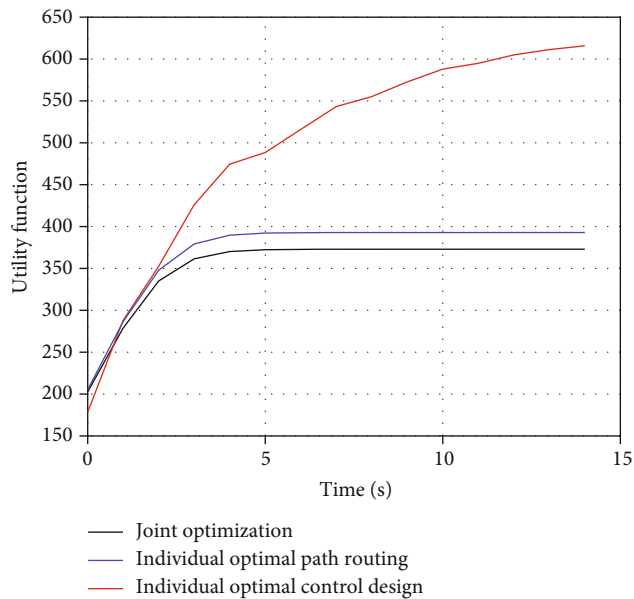


FIGURE 12: Utility function comparison of three algorithms with the sampling period  $T = 1.0$  s.

## 5. Conclusions

In this paper, an improved joint optimization scheme of the WSN system is proposed taking into account the network-induced time delays caused by wireless communication. The WSN is modeled as a linear system in discrete-time domain and the joint optimization problem is formulated as a quadratic utility function, which can be decomposed into two subproblems, and then, a three-step algorithm is designed in the closed-loop feedback control. Finally, a case study of the LFC in power grid system is investigated to

demonstrate the effectiveness of the proposed joint optimization algorithm that better control stability and power consumption efficiency are achieved.

## Data Availability

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

## Conflicts of Interest

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

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (62001011), the Beijing Wuzi Youth Foundation (2020XJQN08), the Beijing Natural Science Foundation (L202016), the Scientific Research Plan of Beijing Municipal Commission of Education (KM201910005026), and the Beijing Nova Program of Science and Technology (Z191100001119094).


## References

- [1] 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.
- [2] J. Xiong, J. Ren, L. Chen et al., "Enhancing privacy and availability for data clustering in intelligent electrical service of IoT," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1530–1540, 2019.
- [3] D. Wu, R. Bao, Z. Li, H. Wang, H. Zhang, and R. Wang, "Edge-cloud collaboration enabled video service enhancement: a hybrid human-artificial intelligence scheme," *IEEE Transactions on Multimedia*, vol. 23, pp. 2208–2221, 2021.
- [4] N. Primeau, R. Falcon, R. Abielmona, and E. M. Petriu, "A review of computational intelligence techniques in wireless sensor and actuator networks," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2822–2854, 2018.
- [5] L. Da Xu, W. He, and S. Li, "Internet of things in industries: a survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.
- [6] J. Xiong, M. Zhao, M. Z. A. Bhuiyan, L. Chen, and Y. Tian, "An AI-enabled three-party game framework for guaranteed data privacy in mobile edge crowdsensing of IOT," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, pp. 922–933, 2021.
- [7] X. Xue and J. Zhang, "Matching large-scale biomedical ontologies with central concept based partitioning algorithm and adaptive compact evolutionary algorithm," *Applied Soft Computing*, vol. 106, article 107343, 2021.
- [8] X. Xue, C. Yang, C. Jiang, P.-W. Tsai, G. Mao, and H. Zhu, "Optimizing ontology alignment through linkage learning on entity correspondences," *Complexity*, vol. 2021, Article ID 5574732, 12 pages, 2021.
- [9] Z. Wang, Y. Gao, C. Fang, L. Liu, H. Zhou, and H. Zhang, "Optimal control design for connected cruise control with

- stochastic communication delays,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 357–369, 2020.
- [10] D. Wu, X. Han, Z. Yang, and R. Wang, “Exploiting transfer learning for emotion recognition under cloud-edge-client collaborations,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 2, pp. 479–490, 2020.
- [11] F. Xia, “QoS challenges and opportunities in wireless sensor/actuator networks,” *Sensors*, vol. 8, no. 2, pp. 1099–1110, 2008.
- [12] J. Xiong, R. Ma, L. Chen et al., “A personalized privacy protection framework for mobile crowdsensing in IIoT,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4231–4241, 2020.
- [13] I. F. Akyildiz and I. H. Kasimoglu, “Wireless sensor and actor<sup>\*</sup> networks: research challenges,” *Ad Hoc Networks*, vol. 2, no. 4, pp. 351–367, 2004.
- [14] X. Xue, X. Wu, C. Jiang, G. Mao, and H. Zhu, “Integrating sensor ontologies with global and local alignment extractions,” *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6625184, 10 pages, 2021.
- [15] L. Zhang, H. Gao, and O. Kaynak, “Network-induced constraints in networked control systems—a survey,” *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 403–416, 2013.
- [16] J. Nilsson, B. Bernhardsson, and B. Wittenmark, “Stochastic analysis and control of real-time systems with random time delays,” *Automatica*, vol. 34, no. 1, pp. 57–64, 1998.
- [17] H. Shousong and Z. Qixin, “Stochastic optimal control and analysis of stability of networked control systems with long delay,” *Automatica*, vol. 39, no. 11, pp. 1877–1884, 2003.
- [18] X.-M. Zhang, Q.-L. Han, A. Seuret, F. Gouaisbaut, and Y. He, “Overview of recent advances in stability of linear systems with timevarying delays,” *IET Control Theory & Applications*, vol. 13, no. 1, pp. 1–16, 2019.
- [19] R. Deng, R. Lu, C. Lai, T. H. Luan, and H. Liang, “Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption,” *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1171–1181, 2016.
- [20] A. Yousefpour, G. Ishigaki, and J. P. Jue, “Fog computing: Towards minimizing delay in the internet of things,” in *2017 IEEE international conference on EDGE computing (EDGE)*, pp. 17–24, IEEE, Honolulu, HI, USA, 2017.
- [21] Y.-B. Zhao, G.-P. Liu, Y. Kang, and L. Yu, “Stochastic stabilization of packet-based networked control systems,” in *Packet-Based Control for Networked Control Systems*, pp. 77–85, Springer, 2018.
- [22] Z. Wang, Y. Guo, Y. Gao, C. Fang, M. Li, and E. Sun, “Joint optimization of control law and power consumption for wireless sensor and actuator networks,” in *2019 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, IEEE, Waikoloa, HI, USA, 2019.
- [23] J. Aponte-Luis, J. Gómez-Galán, F. Gómez-Bravo, M. Sánchez-Raya, J. Alcina-Espigado, and P. Teixido-Rovira, “An efficient wireless sensor network for industrial monitoring and control,” *Sensors*, vol. 18, no. 2, p. 182, 2018.
- [24] C. S. Abella, S. Bonina, A. Cucuccio et al., “Autonomous energy-efficient wireless sensor network platform for home/office automation,” *IEEE Sensors Journal*, vol. 19, no. 9, pp. 3501–3512, 2019.
- [25] Z. Li, Y. Jiang, Y. Gao, L. Sang, and D. Yang, “On buffer-constrained throughput of a wireless-powered communication system,” *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 2, pp. 283–297, 2019.
- [26] M. Kubisch, H. Karl, A. Wolisz, L. C. Zhong, and J. Rabaey, “Distributed algorithms for transmission power control in wireless sensor networks,” in *2003 IEEE Wireless Communications and Networking, 2003. WCNC 2003*, pp. 558–563, New Orleans, LA, USA, 2003.
- [27] M. J. Miller and N. H. Vaidya, “A mac protocol to reduce sensor network energy consumption using a wakeup radio,” *IEEE Transactions on Mobile Computing*, vol. 4, no. 3, pp. 228–242, 2005.
- [28] Y. Sadi, S. C. Ergen, and P. Park, “Minimum energy data transmission for wireless networked control systems,” *IEEE Transactions on Wireless Communications*, vol. 13, no. 4, pp. 2163–2175, 2014.
- [29] Y. Jiang, Q. Liu, F. Zheng, X. Gao, and X. You, “Energy-efficient joint resource allocation and power control for D2D communications,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, pp. 6119–6127, 2016.
- [30] Q. Jin, G. Wu, K. Boriboonsomsin, and M. J. Barth, “Power-based optimal longitudinal control for a connected eco-driving system,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2900–2910, 2016.
- [31] J. Ren, G. Yu, Y. Cai, and Y. He, “Latency optimization for resource allocation in mobile-edge computation offloading,” *IEEE Transactions on Wireless Communications*, vol. 17, no. 8, pp. 5506–5519, 2018.
- [32] H. Al-Tous and I. Barhumi, “Differential game for resource allocation in energy harvesting wireless sensor networks,” *IEEE Transactions on Green Communications and Networking*, vol. 4, no. 4, pp. 1165–1173, 2020.
- [33] R. A. Gupta and M.-Y. Chow, “Networked control system: overview and research trends,” *IEEE Transactions on Industrial Electronics*, vol. 57, no. 7, pp. 2527–2535, 2009.
- [34] X. Ge, F. Yang, and Q.-L. Han, “Distributed networked control systems: a brief overview,” *Information Sciences*, vol. 380, pp. 117–131, 2017.
- [35] L. Mo, X. Cao, Y. Song, and A. Kritikakou, “Distributed node coordination for real-time energy-constrained control in wireless sensor and actuator networks,” *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 4151–4163, 2018.
- [36] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [37] Z. Wang, X. Wang, L. Liu, and M. Huang, “Optimal state feedback control for wireless networked control systems with decentralised controllers,” *IET Control Theory & Applications*, vol. 9, no. 6, pp. 852–862, 2015.
- [38] R. Sriram, G. Manimaran, and C. S. R. Murthy, “Preferred link based delay-constrained least-cost routing in wide area networks,” *Computer Communications*, vol. 21, no. 18, pp. 1655–1669, 1998.
- [39] S. Ping, “Delay measurement time synchronization for wireless sensor networks,” *Intel Research Berkeley Lab*, vol. 6, pp. 1–10, 2003.
- [40] S. Lindsey, C. Raghavendra, and K. M. Sivalingam, “Data gathering algorithms in sensor networks using energy metrics,” *IEEE Transactions on Parallel & Distributed Systems*, vol. 9, pp. 924–935, 2002.
- [41] M. Haenggi and D. Puccinelli, “Routing in ad hoc networks: a case for long hops,” *IEEE Communications Magazine*, vol. 43, no. 10, pp. 93–101, 2005.

## Research Article

# Enterprise Financial Risk Management Using Information Fusion Technology and Big Data Mining

Huabo Yue,<sup>1</sup> Haojie Liao ,<sup>1,2,3</sup> Dong Li,<sup>2</sup> and Ling Chen<sup>3</sup>

<sup>1</sup>College of Graduate Studies, Master of Management Program in Management (International Program) WALAILAK University, 222 Thaiburi, Thasala, Nakhon Si Thammarat 80160, Thailand

<sup>2</sup>Accounting and Audit School, Guangxi University of Finance and Economics, Nanning, Guangxi 530003, China

<sup>3</sup>College of Graduate Studies, Bansomdejchaopraya Rajabhat University, Bangkok10600, Thailand

Correspondence should be addressed to Haojie Liao; 2017110007@gxufe.edu.cn

Received 29 September 2021; Revised 2 November 2021; Accepted 5 November 2021; Published 15 December 2021

Academic Editor: Chin-Ling Chen

Copyright © 2021 Huabo Yue 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.

This paper aims to study enterprise Financial Risk Management (FRM) through Big Data Mining (BDM) and explore effective FRM solutions by introducing information fusion technology. Specifically, big data technology, Support Vector Machine (SVM), Logistic regression, and information fusion approaches are employed to study the enterprise financial risks in depth. Among them, the selection of financial risk indexes has a great impact on the monitoring results of the SVM-based FRM model; the Logistic regression-based FRM model can efficiently classify financial risks; the information fusion-based FRM model uses a fusion algorithm to fuse different information sources. The results show that the SVM-based and Logistic regression-based FRM models can manage and classify enterprise financial risks effectively in practice, with a classification accuracy of 90.22% and 90.88%, respectively; by comparison, the information fusion-based FRM model beats SVM-based and Logistic regression-based FRM models by presenting a classification accuracy as high as 95.18%. Therefore, it is concluded that the information fusion-based FRM is better than the SVM-based and Logistic regression-based models; it can integrate and calculate multiple enterprise financial risk data from different sources and obtain higher accuracy; besides, big data technology can provide important research methods for enterprise financial risk problems; SVM-based FRM model and Logistic regression-based FRM model can well classify enterprise financial risks, with relatively high accuracy.

## 1. Introduction

Today, the fast socio-economic development features a new technological revolution by Information Technology (IT), such as big data [1–3], cloud computing [4–6], and the Internet of Things (IoT), which is transforming people's life, work, and the society towards informatization and intellectualization; moreover, thanks to IT [7–9] and the e-commerce industry based on it, the connection in between people and enterprises is getting ever closer despite their geographic distances and cultural or political barriers. In particular, statistics of the development status of e-commerce enterprises reveals that financial situation and financial risks [10–12] can determine how far and high an enterprise can develop; thus, the research of these factors has great practical significance. The Big Data Mining

(BDM) approach might be just born to analyze enterprise financial data with its excellent identification effect, thereby being able to give early warning against enterprise financial risks [13–15].

To improve the level of enterprise Financial Risk Management (FRM) [16], scholars have conducted numerous studies and developed some effective theoretical methods. The financial risk of an entity can be defined as the entity's possibility of money-losing in financial activities. In terms of financial risk theory analysis, there are large numbers of practical cases, some foreign scholars believe that the causes behind the enterprise financial risks are diverse and need specified analysis. Valaskova et al. (2018) [17] once argued that financial risks could be solved by regression analysis. Some experts pointed out that the economic situation was a critical enterprise financial risk factor [18, 19]. Thereupon,



some scholars put forward risk minimization, risk transfer, risk tolerance, and risk treatment process to prevent the occurrence of financial risk problems. Zhang et al. (2021) [20] confirmed that big data and fuzzy Analytic Hierarchy Process (AHP) could efficiently and accurately monitor the financial risks. After studying the enterprise financial risks [21], some scholars believe that enterprise financial risks can be effectively reduced by combining financial risk management with the mathematical model. Domestic scholars also put forward some strategic views and research results, such as using the dynamic portfolio to prevent financial risks. The monitoring and analysis of enterprise financial risk [22–24] ultimately aim to effectively manage the risks. Chinese scholars have established and improved the enterprise internal risk prevention and management system [25–27].

To sum up, the current research on enterprise financial risk has not involved the combination of information fusion and big data technology to study enterprise financial risk. Although there have been attempts on the application of DM to enterprise financial management, they have shown great deficiencies in terms of security and classification accuracy, so there is still much room for improvement. Given these shortcomings, this paper optimizes their deficiencies. Thereupon, three models are introduced into Big Data Mining (BDM) to delve into enterprise financial risks: Support Vector Machine (SVM, SVMs)-based FRM model, Logistic regression-based FRM model, and the information fusion-based FRM model. The innovation of this paper is to combine BDM with the SVM model, Logistic regression model, and information fusion technology, separately to study the enterprise financial risks. Consequently, the combination method can well classify enterprise financial risks, with very high accuracy. The contents provide a theoretical basis for the follow-up research, which is of great significance. It is imperative for enterprises to improve their risk-bearing abilities and promote enterprise development. The technical route of this paper reads: 1. The proposal of the research direction, namely the enterprise FRM research; 2. Selection of research methods, namely BDM technology, SVM-based FRM model, Logistic regression-based FRM model, and information fusion-based FRM model; 3. Research results; 4. Conclusions analysis. Figure 1 shows the technical route.

## 2. Monitoring Models under Different Technologies

**2.1. SVM-Based FRM Model.** According to the current research on enterprise FRM, financial risk indexes have a great impact on the monitoring results of the SVM-based FRM model. Researchers prefer the business operational financial data as the input index of the FRM model, which shows that enterprise financial data are universal and widely desirable. Therefore, the following indexes are chosen for the proposed enterprise financial risk model: the financial structure of the enterprise, the ratio of retained earnings to total assets, the current debt to assets ratio, the ratio of asset management, and earnings management indexes [28–30]. Figure 2 shows the specific indexes of FRM.

The basic SVM model is a linear classifier with the largest interval in feature space, which can also be extended to a nonlinear classifier by the kernel function method. SVM can be used to classify two-dimensional (2D) patterns and find the 2D plane of decision-making in vector space compared with perceptron. Both models are classification models. SVM can optimize the hyperplane according to the interval maximization given correctly separable points; while perceptron strives to find a separation hyperplane that can completely and correctly separate the positive instance points from negative instance points in the training set. That is, under SVM multiple-pattern identification, the classifier must be built first.

SVM can well address both nonlinear classification and linear classification problems and deeply mine relevant data because SVM can segment the non-segmentable linear samples and transform them from low-dimensional space to high-dimensional space to obtain the best segmentation plane. If the training data set of the SVM algorithm is linear in the plane and can be segmented, then its decision function is calculated by Eq. (1):

$$y_i = \omega x + b \quad (1)$$

In Eq. (1),  $\omega$  means weight vector, and  $b$  is offset vector. If the training data set of the SVM algorithm is nonlinear in the plane and can also be segmented, then its decision function is shown in Eq. (2):

$$\min \left( \frac{1}{2} \omega^T \omega + C \sum_{i=1}^N \delta_i \right) \quad (2)$$

$$y_i(\omega x + b) \geq 1 - \delta_i, \delta_i \geq 0, i = 1, 2, \dots, N \quad (3)$$

Eq. (3) is a supplement to Eq. (2). In Eq. (2),  $C$  is a penalty factor, and the minimum classification error and maximum spacing of positive and negative categories depend on  $C$ ;  $\delta_i$  denotes a nonnegative relaxation variable. In the SVM-based enterprise FRM model, the enterprise financial data are used as the input vector, and the SVM algorithm model can well classify the enterprise financial risk and further monitor the risks.

**2.2. Big Data Technology.** Big data cover far beyond the traditional database in terms of data acquisition, data storage, data management, and data analysis. Thus, big data have shown many advantages, such as abundant data information, fast data conversion, rich data types, and low-value density in the data processing. Considering these factors, this paper integrates BDM into enterprise financial analysis. The specific process of data generation, storage, analysis, and application of enterprise structured big data will go through several complicated steps, and the interrelationship between these processes constitutes the big data structure. Generally, algorithm prediction or document consultation processes all involve big data collection, data storage, data processing, and specific application.

Specific enterprise FRM process is divided into three parts: financial risk identification, evaluation, and



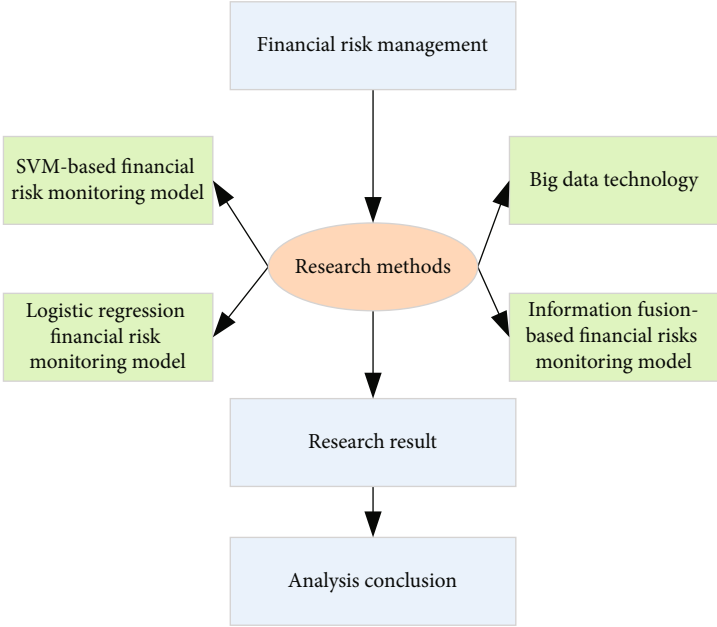


FIGURE 1: Technical route.

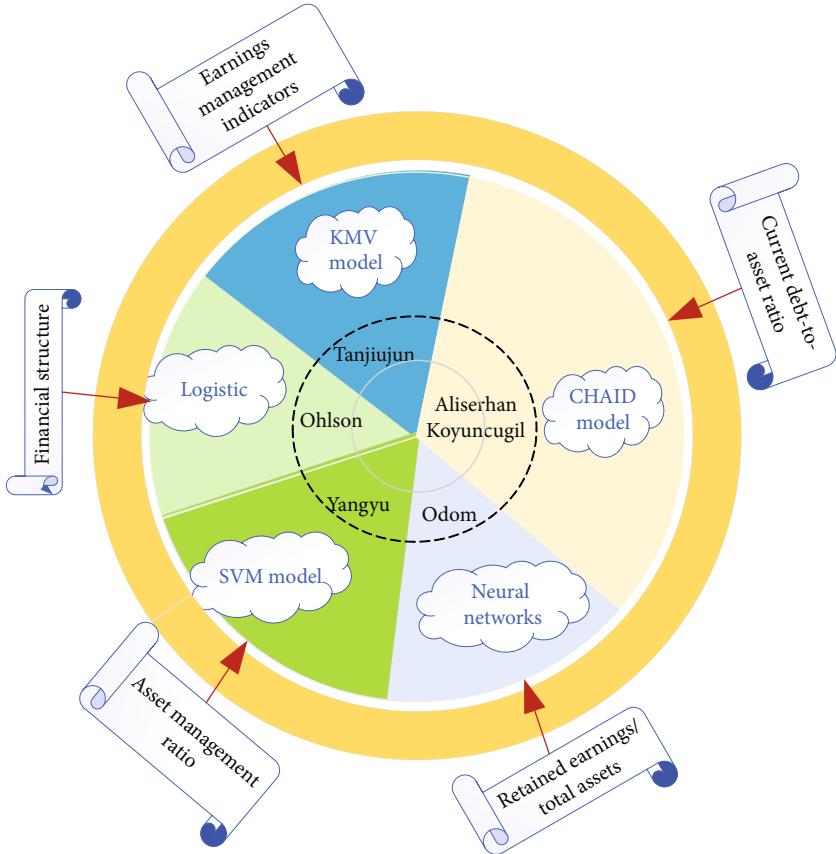


FIGURE 2: FRM indexes.

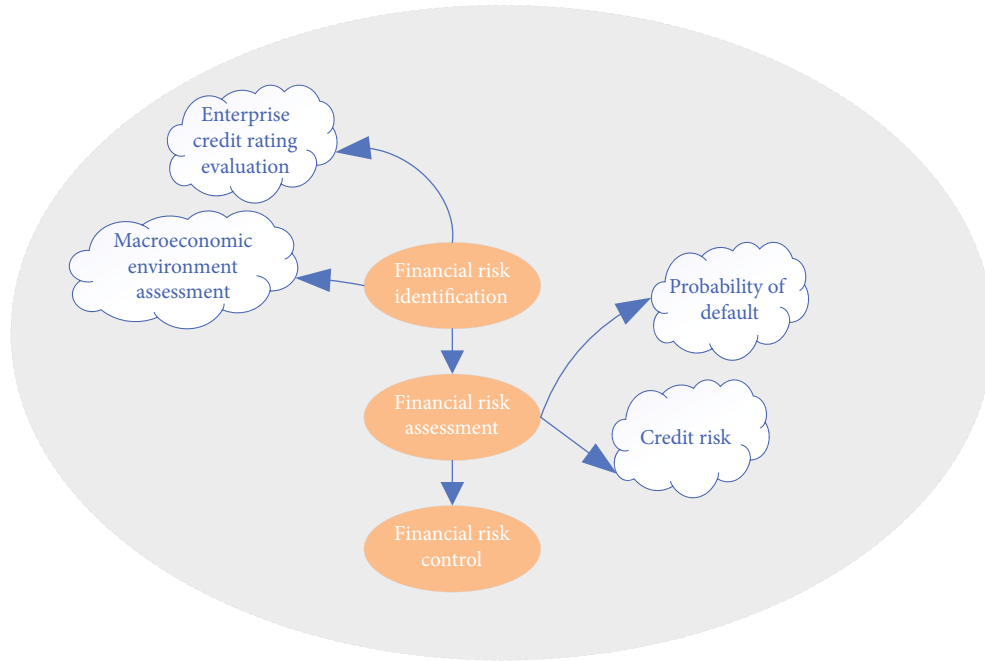


FIGURE 3: Flowchart of enterprise FRM.

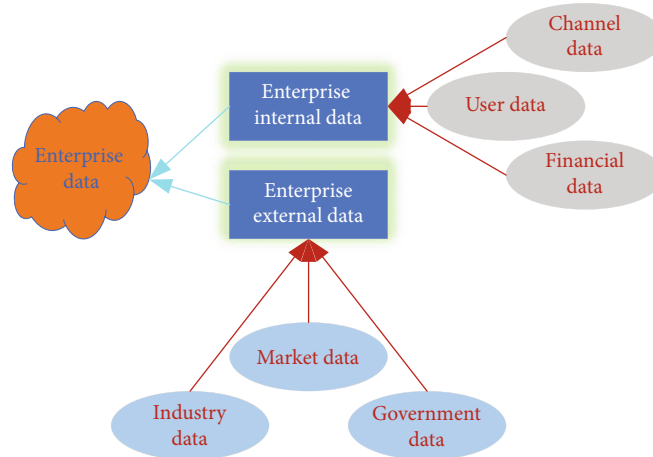


FIGURE 4: Enterprise data type framework.

management. Further, financial risk identification can be subdivided into the evaluation of the macroeconomic environment and the evaluation of enterprise credit rating; financial risk evaluation can be subdivided into default probability and credit risks. Figure 3 shows the enterpriseFRM process.

The selected enterprise operational general indexes are subdivided: first, the enterprise data can be divided into internal data and external data. Internal data contain channel data, financial data, and user data. External data include market data, government data, and industry data. Figure 4 shows the specific data type framework of the enterprise.

Flow data on the enterprise platform encompass the number of visitors, the length of visit, visitor profile, and visitor-pay conversion. The visitor profile can be subdivided into visitor age, visitor gender, and visitor category. Figure 5

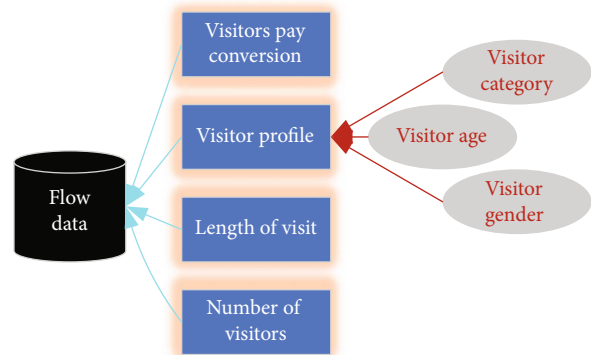


FIGURE 5: Flow data framework of an enterprise platform.

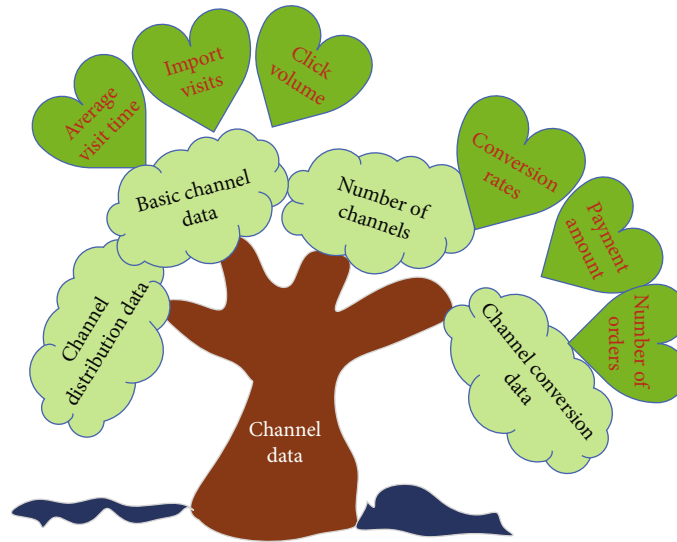


FIGURE 6: Dendrogram of enterprise channel data.

illustrates the flow data framework of the enterprise platform.

Enterprise channel data can be divided into channel distribution data, basic channel data, the number of channels, and channel conversion data. Basic channel data can be subdivided into average visit time, import visits, and click volume. Channel conversion data can be subdivided into conversion rates, the number of orders, and payment amount. Figure 6 presents the channel data tree of the enterprise.

On-platform enterprise data can be divided into four parts: logistics data, visitor transaction data, revenue data, and user data. Figure 7 shows the data framework of on-platform enterprise data.

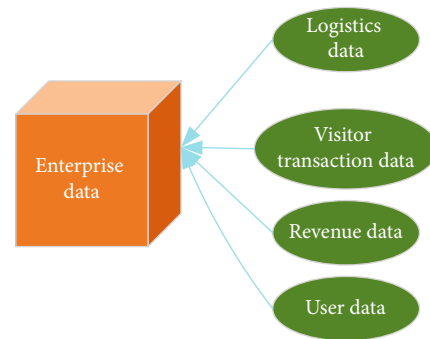


FIGURE 7: Data framework of on-platform enterprises.

On-platform business-customer transaction data can be divided into order data, calculated data, and payment data. This division can completely analyze the on-platform customer transaction data and implement it throughout the whole transaction process. Order data are subdivided into order channel distribution, payment amount, and order amount. Calculated data are subdivided into conversion rates and payment rates. Payment data are subdivided into payment amount, payment types, and payment results. Figure 8 is a dendrogram of on-platform business-customer transaction data.

On-platform user data are divided into member user, order user, paying user, and user retention rate. Figure 9 displays the on-platform user data.

Today, many enterprises have established big data analysis platforms to monitor and manage financial risks through big data. The specific plate distribution of analysis model for enterprise big data platform is as follows: first, the model is stratified into data control layer, process scheduling layer, and internal and external structure data layer; second, in terms of local division, the data management platform is divided into data standard, data quality, metadata, and data security; the process scheduling platform is divided

into process scheduling, monitoring and warning, and warning model; application data area is divided into user management and risk management; the big data area is divided into the big data to be processed and the big data processed; external enterprise users can be subdivided into business sand table exercise, exercise data area, subject data, and subject data area. The analysis model of enterprise big data platform is shown in Figure 10.

**2.3. Logistic Regression-based FRM Model.** Logistic regression and multiple linear regression are similar and both belong to the generalized linear model while differing in the dependent variables. The dependent variables of a binomial distribution, namely Logistic regression, can either be dichotomous or multi-classified, in which dichotomous is more commonly used and easier to explain. Thus, binary Logistic regression is most commonly used in practice.

The Logistic regression method can efficiently handle data classification problems and can be used for data monitoring and analysis. The linear regression model can be implemented by the maximum likelihood estimation method under the Logistic regression model so that the data set of binary variables can be classified [31–33]. Figure 11

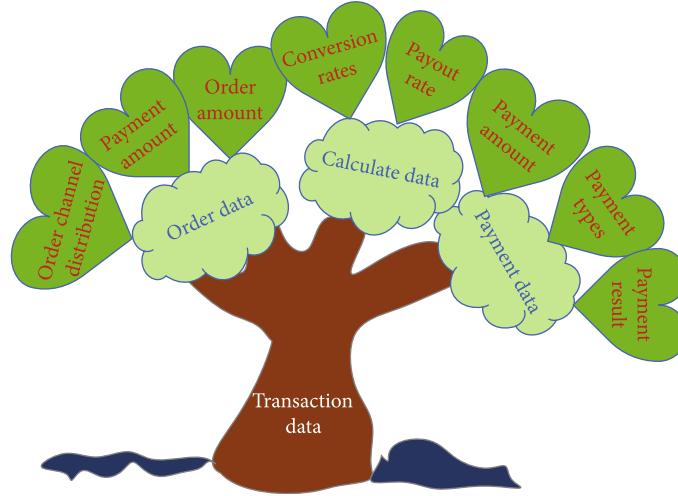


FIGURE 8: Dendrogram of on-platform business-customer transaction data.

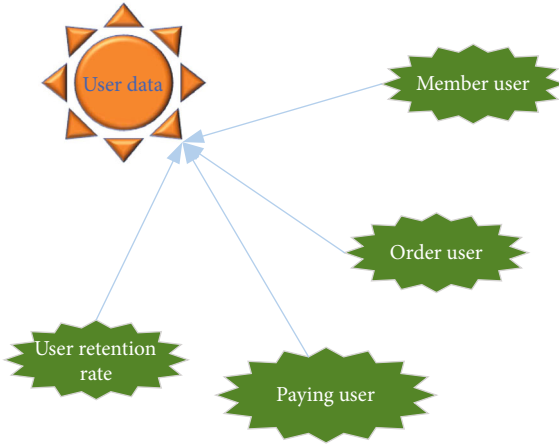


FIGURE 9: On-platform user data framework.

demonstrates the specific financial indexes of the model input data.

The parameters of the Logistic regression model function are set as follows:  $x_i$  means the interpretation of independent variables, and  $y_i$  must obey the distribution of Eq. (4):

$$P(y_i = 1|x_i) = P[\varepsilon_i \leq (\alpha + \beta x_i)] = \frac{1}{1 + e^{\alpha + \beta x_i}} \quad (4)$$

$$\beta = (\beta_0, \beta_1) \quad (5)$$

Eq. (5) is the vector  $n \times 1$ .  $\beta_0$  means constant, and  $\beta_1$  represents a coefficient vector. The enterprise FRM model implemented by the Logistic regression model is shown in Eq. (6):

$$y_i = \ln\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n \quad (6)$$

This Regression model can be transformed into Eq. (7) through calculation:

$$p_i = (y_i = 1|X) = \frac{1}{1 + \exp[-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n)]} \quad (7)$$

In Eq. (7),  $y$  means monitoring and judgment, and  $P$  refers to probability support for judgment results.

**2.4. Information Fusion-Based FRM Model.** The information fusion-based FRM model [34–36] uses a fusion algorithm to fuse different information sources. Dempster-Shafer's (DS) evidence theory is a famous information fusion method with good practicability. DS evidence theory can be used to process data from different sources and finally, transform them into output results. DS evidence theory aims to gain trust. Based on probability, the fusion processing is conducted according to specific rules, and the trust function is particularly important in this process. Thereupon, the framework of the information fusion-based FRM model is built. Here, it can be assumed that  $U$  is a nonempty set composed of multiple elements, which is the proposed framework. Then, the trust structure is set, namely, the Basic Probability Assignment (BPA) function, which can be simplified to an  $m$  function. This probability distribution has the characteristics of Eq. (8) and Eq. (9).

$$U = \{U_1, U_2, \dots, U_n\} \quad (8)$$

$$m(\emptyset) = 0 \quad (9)$$

$$\sum_{A \subseteq U} m(A) = 1 \quad (10)$$

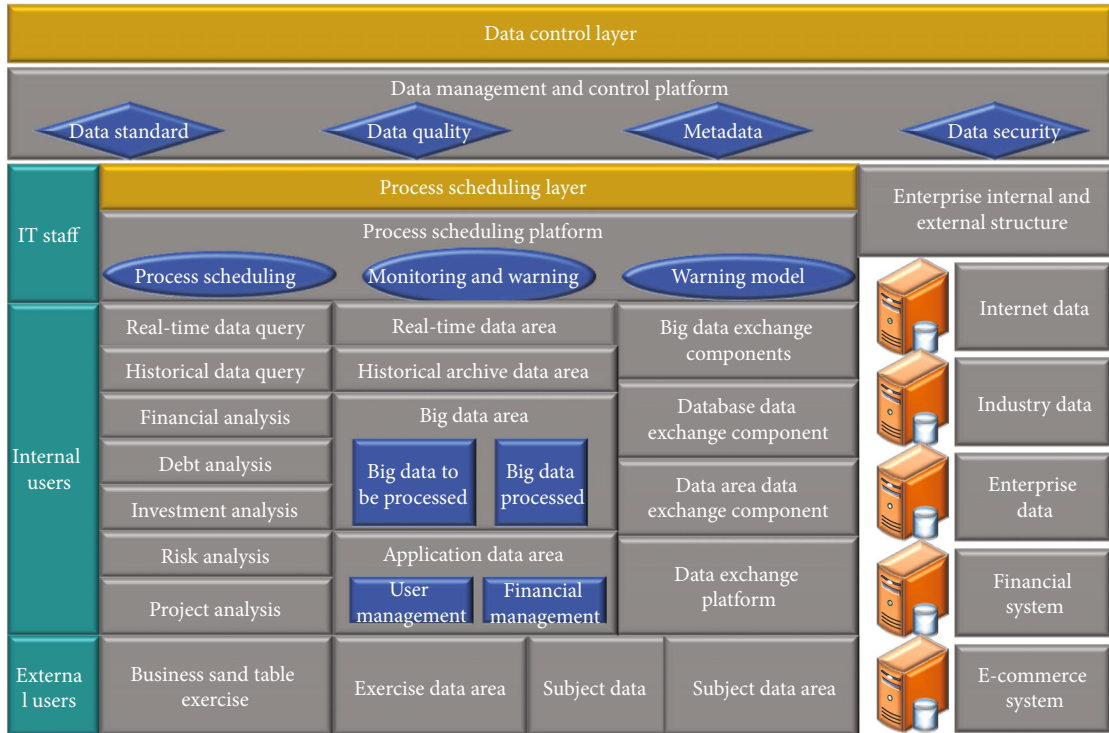


FIGURE 10: Analysis model for enterprise big data platform.

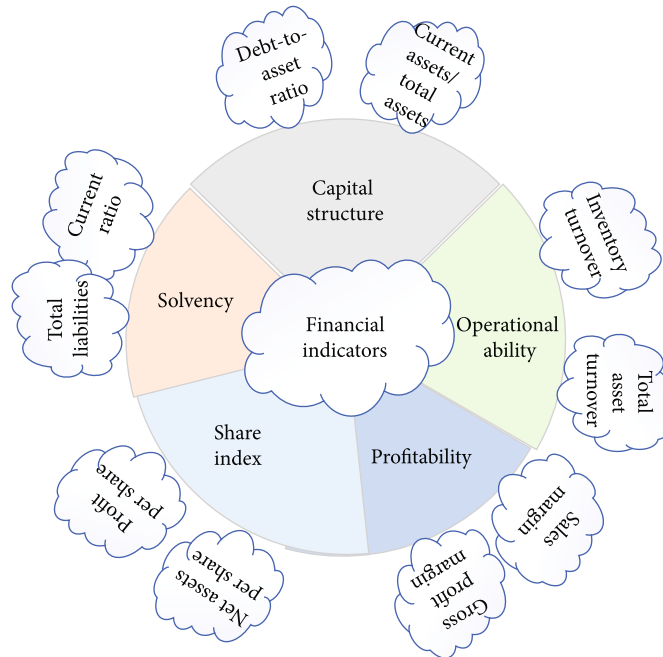


FIGURE 11: Model indexes.



Here, the subset of the U set is A, then  $m(A)$  is the probability distribution function. BeliefFunction (Bel) is calculated by Eq. (11):

$$Bel(A) = \sum_{B \subseteq A} m(B) \quad (11)$$

$Bel(A)$  is the expression of A's belief, which represents the reliability function of the U set. In Eq. (12),  $m_1$  and  $m_2$  stand for the trust degree of the two reliability functions  $Bel_1$  and  $Bel_2$  based on U set, respectively. Then, the fusion algorithm can be expressed as Eq. (12):

$$m(C) = \begin{cases} \frac{\sum_{A_i \cap B_j = C} m_1(A_i) * m_2(B_j)}{1 - \sum_{A_i \cap B_j = \emptyset} m_1(A_i) * m_2(B_j)}, & A \cap B \neq \emptyset \\ 0, & A \cap B = \emptyset \end{cases} \quad (12)$$

$$\sum_{A_i \cap B_j = \emptyset} m_1(A_i) * m_2(B_j) < 1, \forall C \subseteq U, C \neq \emptyset, m(\emptyset) = 0 \quad (13)$$

Eq. (13) is supplementary to Eq. (12).

$$K = \sum_{A_i \cap B_j = \emptyset} m_1(A_i) * m_2(B_j) \quad (14)$$

In Eq. (14), K means the degree of conflict between different pieces of evidence. Equation Eq. (15) indicates the probability of classified monitoring.

$$p = (p_0, p_1) \quad (15)$$

$p_0$  denotes the probability of financial normality under enterprise FRM.  $p_1$  refers to the enterprise financial risk probability under monitoring.

$$P_{SVM} = (p_0, p_1) \quad (16)$$

Eq. (16) stands for the probability of the SVM-based FRM model.

$$P_{logistic} = (p'_0, p'_1) \quad (17)$$

Eq. (17) expresses the probability of the Logistic regression-based FRM model.

$$m(C) = \frac{\sum_{F_i \cap F_j = C} m_1(F_i) * m_2(F_j)}{1 - \sum_{F_i \cap F_j = \emptyset} m_1(F_i) * m_2(F_j)} \quad (18)$$

Eq. (18) represents the fusion calculation based on DS evidence theory and combined with trust.

$$\sum_{F_i \cap F_j = \emptyset} m_1(F_i) * m_2(F_j) < 1 \quad (19)$$

$$\forall C \subseteq U, C \neq \emptyset, m(\emptyset) = 0 \quad (20)$$

$$m_1(F_i) = p_i, m_2(F_j) = p'_j \quad (21)$$

Eqs. (19)-(21) are supplements to Eq. (18). To sum up, the trust degree of  $m(C)$  is obtained through the SVM and Logistic regression model. Then, the information fusion-based FRM model is implemented, which can effectively monitor the enterprise financial risk and improve the model reliability; this is of great significance for obtaining the specific situation of enterprise financial risk [37, 38].

### 3. Results and Discussion

**3.1. Analysis of Enterprise Financial Risk.** Following the investigation of deposits and loans of relevant banks, the relevant conditions of several banks are investigated to ensure the data authenticity, and the relevant data are summarized in Figure 12, in which the deposit loan ratio is 69.6% in 2018, 71.3% in 2019, and 72.3% in 2020; the ratio of liquidity in 2018, 2019, and 2020 are 34.2%, 34.6%, and 33.8%, respectively; the standard value in 2018, 2019, and 2020 is 25%. Obviously, the deposit loan ratio is increasing yearly from 2018 to 2020, showing that the deposit loan ratio has a great impact on liquidity and potential financial risks. The enterprise liquidity index is shown in Figure 12.

Figure 13 indicates the risk of profitable financing situations: the asset profit margin is 1.66% in 2018, 1.68% in 2019, and 1.6% in 2020; the average bank fund is 1.38% in 2018, 1.36% in 2019, and 1.3% in 2020; the capital profit margin is 1.16% in 2018, 1.15% in 2019, and 1.1% in 2020. Apparently, from 2018 to 2020, the capital profit margin has been the lowest, while the asset profit margin has been the highest. The level of capital profit determines the profitability and reflects the enterprise FRM. According to quantitative data analysis, there is a need for enterprises to strengthen their assets to prevent potential financial risks. The enterprise profitability index is shown in Figure 13.

**3.2. Financial Risk Analysis Based on Information Fusion Technology.** This section is divided into two parts according to the sample data. The first part is the test, and the second part is to verify the accuracy of the SVM-based FRM model and Logistic regression-based FRM model. In Figure 14, (a) presents the specific financial risk probability of five enterprises under the SVM-based FRM model: the probability of enterprise 1, 2, 3, 4, and 5 is 0.85, 0.95, 0.83, 0.88, and 0.93, respectively; (b) showcases specific financial risk probability of five enterprises under Logistic regression-based FRM model: the probability of enterprise 1, 2, 3, 4, and 5 is 0.95, 1, 0.98, 1, and 0.86, respectively; (c) displays specific financial risk probability of five enterprises under information fusion-based FRM model: the probability of enterprise 1, 2, 3, 4, and 5 is 0.99, 1, 0.99, 1, and 0.98, respectively. Figure 14 reveals the monitoring results of different enterprise FRM models.

Figure 15 signifies that the risk classification accuracy under the SVM-based, Logistic regression-based, and information fusion-based FRM models are 90.22%, 90.88%, and 95.18%, respectively; while the monitoring error rate is 9.68%, 9.56%, and 4.68%. Figure 15 shows the comparison of the accuracy and error rate of different enterprise FRM models.

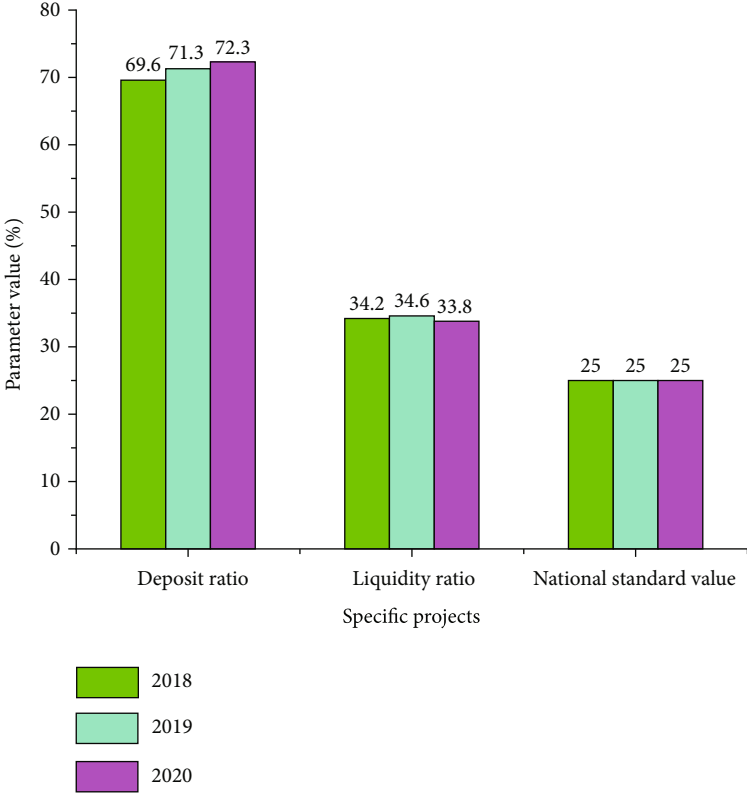


FIGURE 12: Enterprise liquidity index.

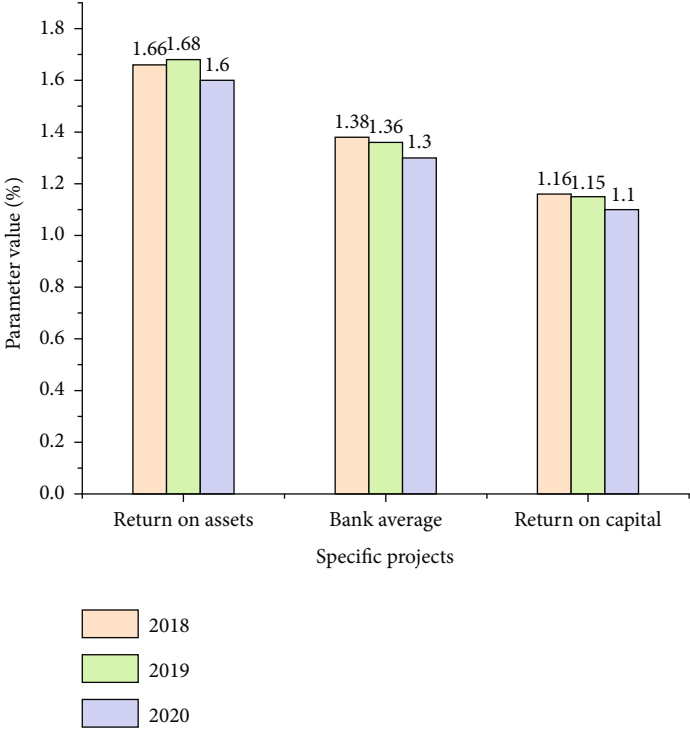


FIGURE 13: Enterprise profitability index.

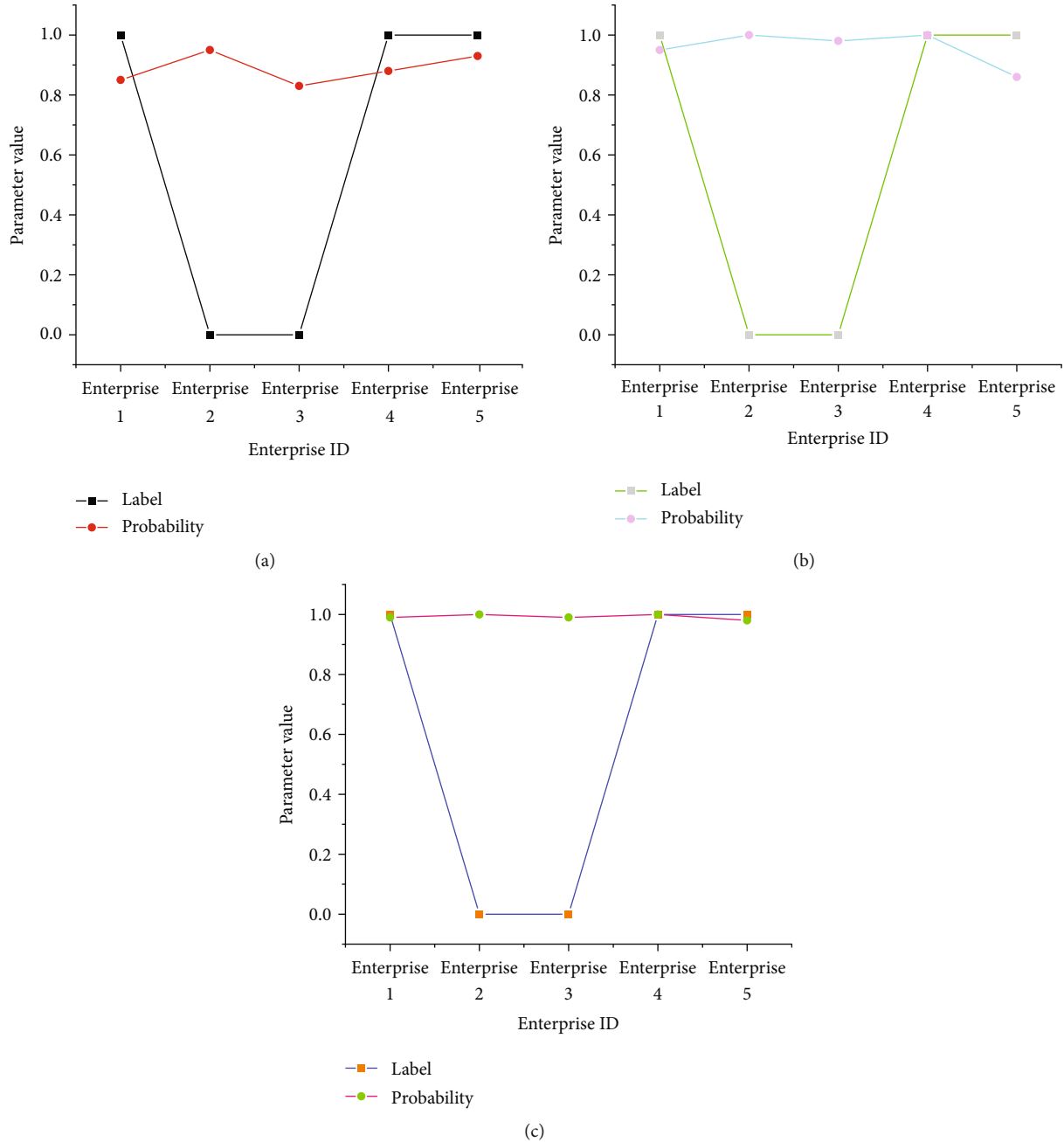


FIGURE 14: Monitoring results of different FRM models (a: Result of SVM-based FRM model; b: Result of Logistic regression-based FRM model; c: Results of information fusion-based FRM model).

In Figure 15, the ordinate represents the index value. The specific index value can be obtained through the mapping software. Figures 14 and 15 suggest that the SVM-based FRM model and Logistic regression-based FRM model have shown excellent classification effect for enterprise financial risks with an accuracy of 90.22% and 90.88%, respectively; by comparison, the information fusion-based FRM model beats the SVM-based FRM model and Logistic regression-based FRM model with an accuracy as high as 95.18%. Thus, the information fusion-based FRM model outperforms the SVM-based FRM model and Logistic regression-based FRM model; it can process and calculate the enterprise

financial risk data from different sources and obtain higher accuracy. The analysis of specific data shows that although this paper uses the real-time database to solve the uncertainty in information fusion, the fusion results from the combination of information fusion and BDM technology are still unsatisfactory; this will be the research focus in the future.

In summary, the analysis of specific sample data implies that the accuracy of the SVM-based FRM model and Logistic regression-based model in enterprise financial risk classification is relatively high, which are 90.22% and 90.88%, respectively; by contrast, the classification accuracy of information

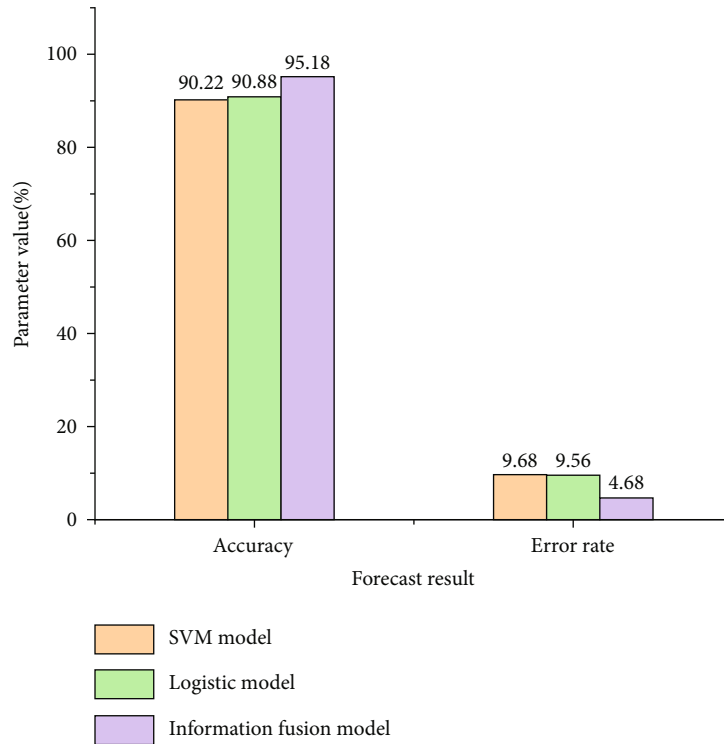


FIGURE 15: Comparisons of accuracy and error rate of enterprise FRM models.

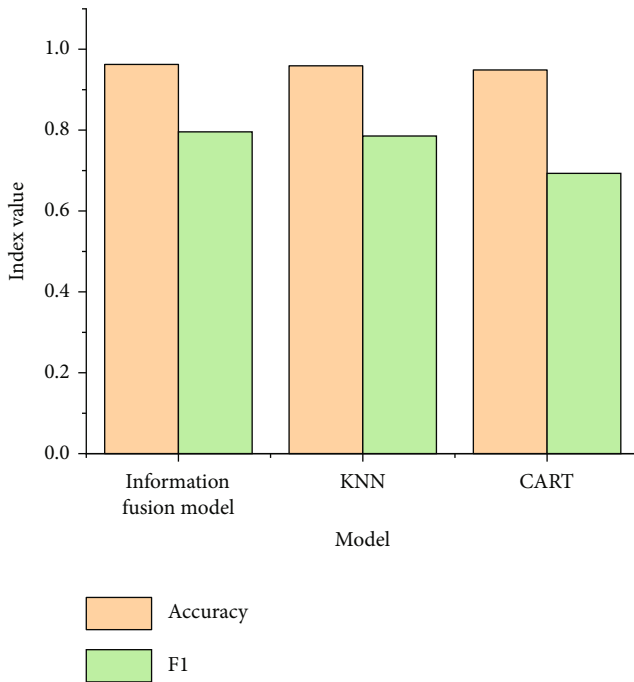


FIGURE 16: Comparison of Accuracy mean and F1 value of each model.

fusion-based FRM is significantly higher than SVM-based FRM model and Logistic regression-based FRM model with an accuracy of 95.18%. This shows that the information fusion-based FRM model is better than the SVM-based FRM model and Logistic regression-based FRM model.

Compared with other monitoring and classification models, the actual application effect of the information fusion-based enterprise FRM model will be more outstanding and have more obvious advantages.

**3.3. Performance Comparison between Information Fusion-Based FRM Model and Similar Algorithms.** This section further verifies the accuracy and classification effect of the proposed information fusion-based FRM model practical application by comparative analysis with K-nearest neighbor (KNN) and Classification And Regression Tree (CART) from Accuracy mean and F1 score. The experimental results are shown in Figure 16.

Figure 6 corroborates that the mean Accuracy and F1 value of the proposed information fusion-based FRM model is 0.9626 and 0.7958, respectively. Compared with KNN and CART models, the proposed information fusion-based FRM model has better practical applicability, a more prominent risk prediction effect, and can provide good algorithm support for enterprise FRM.

#### 4. Conclusion

The advancement of science and technology, especially, state-of-art technologies, innovates people’s lives while bringing challenges at the same time. Likewise, the rapid development and growth of enterprises are also followed by various financial risks. Aiming at the current situation of enterprise financial risk, this paper makes an in-depth study on enterprise financial risk based on BDM, SVM, Logistic regression, and information fusion technology. The

following conclusions are drawn: big data technology can provide important research methods for enterprise financial risk problems. SVM-based FRM model and Logistic regression-based FRM model can well classify enterprise financial risks with high accuracy; the information fusion-based FRM model can further improve the classification accuracy of enterprise financial risks and shows high reliability and effectiveness; additionally, different enterprise risk indexes are analyzed, finding that there is a need for enterprises to strengthen FRM under big data, especially, the management of liquidity and profitability indexes. The shortcomings of this paper are summarized: although the proposed information fusion-based FRM model improves the classification accuracy of enterprise financial risks; but its accuracy might be able to get a higher level in future research. The follow-up research will further improve the enterprise FRM performance based on the drawn conclusions.

### Data Availability

The labeled dataset used to support the findings of this study is available from the corresponding author upon request.

### Conflicts of Interest

The authors declare no competing interests.

### Acknowledgments

Phase results of 2021 Guangxi Education Science Planning Funding Key Project (2021A038); Stage results of 2019 Guangxi Higher Education Undergraduate Teaching Reform Project Key Topic (2019JGZ145); 2018 National Social Science Foundation of China (18BJY015); 2018 Academic Research Project of Master of Taxation (SWX20180010); Phase results of the Integration Innovation Project (2020JGA166) of Guangxi University for Nationalities in 2020; 2015 China-ASEAN Economic and Trade Development and South China Sea Strategic Collaborative Innovation Center Project (15&YBB06).

### References

- [1] D. Blazquez and J. Domenech, "Big data sources and methods for social and economic analyses," *Technological Forecasting and Social Change*, vol. 130, pp. 99–113, 2018.
- [2] M. K. Saggi and S. Jain, "A survey towards an integration of big data analytics to big insights for value-creation," *Information Processing & Management*, vol. 54, no. 5, pp. 758–790, 2018.
- [3] A. Oussous, F. Z. Benjelloun, A. Ait Lahcen, and S. Belfkih, "Big data technologies: a survey," *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 4, pp. 431–448, 2018.
- [4] M. Abdel-Basset, M. Mohamed, and V. Chang, "NMCDA: a framework for evaluating cloud computing services," *Future Generation Computer Systems*, vol. 86, pp. 12–29, 2018.
- [5] B. Varghese and R. Buyya, "Next generation cloud computing: New trends and research directions," *Future Generation Computer Systems*, vol. 79, pp. 849–861, 2018.
- [6] C. Stergiou, K. E. Psannis, B. G. Kim, and B. Gupta, "Secure integration of IoT and cloud computing," *Future Generation Computer Systems*, vol. 78, pp. 964–975, 2018.
- [7] S. Park, G. J. Choi, and H. Ko, "Information technology-based tracing strategy in response to COVID-19 in South Korea—privacy controversies," *JAMA*, vol. 323, no. 21, pp. 2129–2130, 2020.
- [8] M. S. Rad, M. Nilashi, and H. M. Dahlan, "Information technology adoption: a review of the literature and classification," *Universal Access in the Information Society*, vol. 17, no. 2, pp. 361–390, 2018.
- [9] J. Benitez, G. Ray, and J. Henseler, "Impact of information technology infrastructure flexibility on mergers and acquisitions," *MIS Quarterly*, vol. 42, no. 1, pp. 25–43, 2018.
- [10] Pavlo Tychyna Uman State Pedagogical University, Ukraine, I. Korol, A. Poltorak, and Mykolayiv National Agrarian University, Ukraine, "Financial risk management as a strategic direction for improving the level of economic security of the state," *Baltic Journal of Economic Studies*, vol. 4, no. 1, pp. 235–241, 2018.
- [11] K. Valaskova, T. Kliestik, L. Svabova, and P. Adamko, "Financial risk measurement and prediction Modelling for sustainable development of business entities using regression analysis," *Sustainability*, vol. 10, no. 7, p. 2144, 2018.
- [12] A. Kim, Y. Yang, S. Lessmann, T. Ma, M. C. Sung, and J. E. V. Johnson, "Can deep learning predict risky retail investors? A case study in financial risk behavior forecasting," *European Journal of Operational Research*, vol. 283, no. 1, pp. 217–234, 2020.
- [13] Q. Yang, Y. Wang, and Y. Ren, "Research on financial risk management model of internet supply chain based on data science," *Cognitive Systems Research*, vol. 56, pp. 50–55, 2019.
- [14] C. Brooks, I. Sangiorgi, C. Hillenbrand, and K. Money, "Experience wears the trousers: exploring gender and attitude to financial risk," *Journal of Economic Behavior & Organization*, vol. 163, pp. 483–515, 2019.
- [15] L. Nguyen, G. Gallery, and C. Newton, "The joint influence of financial risk perception and risk tolerance on individual investment decision-making," *Accounting & Finance*, vol. 59, Supplement 1, pp. 747–771, 2019.
- [16] C. Sathyamoorthi, M. Mapharing, M. Mphoeng, and M. Dzimiri, "Impact of financial risk management practices on financial performance: evidence from commercial banks in Botswana," *Applied Finance and Accounting*, vol. 6, no. 1, pp. 25–39, 2020.
- [17] K. Valaskova, T. Kliestik, and M. Kovacova, "Management of financial risks in Slovak enterprises using regression analysis," *Oeconomia Copernicana*, vol. 9, no. 1, pp. 105–121, 2018.
- [18] B. J. Ali and M. S. Oudat, "Financial risk, and the financial performance in listed commercial and investment banks in Bahrain bourse," *International Journal of Innovation, Creativity and Change*, vol. 13, no. 12, pp. 160–180, 2020.
- [19] K. W. Lee, "The usage of derivatives in corporate financial risk management and firm performance," *International Journal of Business*, vol. 24, no. 2, pp. 113–131, 2019.
- [20] H. Zhang, A. Khurshid, W. A. Xinyu, and A. M. Băltăţeanu, "Corporate financial risk assessment and role of big data; new perspective using fuzzy analytic hierarchy process," *Journal for Economic Forecasting*, vol. 2, pp. 181–199, 2021.
- [21] R. Myšková and P. Hájek, "Mining risk-related sentiment in corporate annual reports and its effect on financial



- performance,” *Technological and Economic Development of Economy*, vol. 26, no. 6, pp. 1422–1443, 2020.
- [22] A. Alshehhi, H. Nobanee, and N. Khare, “The impact of sustainability practices on corporate financial performance: literature trends and future research potential,” *Sustainability*, vol. 10, no. 2, p. 494, 2018.
- [23] R. Oduro, M. A. Asiedu, and S. G. Gadzo, “Impact of credit risk on corporate financial performance: evidence from listed banks on the Ghana stock exchange,” *Journal of Economics and International Finance*, vol. 11, no. 1, pp. 1–14, 2019.
- [24] J. Xie, W. Nozawa, M. Yagi, H. Fujii, and S. Managi, “Do environmental, social, and governance activities improve corporate financial performance?,” *Business Strategy and the Environment*, vol. 28, no. 2, pp. 286–300, 2019.
- [25] N. Mselmi, T. Hamza, A. Lahiani, and M. Shahbaz, “Pricing corporate financial distress: empirical evidence from the French stock market,” *Journal of International Money and Finance*, vol. 96, pp. 13–27, 2019.
- [26] S. K. Y. Augustine, “The effect of financial risk and environmental risk on mining company performance with good corporate governance as moderating variables,” *Risk*, vol. 11, no. 11, 2019.
- [27] J. Oláh, S. Kovács, Z. Virglerova, Z. Lakner, M. Kovacova, and J. Popp, “Analysis and comparison of economic and financial risk sources in SMEs of the Visegrad group and Serbia,” *Sustainability*, vol. 11, no. 7, p. 1853, 2019.
- [28] J. S. Raj and J. V. Ananthi, “Recurrent neural networks and nonlinear prediction in SVMs,” *Journal of Soft Computing Paradigm (JSCP)*, vol. 1, no. 1, pp. 33–40, 2019.
- [29] J. Xu, C. Xu, B. Zou, Y. Y. Tang, J. Peng, and X. You, “New incremental learning algorithm with SVMs,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 11, pp. 2230–2241, 2019.
- [30] B. Ghaddar and J. Naoum-Sawaya, “High dimensional data classification and feature selection using support vector machines,” *European Journal of Operational Research*, vol. 265, no. 3, pp. 993–1004, 2018.
- [31] C. Y. J. Peng, K. L. Lee, and G. M. Ingersoll, “An introduction to logistic regression analysis and reporting,” *The Journal of Educational Research*, vol. 96, no. 1, pp. 3–14, 2002.
- [32] C. Mood, “Logistic regression: why we cannot do what we think we can do, and what we can do about it,” *European Sociological Review*, vol. 26, no. 1, pp. 67–82, 2010.
- [33] S. J. Press and S. Wilson, “Choosing between logistic regression and discriminant analysis,” *Journal of the American Statistical Association*, vol. 73, no. 364, pp. 699–705, 1978.
- [34] Z. Duan, T. Wu, S. Guo, T. Shao, R. Malekian, and Z. Li, “Development and trend of condition monitoring and fault diagnosis of multi-sensors information fusion for rolling bearings: a review,” *The International Journal of Advanced Manufacturing Technology*, vol. 96, no. 1-4, pp. 803–819, 2018.
- [35] L. Zhou, C. Zhang, Z. Qiu, and Y. He, “Information fusion of emerging non-destructive analytical techniques for food quality authentication: a survey,” *TrAC Trends in Analytical Chemistry*, vol. 127, article 115901, 2020.
- [36] Q. Tian, J. Jia, and C. Hou, “Research on fingerprint identification of wireless devices based on information fusion,” *Mobile Networks and Applications*, vol. 25, no. 6, pp. 2359–2366, 2020.
- [37] Y. Gong, X. Su, H. Qian, and N. Yang, “Research on fault diagnosis methods for the reactor coolant system of nuclear power plant based on D-S evidence theory,” *Annals of Nuclear Energy*, vol. 112, pp. 395–399, 2018.
- [38] Q. Zhao, S. Wang, K. Wang, and B. Huang, “Multi-objective optimal allocation of distributed generations under uncertainty based on D-S evidence theory and affine arithmetic,” *International Journal of Electrical Power & Energy Systems*, vol. 112, pp. 70–82, 2019.

## Research Article

# Intrusion Detection Analysis of Internet of Things considering Practical Byzantine Fault Tolerance (PBFT) Algorithm

Leixia Li,<sup>1</sup> Yong Chen,<sup>2</sup> and Baojun Lin <sup>3</sup>

<sup>1</sup>University of Chinese Academy of Sciences, Beijing 100049, China

<sup>2</sup>Shijiazhuang Tiedao University, Shijiazhuang 050043, China

<sup>3</sup>Innovation Academy for Microsatellites of CAS, Shanghai 200000, China

Correspondence should be addressed to Baojun Lin; [linbj@microstate.com](mailto:linbj@microstate.com)

Received 12 October 2021; Revised 15 November 2021; Accepted 22 November 2021; Published 11 December 2021

Academic Editor: Yuemin Ding

Copyright © 2021 Leixia 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.

In order to improve the security performance and accuracy of the Internet of things in the use process, it is necessary to use the Internet of things intrusion detection method. At present, the problem of inconsistency between the accuracy of detection results and nodes is more prominent when the Internet of things intrusion detection methods are running. This paper proposes a practical Byzantine fault-tolerant intrusion detection method for the use process of the Internet of things. This method introduces the intrusion detection method and the operation function of foreign attackers on the basis of practical Byzantine fault tolerance; using the expected utility function to the corresponding benefit function of practical Byzantine fault tolerance, the results of Internet of things intrusion detection model can be effectively calculated. Finally, the experimental results show that compared with the existing intrusion detection methods, the proposed method can effectively reduce the energy consumption of the Internet of things in the operation process, can effectively reduce 14.3% and 7.8%, and can effectively reduce the energy consumption of the Internet of things in the operation process.

## 1. Introduction

The Internet of things is a separate network established in the form of self-organization. The traditional Internet of things data can not meet people's monitoring needs compared with simple ushering in the era of Internet of things intrusion detection and analysis network [1–3]. Internet of things intrusion detection analysis is to add CMOS microcameras, microphones, and other facilities on the premise of using traditional Internet of things nodes to realize the performance of image acquisition, audio, video, and other data. Realize the application of the Internet of things intrusion detection and analysis network in the monitoring field, adopt the combination of the advantages of the traditional Internet of things and the Internet of things intrusion detection and analysis network, and use the coordination function of the combination to achieve the purpose of long-term, effective, and accurate monitoring in the environment. The system can realize the effect of real-time monitoring of a wireless multimedia network. When there are problems in

the Internet of things or equipment on the network, the Internet of things intrusion detection system will receive early warning information, and the system will locate the fault information and inform the next level system [4–7].

In order to improve the efficiency of Internet of things intrusion detection and reduce the energy consumption in the use of Internet of things, this paper proposes an Internet of things intrusion detection method based on the practical Byzantine fault-tolerant algorithm. By giving priority to identifying nodes with high reliability for intrusion detection, all nodes in the network can be published at the end of detection, so as to realize the analysis of Internet of things intrusion detection.

## 2. Intrusion Detection Principle of Mobile Internet of Things

At present, the intrusion detection method for mobile network is to combine the principal component calculation method with fuzzy C uniform calculation to check the

mobile network. The fuzzy  $c$ -means method is used to reduce the data aggregation in the mobile network, and the principal component method is used to reduce the data information after the cluster. The component aggregation of the data after dimension reduction needs to be compared with the corresponding dimension number and set the value. If the set value is the same as the comparison result, an alarm will be triggered to detect the mobile network.

Using the calculation method of principal component, the variable  $X$  of the sample is changed and substituted into the low-dimensional space  $Y$  [8, 9]. The formula is as follows:

$$Y = W^T X. \quad (1)$$

In the formula, the number of samples in the mobile network is represented by  $X$ , which is composed of the number of  $M$  observation objects and the number of  $N$  columns, and represents a value in the coordinates of  $M$ . The orthogonal photographic data combined by the sample covariance matrix is represented by  $W$  and calculated in the form of sample matrix.  $T$  stands for transpose matrix.  $C$  represents the covariance value of the sample.

$$C = N^{-1} \sum_{i=1}^N (X - u_i)^2, \quad (2)$$

where  $u_i$  is the average value and the following formula exists:

$$CW_i = \lambda_i W_i. \quad (3)$$

In the formula,  $i = 1, 2, \dots, N$  and  $W_i$  represent the  $i$ -th column sample covariance matrix existing in  $W$  in the orthogonal matrix,  $\lambda_i$  represents the special values occurring in matrix  $C$ , and  $W_i$  represents the special values represented by and special values. Arrange the data in order according to the size of the data, and combine the value  $\lambda_i$  corresponding to the first  $l$  special values to obtain the  $L$ -dimensional data after dimension reduction.

Set the random correlation matrix after  $u$  initialization to take value in the interval  $[0, 1]$ , and meet the constraints of the following formula:

$$\left\{ \begin{array}{l} \sum_{i=1}^c u_{ij} = 1, \quad \forall j = 1, \dots, n, \\ \forall i, \forall j, u_{ij} \in [0, 1], \\ \forall i, \sum_{j=1}^c u_{ij} > 0. \end{array} \right. \quad (4)$$

Let  $v_i$  represent  $c$  cluster centers  $i = 1, 2, \dots, c$ , and the calculation formula of cluster center  $v_i$  is

$$v_i = \frac{\sum_{j=1}^n u_{ij} \cdot x_j}{\sum_{j=1}^n u_{ij}}, \quad \forall i. \quad (5)$$

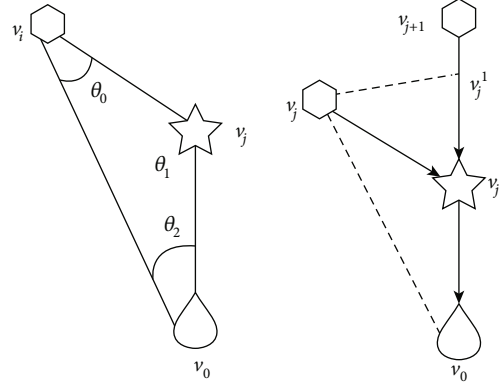


FIGURE 1: Distributed link design of IoT nodes.

Calculate the cluster center and the second from the formula. The Euclidean distance between this sample is  $d_{ij}$ .

$$d_{ij} = \|v_i - x_j\|. \quad (6)$$

The fuzzy optimal solution is obtained by the objective function  $J$ .

$$J = \sum_{i=1}^c \sum_{j=1}^n u_{ij}^m d_{ij}^2. \quad (7)$$

In the formula,  $M$  represents the fuzzy weighting factor, and the objective function  $J$  represents the sum of squares of distances between each cluster center and each sample. Set the threshold  $\zeta$ . When the target relationship value  $J$  is greater than the threshold  $\zeta$ , prove that the sample data set is intrusion data, alarm, and complete the intrusion detection of mobile network.

### 3. Optimized Deployment of IoT Nodes and Information Fusion Processing

**3.1. Optimized Deployment Design of IoT Nodes.** In order to realize real-time feedback control of online learning and monitoring of things, firstly, the optimal structure design of single-networked nodes requires the use of balanced sensor node control methods for single-network node link distribution model structure, assuming that there are  $k$  ( $k \geq 2$ ) nodes in single network disjoint paths. Between  $v$  and  $v$  ( $v$ ), why the aggregate distribution of E, SN, RN nodes is  $V = \{v_0, v_1, v_2, \dots, v_n, v_{n+1}, v_{n+m}\}$ , the redundancy capacity of a single-layer relay node is described as  $RC(v_j) = C(v_j) - w(v_i)$ , and in Figure 1 in the defined multimedia monitoring area, the tree structure and cross structure are used to periodically forward data packets to the sensor node and cluster head node. The node dispersive link design that divides the time window between the sensor node and the cluster head node is presented.

As shown in Figure 1, the Internet of things is initialized through the monitoring node link model under multimedia, and the cluster head vector group  $D_n$  of the multimedia

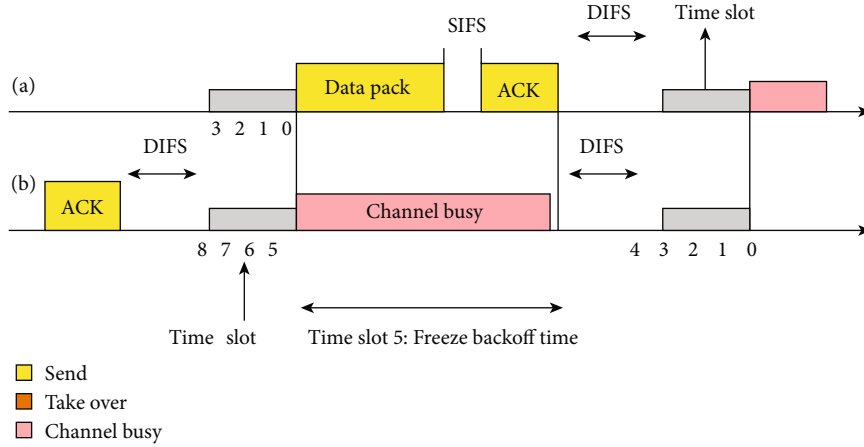


FIGURE 2: Layout of the channel allocation of two-layer relay nodes for multimedia monitoring.

sensing data node is constructed, and the data set is divided into multiple two-dimensional subregions  $A_k$  according to the width of the time window.  $A_k$  is mainly a two-dimensional information entropy group  $|d_{n-\max} - d_{n-\min}| \cdot (1/K)$ , which satisfies  $A_k = A$ . According to Figure 2, there is channel allocation layout of two relay nodes under the multimedia monitor.

If there is a certain space between the SN and sink of the Internet of things,  $k = 1$  is initialized, and the position distribution set of multimedia sensor nodes under monitoring is  $S = \{s_1, s_2, \dots, s_n\}$ , and the double-layer relay nodes are uniformly distributed. The layout method is constituted, using the obtained configuration feature distribution equation. The formula can be expressed by the following formula:

$$f(v_i, v_k) \geq 0, \quad (8)$$

$$\sum f(v_0, v_k) = 0 \quad (1 \leq k \leq n + m), \quad (9)$$

$$\sum f(v_i, v_k) = 0 \quad (1 \leq k \leq n), \quad (10)$$

$$\sum (f(v_j, v_k) - f(v_h, v_j)) = 0, \quad (n + 1 \leq k \leq n + m, 1 \leq h \leq n + m). \quad (11)$$

**3.2. Information Fusion Processing of Sensor Nodes in the Cluster.** Based on the detailed design of the corresponding configuration design of the Internet of things nodes, the data is extracted for the two-dimensional entropy feature corresponding to the sensor nodes in the cluster to alleviate the computational overhead information of monitoring multimedia [8]. The processing steps are as follows:

- (Step 1) Enter the geographic coordinates of the Internet of things SN, sink, and initialize the operation of the scope of the Internet of things monitor.
- (Step 2) Determine the sleep time. In the information data center, the distance  $d = \{d_1, d_2, \dots, d_n\}$  between SN and sink is sorted. The group corresponding to the cluster location distribution

of multimedia sensor nodes in the monitoring area is  $S = \{s_1, s_2, \dots, s_n\}$ .

- (Step 3)  $k = 1$  is initialized, the pseudorandom number adaptive sorting of the multimedia sensor sequence is determined from the current position, and  $\text{Tag} = 1$  is placed.
- (Step 4) When  $d_k \geq d_0$  and  $\text{Tag}(k) = 1$ , obtain the best position of the data cluster fusion center and move to step 5. Otherwise, the algorithm ends.
- (Step 5) Setting the maximum number of hops for the learning factor  $s_k$  of the Internet of things.  $\text{count\_max}(s, v)\chi_k$  is associated with the sensing IoT route itself and  $\text{Tag}(k) = 0$ .

$$\left| \frac{d(s_k, v_0)}{d_0} - \chi_k \right| \leq 0.5. \quad (12)$$

- (Step 6) Select nodes randomly arranged on the connection line of  $s_k$  and sink under the sensing Internet of things, and adjust the energy and resources of a network node calculated by RN in the monitoring area ( $\chi_k - 1$ ) under the sensing Internet of things, and set  $s_k$ . The next hop is marked as  $k$ , and the multimedia detection information output by the sink node in the range of  $d_0$  is quantified, fused, tracked, and identified.

### 3.3. Optimization of IoT Learning Monitoring Feedback

**3.3.1. Channel Balanced Allocation Design of the Internet of Things.** In any state error under the sensor IoT node, the node variable can be defined as  $f(T)$  and, at the same time, clarify the different characteristics of random nodes in the cluster.

$$\chi_k \leq \text{hop\_count\_max}(s_k, v_0) \left| \frac{d(s_k, v_0)}{d_0} - \chi_k \right| \leq 0.5. \quad (13)$$

The threshold for selecting data is  $\{\text{xmax}, \text{xmin}\}$ . By defining the connection line between each node SN and sink in the cluster, the maximum number of hops can be obtained and the node dimension entropy of IoT learning can be obtained. Meet the following formula:

$$d(v_i, v_0) = d(v'_i, v_0). \quad (14)$$

The average value of data near the monitoring node learned through the Internet of things has the spatial characteristics of the distribution of monitoring feedback data

$$\begin{aligned} d(v'_i, v'_j) &> \frac{1}{2}d(v_{i+1}, v_j); d(v'_i, v_j) \\ &= \frac{1}{2}d(v_{i+1}, v_j); d(v'_i, v_j) < \frac{1}{2}d(v_{i+1}, v_j). \end{aligned} \quad (15)$$

The information probabilities  $l$  and  $g$  of the unit data subset are integers, assuming that the probability weighted distance of each node determines the threshold group. After  $d(v'_i, v_j) > (1/2)d(v_{i+1}, v_j)$ , the channel allocation control function used for data transmission in a single node  $i$  has been described.

$$\begin{aligned} d(v_{i+1}, v_j)/d_0 &= l \frac{d(v_{i+1}, v_j)}{d_0} = l, \\ \frac{d(v_i, v_j)}{d_0} &\longrightarrow \lambda; \left[ \frac{d(v_i, v_j)}{d_0} \right] + 1 = \lambda + 1, \\ \frac{d(v_{i+1}, v_j)}{d_0} &= \lambda \cdot \gamma. \end{aligned} \quad (16)$$

**3.3.2. IOT Transmission Delay Control.** The linear shift channel allocation method is used, and the intelligent search algorithm for the transmission delay control of the Internet of things is used to perform the processing of the Internet of things learning monitoring feedback link equalization.

$$e = \frac{1}{n\_ever} \sum_{i=1}^{n\_ever} \text{sqr}t((x_i - x\wedge_i)^2 + (y_i - y\wedge_i)^2), \quad (17)$$

$$\text{hop\_count\_max}(v_i, v_0) = \left\lceil \frac{d(v_i, v_0)}{d_0} + 1 \right\rceil. \quad (18)$$

When the distance from the SN to the sink is arranged in descending order and the load of the object's network cluster head  $n_i$  is constant, the global balanced scheduling method is used to perform real-time feedback control of monitoring information. The fuzzy adaptive weighted control processing of multimedia sensor nodes describes the critical thresholds related to physical network learning to monitor real-time feedback.

$$E_{Tx}(L, d) = \begin{cases} LE_{\text{elect}} + L\varepsilon_{fs}d^2, & d < d_0, \\ LE_{\text{elect}} + L\varepsilon_{mp}d^4, & d > d_0, \end{cases} \quad (19)$$

$$E_{Rx}(L) = LE_{\text{elect}}. \quad (20)$$

In the formula,  $E_{\text{elect}}$  represents the global energy equalization coefficient. Through the above processing, the global equalization control of the transmission link of the Internet of things is realized, and the real-time feedback capability of multimedia information learning and monitoring is improved through the optimal configuration of the smart phone node.

**3.3.3. IoT Performance Monitoring Model.** RTFM is a working group established by IETF. We have proposed a general framework for describing and measuring single network services. And based on this framework, based on RTFM, the Internet of things intrusion detection and real-time risk based on the real-time monitoring platform of the Internet of Thousands of Things are proposed. The early warning and real-time risk early warning system model is shown in Figure 3. The model is classified into four modules: rule input system, flow collection system, data analysis system, and database system.

In this model, the traffic collection system is based on the rules set by the rule input system. Thousands of real-time Internet of things traffic filter and aggregate the Internet of things traffic, save the effective data in the database system, and provide it for data analysis. The system handles it. The data analysis system processes valid data to obtain the changes and distribution information of the Internet of things intrusion to predict, adjust, and manage the actions of the Internet of things.

The flow of the system model is shown in Figure 4.

In the shared media Internet of things, any packet that flows through the Internet of things is higher than the grouping required by the hardware configuration of the Internet of things segment business, making it impossible to process the intercepted packets in time. As long as the server used for IoT intrusion analysis is installed in the network segment interconnected with the outside world and the network card of the machine is set to "hybrid" mode, all IP data packets entering and leaving the Internet of things can be captured, and if the IP packets are analyzed and then compiled if successful, you can get the necessary information such as the source address, destination address, data volume, and application protocol. Its advantage is that it does not change the structure of the Internet of things, does not increase the load of the Internet of things, and does not occupy the resources of the Internet of things. It has nothing to do with the waiting time of the Internet of things and does not affect the network usage of user items.

This article uses Raw Socket to implement the Sniffer method is relatively simple but only cuts the packet above the IP layer and does not contain frame information. It can not meet some special requirements. From the analysis



of the current Internet of things intrusion model, it can be seen that the entire Internet of things intrusion is mainly the practical Byzantine Fault Tolerance (PBFT) algorithm traffic, and the changes in the practical Byzantine Fault Tolerance (PBFT) algorithm traffic basically reflect the changes in the entire Internet of things intrusion, so it can use the Practical Byzantine Fault Tolerance (PBFT) algorithm traffic instead of total traffic to analyze the performance of the Internet of things; that is, you can use Raw Socket to obtain traffic information.

### 3.4. Practical Byzantine Fault Tolerant Intrusion Detection Method

**3.4.1. Practical Byzantine Fault Tolerance Algorithm.** A practical Byzantine fault-tolerant algorithm is constructed through four tuples, the income function of attacker and intrusion detection system is obtained according to the model, the attack strategy group and defense strategy group are constructed according to the income function, the income matrix of the game model is obtained by using the desired function, and the Nash balance of the game model is calculated according to the income matrix.

The representative is the practical Byzantine fault-tolerant algorithm, which represents the model  $R_{\text{RDM}}$  through the four tuple attender, action, profits, and times.

$$R_{\text{RDM}} = (\text{attender, action, profits, times}). \quad (21)$$

In the formula, attender has intrusion detection system and intruder. The attacker is replaced by  $a$  and the intrusion detection system is represented by  $B$ . The intrusion detection system is different from the attacker's attack space: attacker  $a$ 's attack space.  $A_a = (N, A, M, P)$  includes normal, attack, abnormal, and preattack. Formula  $A_d = (C, R, W, D)$  represents the action space of intrusion detection system. It includes continuous execution, execution, early warning, and protection.

Set the respective representatives of  $U_a(A_a)$  and  $U_d(A_d)$  as the revenue function of attacker and intrusion detection system,  $T$  represents the total number of games, and equation (21) was converted into the following equation:

$$R_{\text{RDM}} = (a, d; A_a, A_d; U_a(A_a), U_d(A_d); T). \quad (22)$$

Make attack strategy suit and defense strategy suit according to formula (22). The expressions are as follows.

$$S_a = (S_N, S_M, S_P, S_A), \quad (23)$$

$$S_d = (S_C, S_R, S_W, S_D). \quad (24)$$

In the formula,  $S_N$ ,  $S_M$ ,  $S_P$ , and  $S_A$  represent normal, attack, abnormal, and preattack action strategies, respectively.  $S_C$ ,  $S_R$ ,  $S_W$ , and  $S_D$ , respectively, indicate continued execution, recommended execution, alarm, and protective action.  $S_{ad} = (s_a, s_d | s_a \in S_a, s_d \in S_d)$  expressed the action strategy of both sides of the bureau.

In the case of action strategy  $S_{ad} = (S_A, S_C)$ , the attacker obtains the highest benefit in the mobile network. In action

strategy  $S_{ad} = (S_P, S_D)$ , the intrusion detection system is beneficial to the attacker when the attacker attacks the moving body [10, 11]. If an attacker wants to attack a moving body on the network, the intrusion detection system will take preventive measures. Through the above analysis, the intrusion detection system and the attacker's action strategy are converted into the corresponding preference set, and the expected utility function is used to set the preference set in the interval  $[0, 1]$ .  $U(X)$  represents the effective function of both sides of the game. The behavior of  $U(X)$  is as follows:

$$U(X) = \sum_{S_{ad}} P_1 u(x_1) + \dots + P_k u(x_k). \quad (25)$$

The equation represents the probability of intrusion detection system or attacker adopting different strategies in mobile network each time and represents the benefits of various strategies. Under ideal conditions, the probability of different action processing by attacker and intrusion detection system is 0.25.  $u(x_i)$  can calculate the income matrix of the game model and get the balance of the game model.

**3.5. Optimization Method of Practical Byzantine Fault-Tolerant Intrusion Detection Method.** The adjustment value of the balance calculation method can be fed back randomly, the practical performance of the mobile network is increased by the adjusted probability of  $P_i$ , the profit factor  $P_i$  is substituted into the function formula, and the intrusion detection method is actually optimized. The setting of  $P^* = (P_1^*, P_2^*, \dots, P_i^*)$  represents the random optimal equilibrium value that occurs in the real Byzantine calculation method.  $P_i^* = (P_1^*, P_2^*, \dots, P_k^*)$  represents the situation of intrusion detection system and tactics used by attackers in a game.  $P_i^*$  represents the tactical hybrid probability set by the intrusion detection system and the attacker.  $\Delta_i$  represents a collection of hybrid tactics.

$$\Delta_i = \{P_i^* = (P_1^*, P_2^*, \dots, P_k^*)\}. \quad (26)$$

The mixed strategic space  $\Delta = \prod \Delta_i$  of the intrusion detection system and the attacker is obtained by equation (26). Through the strategy space  $\Delta$ , the probability function  $\pi_i^*$  of the intrusion detection system and the attacker's choice of action strategy is obtained.

$$\pi_i^* = \frac{\lambda u(x_i)}{\sum_{k=1}^i \lambda u(x_i)}. \quad (27)$$

In the formula,  $\lambda$  denotes the weighting coefficient. When  $\lambda$  approaches infinity, the stochastic optimization reflects that the equilibrium is close to nanometer equilibrium, and the final result can be obtained by equation (26).

The ideal probability in a normal mobile network is almost zero, and attackers use different attack methods and strategies to attack the mobile network. Formulas (1) and (14) are optimized consistently to improve the detection accuracy of the actual Byzantine intrusion detection method. Suppose  $\delta \in [0, 1]$  is representative of the profit factor. The

profit of both parties during the game is determined by the profit factor. The larger the value of  $\delta$ , the more important the intrusion detection system or the attacker attaches to the overall profit and the higher the rate of return. The smaller value of  $\delta$  indicates that the intrusion detection system or the attacker pays less attention to the overall income. If the efficiency function is imported, the following formula can be obtained:

$$U(X) = \sum_{k=1}^{\infty} (1 - \pi_i^*)^{k-1} \pi_i^* \delta^{k-1}. \quad (28)$$

Formula (28) belongs to the revenue function. Players in each game need to actively change their action tactics to maximize the revenue value, check malicious attacks in the mobile network according to the revenue function, and delete the combination of malicious nodes to improve the security performance of the mobile network [12–14].

**3.6. Disciplinary Mechanisms.** The use of disciplinary agencies poses a threat to malicious nodes in the mobile network. The purpose of the retribution mechanism is to have one node representing the attack, but the other nodes are not represented by the malicious node transmitted from the next timeslot. There are three levels of punishment.

- (1) If no malicious node is detected in the previous mobile network game, all nodes remain in the current state in the network, and if the malicious node is detected in the mobile network, it will move to the next step
- (2) Punishment agencies punish malicious nodes and keep other nodes in the mobile network in their original state during punishment
- (3) After malicious node operation, the real data  $(U_1, \dots, U_N)$  of mobile network is shown below. If there is malicious behavior in the third step, it needs to go back to the second step according to the punishment malicious node

If the malicious node  $\vartheta$  becomes normal, the maximum profit of node  $\vartheta$  in the departure time slot is  $\bar{v}_k$ , the profit of node  $\vartheta$  in the punishment  $T_k$  time is  $\dot{v}_k$ , and the profit of node  $\vartheta$  in the normal state is  $v'_k$ . The average discounted utility value  $\hat{U}_k$  of node  $\vartheta$  in mobile Internet of things can be obtained:

$$U_k = (1 - \delta)\bar{v}_k + \delta(1 - \delta^{T_k})\dot{v}_k + \delta^{T_k+1}v'_k. \quad (29)$$

The utility value of the average discount of the node  $\vartheta$  in the game when the node  $\vartheta$  is in the normal state is

$$U_k = (1 - \delta) \sum_{t=0}^{\infty} \delta^t v_k = v_k \quad (30)$$

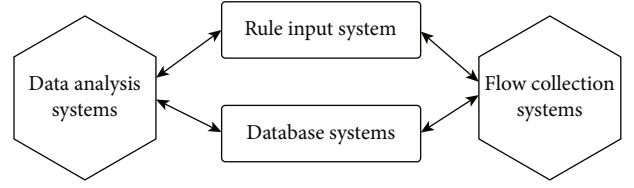


FIGURE 3: System model.

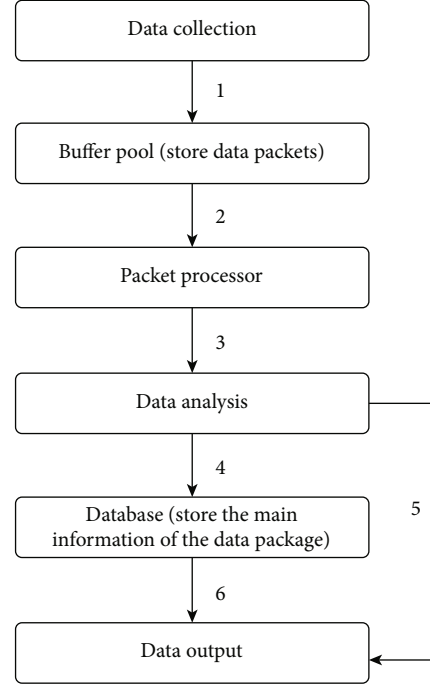


FIGURE 4: System process.

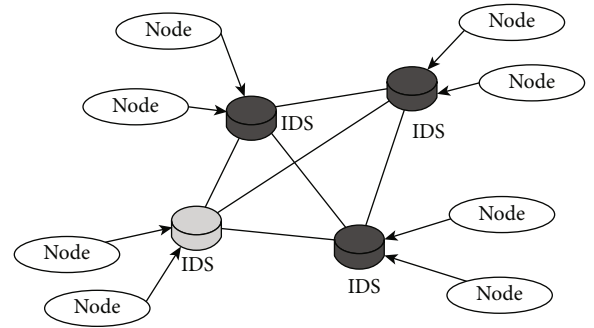


FIGURE 5: Network topology.

According to Formulas (29) and (30), the deviation profit  $\Delta U_k$  of node  $\vartheta$  in mobile network can be obtained.  $\Delta U_k$  is calculated by the following formula:

$$\Delta U_k = \hat{U}_k - U_k. \quad (31)$$

In formula (31), the deviation gain must be below zero, and the deviation gain of node  $\vartheta$  is smaller than the cooperation gain. At this point, no rational node in the mobile network deviates from the normal state.

TABLE 1: Attack types.

Attack types	Attack subtype
Normal	Normal
DoS	Back, land, Neptune, pod, smurf, teardrop, apache2, adpstorm, processtable, worm
Probe	Ipsweep, nmap, portsweep, satan, mascan, saint
U2R	buffer_overflow, loadmodule, perl, rootkit, perl, sqlattack, xterm, ps

## 4. Experimental Results and Analysis

**4.1. Network Topology.** The network cluster structure adopted in this paper is mainly composed of cluster head node and general node, which is a general topology structure in practical application. The common nodes, namely, sensor nodes, are terminal nodes in single network. The cluster head node (that is, the gateway device in a single network) manages the nodes in the cluster and reports data to the outside world. IDS operates at the cluster head node, and each cluster head node performs intrusion detection between it and other cluster head nodes based on PBFT [15]. The specific network topology is shown in Figure 5.

**4.2. Data Set Preprocessing.** The KDD Train of the NSL-JDD data set was tested experimentally. The training model of 20 percent training set allows nodes to randomly select one from the test subset of KDD test-21 to record and perform intrusion detection operations as the node at this time (Table 1).

**4.3. Experimental Evaluation Criteria.** The security performance of intrusion detection method is mainly reflected in the detection rate indicator, especially the ratio of the number of detected malicious nodes to the total number of malicious nodes in the network. Common evaluation criteria are used here: (1) TP (True Positive) indicates the number of samples that are correctly judged as positive types, (2) TN (True Negative) indicates the number of samples that are correctly identified as Negative, and (3) FP (False Positive) indicates the number of samples whose sample error is judged to be negative.

Then, Detection Rate (DR) is defined as

$$DR = \frac{TN}{TN + FN}. \quad (32)$$

The other mode of intrusion detection is energy consumption, which has the characteristics of the following: (1) EV is the energy consumption in node election, (2) EDI is the energy consumption of node intrusion detection, (3) EP is the energy consumption of node to publish detection results, and (4) EC is the statistical energy consumption of node measurement results.

Energy consumption of all nodes in the network is

$$E = \sum_i^N EV_i + \sum_i^{N'} (EID + EP) + \sum_i^N \sum_j^{N''} EC_{ij}. \quad (33)$$

TABLE 2: Simulation parameters.

Parameter	The default value
Network area size	400 × 400
Number of nodes	60~300
Number of abnormal nodes	15%-20%
Initialize weights $w_i$	1
Initialize the trusted list	node $_i, i \in (0, N - 1)$
Detection interval $\Delta t$	70 s
The elapsed time	20 min

**4.4. Experimental Scheme Design.** In order to test the effectiveness of the practical Byzantine fault tolerance algorithm proposed by ontology, the SVM algorithm is used to detect the NSL-KDD data set to obtain the detection rules. In the experimental process of this paper, the relevant rules of flow control and intrusion detection can be realized by using the microcontroller, and the Active Message layer can be used to effectively control the RF module, so as to realize the mutual communication between the communication nodes. Based on the previous single network intrusion detection methods, this paper presents a comparative experimental study. Detailed simulation environment parameters are shown in Table 2.

**4.5. Analysis of Results.** In this paper, simulation experiments are conducted on different types of network nodes and abnormal proportion nodes in the initial state, as shown in Figure 6. Experimental results obtained in different types of network states can be seen. As can be seen from the experimental results in Figure 6, due to the impact of dimensionality reduction, the accuracy of the training data set will also be affected to a certain extent. Therefore, the two-dimensional reduction detection method is used to optimize the model used by the algorithm in this paper to ensure that the detection rate of colleges and universities can be achieved under the condition of a large number of network nodes.

Figure 7 shows the error rate in a single network with different proportions of the three methods attacking nodes. As can be seen from Figure 7, PBFT can modify the detection errors of a single node, and the introduced matching protocol has the lowest error rate. However, false positives occur on the network only when detection errors occur on nodes above  $M + 1$ . On the other hand, IIDS and TDTC methods only rely on the detection results of a single node, especially the TDTC method which has the highest false

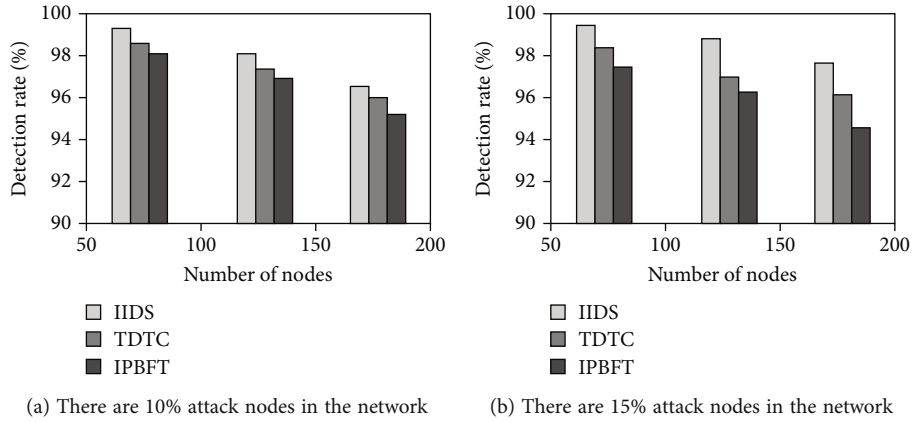


FIGURE 6: Detection rates of three intrusion detection methods under different proportions of abnormal nodes.

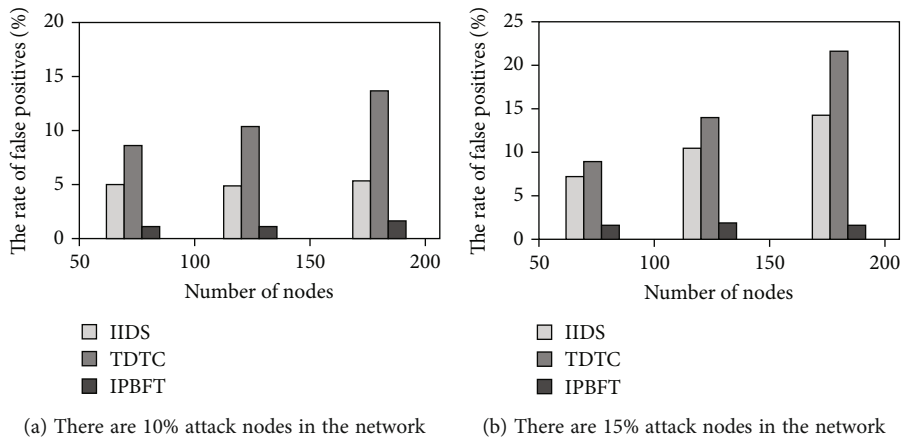


FIGURE 7: False positive rate of three intrusion detection methods under different proportions of abnormal nodes.

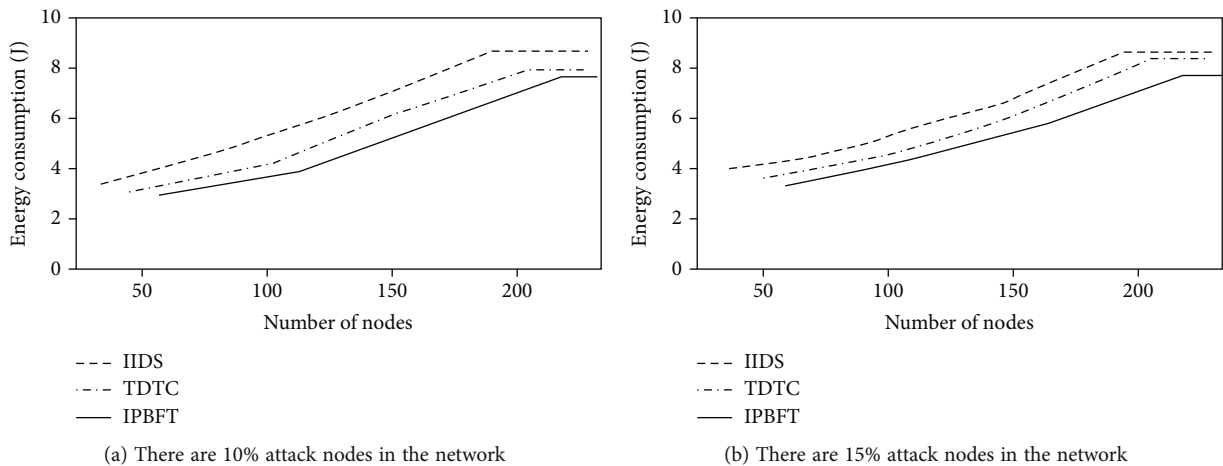


FIGURE 8: Energy consumption of three intrusion detection methods for different proportions of abnormal nodes.

positive rate of the three methods after the accuracy of the data set is reduced.

By evaluating the network intrusion detection method, the network energy consumption can be obtained. Under the same experimental conditions, the algorithm proposed

in this paper can complete the algorithm test under the environment of as little energy consumption as possible. As shown in Figure 8, after setting 15% and 20% of the total number of different nodes for different attackers, the energy consumption of all nodes in the experimental system is

compared and analyzed to ensure that the energy consumption of IPBFT on network nodes is as low as possible. Compared with the IDS algorithm, the energy consumed by IPBFT algorithm can be reduced to 13.5% when the attacking node process takes up 25%, and the network attacking node can also be reduced from 20% to 12.5%. Compared with the TDTC algorithm, IPBFT algorithm can reduce energy consumption by 6.9% when attack nodes account for 15% of network usage. Compared with the other two methods, the IPBFT algorithm in this paper consumes less energy.

## 5. Conclusions

The Internet of things has been widely used in many fields such as life service, machinery manufacturing, medical treatment, economy, and business, but it will lead to the loss of data and information and lead to serious losses when it is invaded by external hackers, viruses, and viruses. In existing mobile Internet of things in the intrusion detection, test results are not accurate, and node inconsistent problem, through the practical Byzantine fault tolerance in the mobile Internet of intrusion detection method, can effectively solve the serious problems that exist in the existing methods, through the experimental results which show that the method can effectively improve the safety and accuracy of the mobile Internet of things.

## Data Availability

Data sharing is not applicable to this article as no data sets were generated or analyzed during the current study.

## Conflicts of Interest

The authors declare no conflicts of interest.

## Acknowledgments

This study is sponsored by Innovation Academy for Microsatellites of CAS.

## References

- [1] S. A. Rahman, H. Tout, C. Talhi, and A. Mourad, "Internet of things intrusion detection: centralized, on-device, or federated learning?," *IEEE Network*, vol. 70, no. 5, pp. 5057–5070, 2020.
- [2] B. B. Zarpelão, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga, "A survey of intrusion detection in Internet of things," *Journal of Network & Computer Applications*, vol. 68, no. 4, pp. 4089–4093, 2017.
- [3] M. Ge, N. Syed, X. Fu, Z. Baig, and A. Robles-Kelly, "Toward a deep learning-driven intrusion detection approach for Internet of things," *Computer Networks*, vol. 21, no. 7, pp. 30–35, 2020.
- [4] P. Nimbalkar and D. Kshirsagar, "Feature selection for intrusion detection system in Internet-of-things (iot)," *ICT Express*, vol. 7, no. 2, pp. 177–181, 2021.
- [5] J. Balasundaram and M. Pushpalatha, "A novel optimized bat extreme learning intrusion detection system for smart Internet of things networks," *International Journal of Communication Systems*, vol. 6, no. 4, pp. 6125–6133, 2021.
- [6] R. Fantacci, F. Nizzi, T. Pecorella, L. Pierucci, and M. Roveri, "False data detection for fog and Internet of things networks," *Sensors*, vol. 19, no. 19, pp. 4235–4507, 2019.
- [7] F. Medjek, D. Tandjaoui, I. Romdhani, and N. Djedjig, "Message from the SmartData-2017 steering chairs," in *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pp. 735–742, Exeter, United Kingdom, 2017.
- [8] G. Thamilarasu, A. Odesile, and A. Hoang, "An intrusion detection system for Internet of medical things," *IEEE Access*, vol. 8, pp. 181560–181576, 2020.
- [9] A. J. Siddiqui and A. Boukerche, "Tempocode-iot: temporal codebook-based encoding of flow features for intrusion detection in Internet of things," *Cluster Computing*, vol. 24, no. 1, pp. 17–35, 2021.
- [10] M. Ahmad, Q. Riaz, M. Zeeshan, H. Tahir, and M. S. Khan, "Intrusion detection in Internet of things using supervised machine learning based on application and transport layer features using unsw-nb 15 data-set," *EURASIP Journal on Wireless Communications and Networking*, vol. 73, 627 pages, 2021.
- [11] A. Bamou, "Intrusion detection in the Internet of things," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 1.5, pp. 1–7, 2020.
- [12] A. Yang, H. Liu, Y. Chen, C. Zhang, and K. Yang, "Digital video intrusion intelligent detection method based on narrow-band Internet of things and its application," *Image and Vision Computing*, vol. 97, no. 4, pp. 103914–103957, 2020.
- [13] J. Arshad, M. A. Azad, M. M. Abdellatif, M. H. U. Rehman, and K. Salah, "Colide: a collaborative intrusion detection framework for Internet of things," *IET Networks*, vol. 8, no. 1, pp. 3–14, 2019.
- [14] S. Halder, A. Ghosal, and M. Conti, "Efficient physical intrusion detection in Internet of things: a node deployment approach," *Computer Networks*, vol. 154, no. MAY 8, pp. 28–46, 2019.
- [15] S. Deshmukh-Bhosale and S. S. Sonavane, "A real-time intrusion detection system for wormhole attack in the rpl based Internet of things," *Procedia Manufacturing*, vol. 32, pp. 840–847, 2019.



## Research Article

# Medical Record Encryption Storage System Based on Internet of Things

Yamei Zhan<sup>1</sup> and Zhaopeng Xuan<sup>2</sup> 

<sup>1</sup>Medical Records Room, The First Hospital of Jilin University, Changchun 130000, China

<sup>2</sup>Hand, Foot Surgery, The First Hospital of Jilin University, Changchun 130000, China

Correspondence should be addressed to Zhaopeng Xuan; [xuanzp@jlu.edu.cn](mailto:xuanzp@jlu.edu.cn)

Received 9 October 2021; Revised 2 November 2021; Accepted 5 November 2021; Published 24 November 2021

Academic Editor: Chin-Ling Chen

Copyright © 2021 Yamei Zhan and Zhaopeng Xuan. 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 Internet of Things takes data as the center, and its core is data storage and management. With the emergence and rapid development of wireless communication technology, with the huge number of terminals in human society, massive data will be generated. Undoubtedly, data storage and management technology will attract much attention. In view of this, this paper proposes a data storage scheme based on the Internet of Things. This paper introduces the Internet of Things technology, designs it from the perspective of the massive data storage system of the Internet of Things, realizes the intelligent processing of data storage, and provides security guarantee for information services. By combing the business process management of doctors, nurses and patients, this paper constructs a medical record encryption management system, makes a comparative analysis before and after the system goes online, and carries out simulation experiments. The simulation results show that (1) the cost of paper is significantly reduced, and the related forms of medical records are more unified and standard, (2) medical record inquiry and reading are more convenient and controllable, and (3) the safety of medical records is well guaranteed. Except that the relevant doctors and nurses of patients can view the relevant medical records, and others have no authority to query and access them. Therefore, the encrypted medical record storage system based on Internet of Things technology can effectively solve the collection, statistics, and integration of patient treatment information, which can be summarized into a unified, shared, and interconnected electronic medical record management system to realize the collection of patient treatment information in the whole process.

## 1. Introduction

At present, e-government, medical, and health information systems need to store personal data. Although the use of personal data is more humanized, the personal data stored in the system may be abused by operators or system administrators. In this context, relevant prevention and control technologies came into being. At the same time, with the development of cloud computing, the demand for arithmetic operations on encrypted personal data is also growing rapidly. The problem of medical record data storage: the electronic medical record system requires long-term preservation of patient information and can be obtained at any time. However, considering the limited life of computer hardware and database capacity, the data cannot be saved

online for a long time. How to realize the reuse of data after data transfer and make the patient information separated from the database still maintain a personal centered structure is one of the difficulties encountered in the process of medical record data storage. Medical record data sharing: medical record data sharing is to ensure that the electronic medical record system can automatically identify the data from other systems and access the medical records written by other hospitals. How to realize the data sharing between different information systems in the same medical institution and the information reuse between different medical institutions is another difficult problem faced by the electronic medical record system.

With the continuous development of social economy, the informatization of all walks of life has been gradually

constructed and improved, which greatly facilitates people's daily life [1, 2]. As far as individuals are concerned, they will generate corresponding information during medical treatment, such as medical information, condition information, and medication information. Traditional management methods are usually filled in manually. On the one hand, safety cannot be guaranteed; on the other hand, it is not conducive to retrospective comprehensive diagnosis of the disease [3, 4]. The development of information technology has spawned a variety of medical business systems, but also convenient for individuals to see a doctor. All kinds of information carriers, from PC Web application to mobile app and WeChat mini program, have greatly enriched the medical experience and made it convenient for people to see a doctor. However, similar to other information systems, information security is still caused by external attacks [5]. The continuous development of cloud computing technology also makes people gradually realize the importance of personal information. In the general trend of hospital management, how to ensure the security of information is extremely important [6, 7]. Different doctors and different patients have different actual needs; so, these issues need to be considered comprehensively [8, 9]. In particular, in the actual treatment process, it is more often to search for relevant cases and provide targeted rescue according to the history and allergy history of existing patients, which delays the treatment time to a certain extent [10, 11].

The continuous development of medical informatization has gradually changed from the initial information flow to the collection of "patient-centered" medical data [12, 13]. In view of these needs and limitations, this paper puts forward a practical encrypted data processing system based on Residents' electronic medical records, combined with cloud storage technology and traditional homomorphic encryption system, which can realize the operation of data without decryption, prevent data leakage, and greatly protect the security of personal information.

## 2. Demand for Electronic Medical Record Whole-Process Management System

### 2.1. Electronic Medical Records

*2.1.1. Definition of Electronic Medical Record.* To electronic medical records, it is different with the traditional medical record, medical institution oriented personnel of the diagnosis and treatment of patients with accordingly, and intervention in the use of information guidance; at the same time, auxiliary by text, graphics, data, etc., through images, oscillogram of multimedia information such as records, covers the corresponding information resources in the process of patients in the hospital the whole.

*2.1.2. Problems Existing in Electronic Medical Records.* Need to be worthy of the electronic medical record still exists some limitations, such as (1) electronic medical record data storage, because this is an incremental process, and when people in the clinic is the cumulative again, if you have to go to the situation of the hospital, you will need to incremental

updates, but is limited by factors such as hardware capacity and service life, the data cannot be stored for a long time, and how to realize the limited storage and effective transfer of personal electronic medical records is extremely important and also a difficult problem worthy of study [14, 15]. (2) The sharing of electronic medical records, ordinary people in different institutions for medical treatment, different institutions have different forms, writing habits, and how to effectively and comprehensively use, is also an important and urgent problem to be solved. (3) Security of electronic medical record: electronic medical record contains many personal privacy issues such as physiology, which are easy to be leaked. At the same time, there are certain security risks in the process of information transmission.

*2.2. Manage Requirements.* The electronic medical record of ordinary people is related to the whole process of personal medical treatment, involving the corresponding data collection, query and analysis, etc. How to carry out the comprehensive management and storage of relevant information realizes multiple summaries and analysis and mining of information, so as to support doctors' medication and analysis, etc. [16, 17].

*2.3. Cloud Storage.* So-called cloud storage is based on the storage of the cloud computing technology, the corresponding data resources stored in the cloud, for others to share and use methods; on the one hand, from the user's own, cloud storage is very convenient, does not need to prepare the corresponding hardware equipment, and storage space can meet, easy access, and high efficiency, can realize the backup and so on; on the other hand, due to the high efficiency of cloud storage, the access efficiency is also high. However, data resources in the cloud have certain security risks, which should be paid attention to [14, 15, 18].

### 2.4. Homomorphic Encryption

*2.4.1. Homomorphic Encryption Technology.* Homomorphic encryption is an encryption method that encrypts a specific ciphertext by calculating the ciphertext. In turn, the ciphertext can be decrypted by the corresponding inverse operation. From another perspective, this technology allows people to perform operations such as retrieval and comparison among encrypted data to obtain correct results without the need to decrypt the data during the entire process. Its significance lies in truly fundamentally solving the problem of confidentiality when entrusting data and its operations to a third party.

*2.4.2. The Principle of Homomorphic Encryption.* Assume that the encryption operation is  $E$ , the plaintext is  $M$ , and the encryption results in  $E$ , as shown in Formula (1),

$$e = E(m), m = D(e). \quad (1)$$

If there is operation  $F$  for plaintext, it can be constructed for  $E$ , as shown in Formula (2):

$$F(e) = E(f(m)). \quad (2)$$

2.4.3. *The Realization of the Homomorphism Algorithm.* Encrypt(pk,  $m$ ), encrypted plaintext message  $m \in \{0, 1\}^*$ , calculates  $E(m) = c = m + p + rpq$ , and the corresponding ciphertext can be obtained.

Decrypt(sk,  $c$ ) calculates  $D(c) = m = c \bmod q$ .

Retrieval( $c$ ): Retrieval( $c_i - c_{i_{\text{index}}}$ ) mod  $q$ , the algorithm is a retrieval algorithm.

(1) *Encryption Process.* Firstly, the corresponding plaintext should be divided into subunits according to the corresponding security requirements, and the encryption operation should be carried out according to the corresponding groups, as shown below:

- (1) Form the corresponding prime number  $P$  through random number and select fixed prime number
- (2) Divide the corresponding messages into plaintext and group them accordingly
- (3) Generate a random number  $R$
- (4) Use encryption algorithms  $C = \sum_i c_i = \sum_i (m_i + P + PQR_i)$  to calculate the ciphertext  $C = c_1, c_2, \dots, c_l$

(2) *Decryption Process.*

- (1) After receiving ciphertext  $C$ , users group ciphertext  $C$  to obtain  $C = c_1, c_2, \dots, c_l$
- (2) Use key  $P$  and decryption algorithm  $m_i = c_i \bmod p$ , to calculate  $m_i$
- (3) Get the plaintext message  $M = m_1, m_2, \dots, m_l$

2.5. *Management Process.* The whole process management of hospital electronic medical record information is divided into three parts, namely, the external authority management of the electronic medical record system based on the Internet of Things, internal closed-loop management, and other systems of the hospital health information exchange standard (Health Level Seven, HL7) heterogeneous database middleware. ① External authority management includes label printing, label verification, fingerprint identification, and identity authentication; ② internal closed-loop management includes outpatient medical records, clinical treatment, surgical schedule, appointment sign-in management, follow-up management, cycle summary, laboratory label management, and cycle management and other links; ③ other systems of the hospital HL7 heterogeneous database middleware: the electronic medical record system and the hospital's existing hospital information system (hospital information system, HIS), laboratory information system (laboratory information system, LIS), image archiving and transmission system (picture archiving and communication systems, PACS), radiology information system (radiology information system, RIS), ECG acquisition, blood transfusion management, and other clinical support systems and business management system data are transformed through HL7 heterogeneous database middleware and realize the functions of information sharing and

mutual visits, as well as data mining and statistical analysis. For patients, the management of medical records can be divided into three basic parts, mainly including the external rights of medical records, internal rights management, and data exchange middleware [19].

(1) External access rights include label inspection, fingerprint identification, and identity authentication; (2) internal rights management mainly includes outpatient medical records, clinical diagnosis and treatment, follow-up management, and cycle management; and (3) data exchange middleware: the corresponding data management system is shared and transmitted to realize the data sharing, query, access, data mining, and statistical analysis of patients' corresponding information. The specific process management of medical record information management is shown in Figure 1

### 3. Construction of Paperless Electronic Medical Record Management System

Patient-centered, the Internet of Things is shared based on the Internet of Things using rfid, laser scanning, global positioning, and other technologies and equipment in the Internet of Things technology, to realize information transmission and exchange and to complete intelligent positioning, identification, tracking, management, and monitoring. It adopts a three-tier network architecture of browser/server (B/S) mode, Windows Server server operating system, 10 Gigabit Ethernet, and Oracle system for database. In the network system deployment plan, a clinical information system server with electronic medical records as the core, a certificate authority (CA) electronic signature management server, a time stamp server, various user terminals in the hospital, a medical record high-speed camera, and a computer is specially set up in the network system deployment plan. Mobile handheld devices, etc., seamlessly record patient medical information data throughout the entire process, collect, count, and integrate data to ensure integration and homogeneity, overcome many problems in traditional medical record management, and complete one-stop electronic medical record management with unification of data format and data sharing system. The paperless electronic medical record management system has good man-machine dialogue and powerful functions. The medical record entry interface is completely consistent with the actual work medical record homepage.

#### 3.1. Main Technical Equipment

- (1) RFID technology. Radio frequency identification (RFID) technology of electronic tags is the communication technology of target and corresponding system through the identification and data collection of infinite signals. The Internet of Things can connect corresponding devices, which is an extension and extension of Internet technology. Medical records set up a unique RFID code and can achieve rapid search and positioning

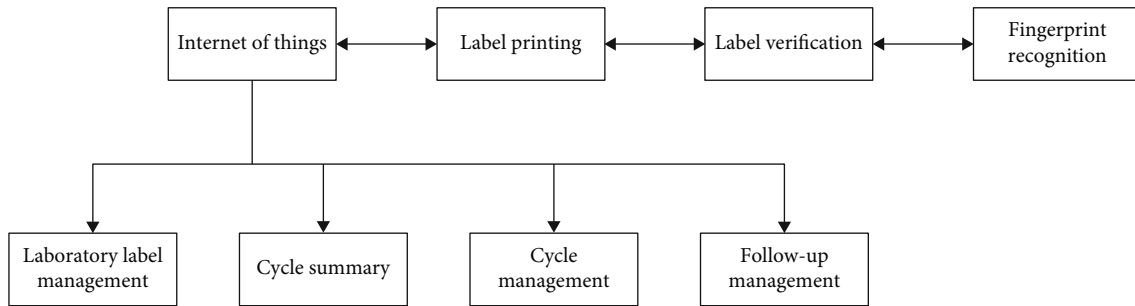


FIGURE 1: Whole process management flow of electronic medical record information.

- (2) CA electronic signature. Use the corresponding files for CA signature storage management, when the system documents are generated, that is, CA signature, first of all, the doctor in the medical record input CA signature, using the corresponding virtual printing technology method in different systems for sampling and identification, in a standard and correct format of electronic medical record management. However, for the CA signature of ordinary people, HD photography can be imported and transmitted to the corresponding system through the corresponding equipment, to achieve the preservation of paper files, and at the same time assist with the original handwriting signature and time stamp mutual authentication; when conditions permit, the corresponding recordings and photos are left for digital confirmation
- (3) Timestamp server. Timestamp service is based on the certification, the national center for timing, and punctual system according to the time of the trusted timestamp, time monitoring system to ensure the accuracy of time, use of time to clear and unified, doctors and patients, makes the electronic medical record access request, and needs to undertake the corresponding authentication, with appropriate permissions and clear requirements, to allow for a visit. The specific service working architecture is shown in Figure 2:
- (4) High-definition shooting of paper medical records and mobile terminals. The basic process of medical record management is to realize the corresponding input and digitization of clinical diagnosis and treatment report by scanning and collecting paper medical record information by using corresponding equipment. On this basis, the corresponding supplement of voice, image, signature, and report can be realized
- (5) Health Level Seven (HL7) heterogeneous data dedicated interface for Iot middleware. The text data conversion interface flow of the paperless electronic medical record management system is shown in Figure 3

3.2. *System Function Modules.* The paperless medical record management system includes doctor's work, nursing work, disease prevention work, medical record quality and safety management, medical record remote borrowing and statistical analysis of scientific research, and other functional modules. Module relationship of paperless electronic medical record management system is shown in Figure 4.

- (1) Doctor work and nursing work module. ① Doctor work module, including outpatient doctor workstation and inpatient workstation, designed according to the diagnosis and treatment process, the patient is admitted to the hospital to establish the medical record home page information, and then the doctor receives the consultation and collects the patient's family medical history, past medical history, various examinations, treatment records, and drug allergies through the system. If the patient needs surgical treatment, the system is connected to the surgical anesthesia management system to automatically read the informed consent and PACS report documents. At the same time, the mobile nursing terminal of the nurse workstation uploads the collected nursing documents, surgical records, rescue records and nursing care operation records, etc.; when the patient is discharged from the hospital, the discharge record is generated, and the electronic health file is completed; ② nursing work module: according to the nurse's execution of the doctor's order and the scan code confirmation of the test specimen, the daily, monthly, and annual workload and total of a single nurse can be counted. The workload of each ward of the hospital provides a reference basis for the performance evaluation of individual nurses and the hospital. The nursing workstation can also record the patient's specimens for examination, the execution of hospital orders, and the disinfection of medical equipment. The paperless electronic medical record management system adopts closed-loop management, which greatly reduces the repeated entry operations of medical workers and has stored a large amount of practical data in the database. Doctors rarely need to type in their work and only need to use the mouse to click on the function. The key and template describe the segment, just input, and modify certain specific content

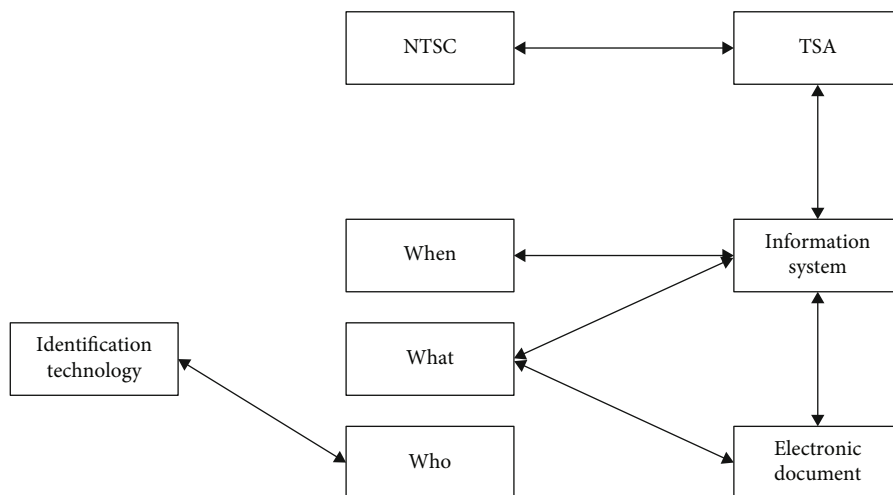


FIGURE 2: Timestamp server working architecture.

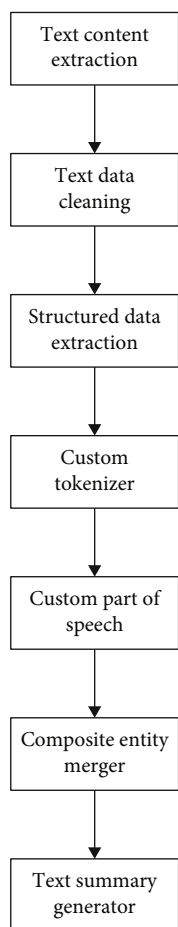


FIGURE 3: Interface flow of text data conversion in paperless electronic medical record management system.

to the disease monitoring and reporting system in the doctor’s workstation with his default authority. The system automatically reads patient information from the paperless electronic medical record management system and performs heterogeneous data through data middleware homogeneous processing, screening correct patient information, and filling in the diagnosis information related to the disease after the doctor’s review; ② hospital infection registration and review submodule: hospital infection management personnel access the paperless electronic medical record management system according to their authority, sorting and patient-related clinical information on hospital infections can be screened, reviewed, and analyzed, and finally generated hospital infection statistical reports. The reports include information such as urethral intubation use, ventilator use, multidrug resistance monitoring, and pathogenic microorganism monitoring

- (3) Medical record quality and safety management module. This module mainly includes two submodules: medical record form review and medical record safety management. ① Medical record form review submodule: when the patient is discharged from the hospital, the head nurse will organize the medical record and check the completeness of the medical record and the order of the medical record through the mobile nursing terminal. After the medical record is issued by the doctor in charge at the doctor’s workstation, the department director and department quality control the personnel conduct the review and then send it to the medical record room. The medical record room staff reviews each electronic medical record. The system also sets up a quality control expert spot check work interface; ② medical record safety management submodule: medical records are archived within 3 days from the day the patient is discharged. The system reminds doctors of necessary tasks. After the medical records

- (2) Disease prevention work module includes 2 submodules of disease reporting and review and hospital feeling registration and review. ① Disease reporting and review submodule: the doctor on duty logs in



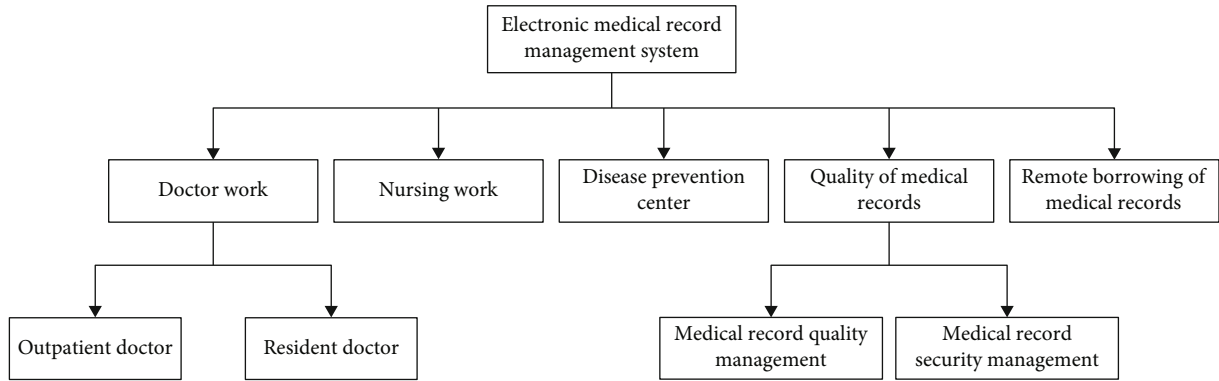


FIGURE 4: Module relationship diagram of paperless electronic medical record management system.

are filed, no one is allowed to damage, destroy, alter, forge, or steal medical records at will, to ensure the integrity and accuracy of the electronic medical records, and not to borrow or modify them at will

- (4) The remote borrowing and reading module of medical records. This module includes permission setting and loan application review and statistics. Doctors authorized by the system can check the patient's medical records, medical history, treatment measures, and insurance details. After the medical record digital filing system is created, digital network resources can be shared. Different personnel can simultaneously access the same medical record in different places. Multiple departments borrow the same medical record at the same time; so, clinicians do not need to go back and forth to the medical record department to read the paper medical records, which saves time for inquiry and retrieval, support discharge follow-up and patient online service platform for mutual visits through interface data, and also provide patients with electronic data copy services such as medical imaging examination images, surgical videos, and interventional operation videos. The current service items that are gradually introduced include some operations of the surgical anesthesia system, system image data and surgical records, PACS original acquired images of digital imaging and communication of medicine (DICOM) 4 format sequence data sets, detailed report data of biological tests, and critical data in emergency and intensive care
- (5) Scientific research statistics module includes the home page search of medical records and the full text search of medical records. Mobile medical devices and wearable devices can not only detect and track personal health data but also help diagnose diseases. Home monitoring devices such as electronic blood pressure monitors have become popular. Wearable devices help big data collect high-quality data, provide healthy decision-making and optimize treatment effects, provide diet adjustment and medical

care solutions, and provide hospitals and scientific research institutes with valuable scientific research data. The data mining statistical analysis function can convert various data streams into standardized data and conduct a comprehensive analysis of the obtained medical big data in a unified manner

A comprehensive analysis of the medical big data is obtained.

As shown in Figure 5, the system in this article includes five modules, namely, the client, the random number generation center, the ciphertext storage center, the decryption module, and the computing center. It is assumed that the modules are independent of each other, and the communication channel between the client and the server is secure.

The functions of each module are as follows:

Client defines the polynomial function used to perform ciphertext operations and sends it to the random number generation center to obtain statistical information about personal data. During the whole process, the client does not know the intermediate process and can only obtain the final result through the above function.

Random number generation center: input function  $f$  and output random number sends it to decryption module, ciphertext storage center, and computing center through steps (1), (2), and (3). Ciphertext storage center protects the privacy of data, encrypts the plaintext, and then passes the encrypted data to the computing center. Without the decryption key, the original data cannot be learned.

Decryption module possesses the decryption key of the system, receives the ciphertext passed from the computing center and calculate it, then randomizes the result with the random number passed by the random number generation center and then passes it to the computing center, and finally decrypts the final result passed from the computing center and return it to Client.

Computing center receives the ciphertext, polynomial function, and random number passed by other modules, performs addition and multiplication calculations on the encrypted ciphertext, then calls the decryption module to decrypt the calculation result, and finally encrypts the final result and transmits it to the decryption module.

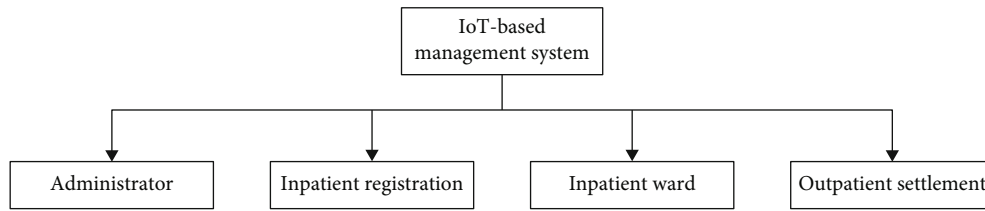


FIGURE 5: The overall structure of the cloud storage system.

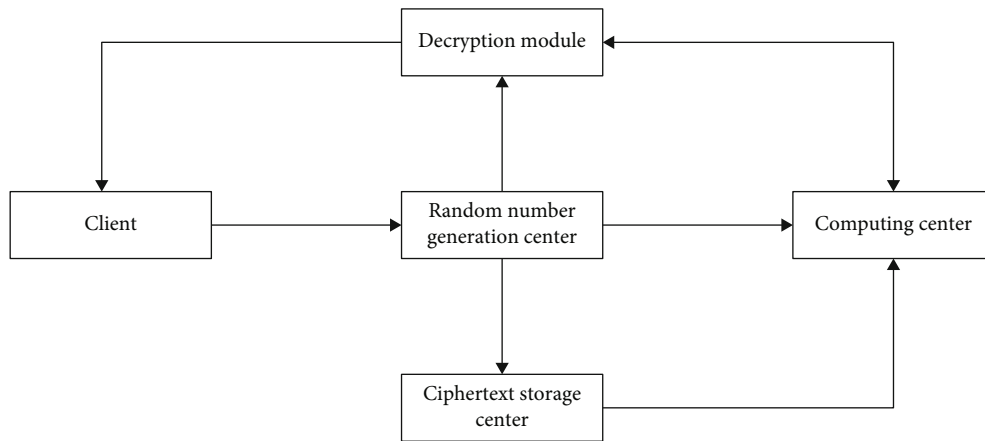


FIGURE 6: Overall structure of the CSS.

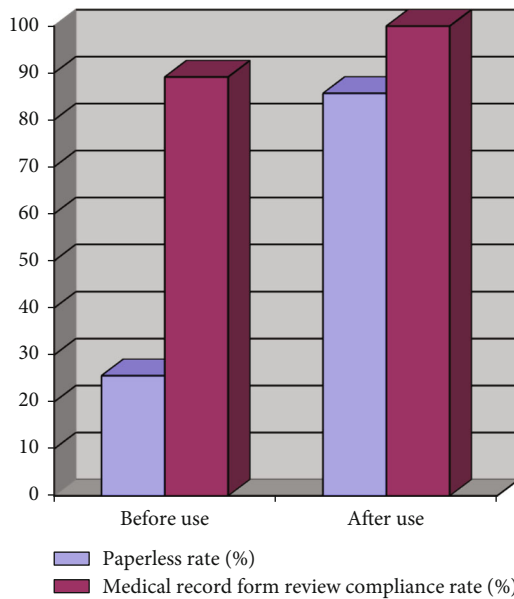


FIGURE 7: Comparison of medical record data before and after the use of paperless electronic medical record management system.

### 4. Simulation Experiment

4.1. *Simulation Environment.* The Internet of Things technology is tested on the PC host. The CPU of the host is 2.4GHz and stored in 1G memory. Set the number of system encryption of medical records to 200 and analyze 6 types of main encryption in the early stage, such as buffer overflow

and restriction conditions. Some false encrypted information is added to the medical record source code to detect the algorithm.

4.2. *Data.* The data of a hospital’s paperless electronic medical record management system in the past two years were selected to compare the paperless rate, medical record review rate, and electronic medical record query times before and after the system was used.

4.3. *Statistical Methods.* SPSS25.0 software was used for statistical analysis of the collected data, and the rate of counting data (%) was expressed by  $X^2$  test.  $P < 0.05$  was considered statistically significant.

4.4. *Application Results.* The original algorithm is applicable when the server is trusted, because when the user requests the server to retrieve the keyword, the user must send the encryption key  $p$  to the server. In this way, the ciphertext data stored by the user will be completely transparent to the server. If the algorithm is applied to the cloud storage system with untrusted server, the information stored on the server is likely to be leaked, and the security of the information cannot be guaranteed.

The cloud storage system adopts a typical client-server model. The overall structure is shown in Figure 6.

As can be seen from the structure diagram in Figure 6, the storage of the system is mainly divided into client terminal, random number generation, ciphertext storage, decryption module, and computing center. Each module is divided into independent parts with safe links.

TABLE 1: Comparison of encryption algorithms based on the Internet of Things and traditional methods.

Encryption method	Check out the actual encryption	Accuracy	Encryption time
RATS	174	86.4%	3'55"
Internet of Things	192	97.5%	2'17"

TABLE 2: Encryption verification statistics of the algorithm for 200 encryptions.

Threshold	Encrypted quantity checked out	False positive	Underreport	Accuracy
1	187	3	13	98.5%
0.94	196	5	8	97.5%
0.84	216	23	14	89.2%
0.74	215	46	28	78.4%
0.62	257	92	36	64.2%

Client defines the polynomial function  $F$  for ciphertext operation and sends it to the random number generation center for obtaining statistical information about personal data.

Random number generation center: input function  $F$  outputs random numbers and sends them to the decryption module, ciphertext storage center, and computing center, respectively, through steps (1), (2), and (3).

Ciphertext storage center protects the privacy of data, encrypts the plaintext, and then transfers the encrypted data to the computing center. Without the decryption key, the raw data cannot be accessed.

Decryption module owns the decryption key of the system.

Computing center receives ciphertext, polynomial functions, and random numbers transmitted by other modules and calculates addition and multiplication of the encrypted ciphertext.

The paperless electronic medical record management system received 160,450 electronic medical records during its two years of online use, and the paperless rate reached 86.77%. The paper cost of all kinds of medical records decreased by 14.2%, and the rate of medical record form examination reached 100% from the original 89%. In two years, the visits of medical workers and patients to query resources through the client increased from 26045 times to 150812 times. The data analysis results showed that the paperless rate, the standard review rate of medical record form, and the number of electronic medical record inquiries after the use of the paperless electronic medical record management system increased significantly compared with the use before, and the differences were statistically significant ( $\chi^2 = 13.22$ ,  $\chi^2 = 9.41$ ,  $\chi^2 = 39.63$ ;  $P < 0.05$ ).

As can be seen from the results in Figure 7, paperless medical records have a small storage space, which is convenient for doctors and patients to query and browse, proving the effectiveness of the encrypted storage of medical records based on the Internet of things. In addition, the paperless

medical records are more secure and reliable, which can ensure the effectiveness, safety, and accuracy of medical records, improve the efficiency of relevant staff, further reduce the burden of medical workers, and improve the real-time storage and safety management of medical records.

It can be seen from Table 1 that compared with the traditional encryption method based on rule matching, the medical record encryption storage algorithm based on the Internet of Things has higher accuracy (97.4%), while the encryption time is shortened by 42.3%, which has the advantages of fast and efficient. According to the experimental contrast of the five thresholds in Table 2, 0.95 is used as the similar matching threshold. The algorithm has better balance, high accuracy, false alarm rate  $\leq 26\%$ , report leakage rate  $\leq 4.5\%$ , false alarm rate, and report leakage rate All remain at a low level. In addition, the threshold can also be adjusted appropriately according to the specific encryption request to meet different encryption requirements to realize the encryption object.

## 5. Conclusions

With the continuous development and application of Internet of Things technology, individuals pay more and more attention to the management of medical information. In view of this demand and limitation, this paper introduces the Internet of Things technology, analyzes and constructs the medical record encryption management system by sorting out the business process of patients, doctors, and nurses, makes a comparative analysis before and after the system goes online, and conducts simulation experiment research. The simulation results show that the medical record encryption storage system based on the Internet of Things technology can effectively solve the collection, statistics, and integration of patients' medical record information, form an electronic medical record management system with unified data format, data sharing, and device interconnection, and realize the whole process of patients' medical record information collection.

## Data Availability

Data sharing are not applicable to this article as no datasets were generated or analyzed during the current study.

## Conflicts of Interest

The authors declare no competing interests.

## Acknowledgments

This study is sponsored by the First Hospital of Jilin University.

## References

- [1] J. Sun, X. Yao, S. Wang, and Y. Wu, "Blockchain-based secure storage and access scheme for electronic medical records in IPFS," *IEEE Access*, vol. 8, no. 2, pp. 59389–59401, 2020.
- [2] A. Tembhare, S. Sibi Chakkaravarthy, D. Sangeetha, V. Vaidehi, and M. Venkata Rathnam, "Role-based policy to

- maintain privacy of patient health records in cloud,” *The Journal of Supercomputing*, vol. 75, no. 9, pp. 5866–5881, 2019.
- [3] Y. Wu, H. Zhou, X. Ma et al., “Using standardised patients to assess the quality of medical records: an application and evidence from rural China,” *Quality & Safety in Health Care*, vol. 29, no. 6, pp. 491–498, 2020.
- [4] S. Allali, M. de Montalembert, V. Brousse et al., “Hepatobiliary complications in children with sickle cell disease: a retrospective review of medical records from 616 patients,” *Clinical Medicine*, vol. 8, no. 9, pp. 1481–1490, 2019.
- [5] F. L. Mirarchi, K. Juhasz, T. E. Cooney et al., “TRIAD XII: are patients aware of and agree with DNR or POLST orders in their medical records,” *Journal of Patient Safety*, vol. 15, no. 3, pp. 230–237, 2019.
- [6] P. Kong, D. Fu, X. Li, and C. Qin, “Reversible data hiding in encrypted medical DICOM image,” *Multimedia Systems*, vol. 27, no. 3, pp. 303–315, 2021.
- [7] S. Haddad, G. Coatrieux, A. Moreau-Gaudry, and M. Cozic, “Joint watermarking-encryption-JPEG-LS for medical image reliability control in encrypted and compressed domains,” *IEEE Transactions on Information Forensics and Security*, vol. 15, no. 3, pp. 2556–2569, 2020.
- [8] S. Kumar, A. K. Bharti, and R. Amin, “Decentralized secure storage of medical records using Blockchain andIPFS: a comparative analysis with future directions,” *Security and Privacy*, vol. 4, no. 5, pp. 1–8, 2021.
- [9] Y. Gon, K. Yamamoto, and H. Mochizuki, “The accuracy of diagnostic codes in electronic medical records in Japan,” *Journal of Medical Systems*, vol. 43, no. 10, pp. 1–7, 2019.
- [10] J. Chong, T. Jason, M. Jones, and D. Larsen, “A model to measure self-assessed proficiency in electronic medical records: validation using maturity survey data from Canadian community-based physicians,” *International Journal of Medical Informatics*, vol. 141, no. 2, p. 104218, 2020.
- [11] I. Huvila, Å. Cajander, M. Daniels, and R. M. Åhlfeldt, “Patients’ perceptions of their medical records from different subject positions [J],” *Journal of the Association for Information Science and Technology*, vol. 4, no. 1, pp. 1–10, 2019.
- [12] S. Liu, W. Nie, D. Gao, H. Yang, J. Yan, and T. Hao, “Clinical quantitative information recognition and entity-quantity association from Chinese electronic medical records,” *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 1, pp. 117–130, 2021.
- [13] G. T. Woods, K. Cross, B. C. Williams, and T. N. A. Winkelman, “Accessing prison medical records in the United States: a national analysis, 2018,” *Journal of General Internal Medicine*, vol. 34, no. 11, pp. 2331–2332, 2019.
- [14] J. Dong, J. Li, and G. Zhen, “A robust watermarking algorithm for medical images in the encrypted domain [J],” *Springer, Cham*, vol. 4, no. 1, pp. 1–10, 2016.
- [15] R. A. Montaez-Valverde, J. J. Montenegro-Idrogo, and R. Vásquez-Alva, “Missing information in medical records: beyond the quality of registration [J],” *Revista Médica de Chile*, vol. 143, no. 6, pp. 812–820, 2015.
- [16] A. Mfp, A. Amb, and A. Baw, “Permutation modification of reversible data hiding using difference histogram shifting in encrypted medical image [J],” *Procedia Computer Science*, vol. 135, no. 5, pp. 727–735, 2018.
- [17] Y. Miao, J. Ma, X. Liu, F. Wei, Z. Liu, and X. A. Wang, “m2-ABKS: attribute-based multi-keyword search over encrypted personal health records in multi-owner setting,” *Journal of Medical Systems*, vol. 40, no. 11, pp. 109–117, 2016.
- [18] K. Amer, “Informatics: ethical use of genomic information and electronic medical records,” *Online Journal of Issues in Nursing*, vol. 20, no. 2, pp. 564–570, 2015.
- [19] C. S. Brixval, L. Thygesen, N. Johansen et al., “Validity of a hospital-based obstetric register using medical records as reference,” *Clinical Epidemiology*, vol. 7, no. 5, pp. 1–9, 2015.