IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 15, 2021, accepted March 28, 2021, date of publication March 31, 2021, date of current version April 9, 2021.
Digital Object Identifier 10.1109/ACCESS.2021.3069992

Efficient Edge Cache Collaboration Transmission Strategy of Opportunistic Social Network in Trusted Community

XIAOMIN WU, LIU CHANG, JINGWEN LUO, AND JIA WU, (Member, IEEE)
School of Computer Science and Engineering, Central South University, Changsha 410075, China
Corresponding author: Jia Wu (jiawu0510@163.com)

This work was supported by the Hunan Provincial Natural Science Foundation of China under Grant 2018JJ3299 and Grant 2018JJ3682.

ABSTRACT With the continuous development of 5G network technology, the amount of data has surged. Network data transmission technology needs improvement. To ensure the reliable transmission of data, reduce network congestion and network energy consumption, the paper proposed an effective edge cache collaboration routing strategy of opportunistic social networks in the trust community, namely the edge collaboration cache trust community routing algorithm (ECTC). The algorithm proposes that the nodes are used to carry information through the cache. Based on the node trust degree, the efficient trust relay node is identified. Build a trusted network based on the trust degree of nodes. Divide local trust community modules into clusters to realize efficient community retrieval. Under the concept of edge nodes, trust nodes are found in the trust community. Edge caches cooperate to share transmitted data, reducing backbone network congestion. The data update efficiency is greatly improved. Through edge nodes in trust community cooperation and communication, efficient data packet delivery is realized, thereby improving the community trust network's routing performance. The simulation experiment results show that the edge collaboration cache trust community routing algorithm (ECTC) can significantly improve delivery effectiveness and reduce network overhead.

INDEX TERMS Community routing, trust community, edge node, cache collaboration.

I. INTRODUCTION

With the continuous development of 5G network technology, mobile network equipment has proliferated in recent years. In the future, the 5G network technology industry will gradually develop to improve mobile communications quality, reduce operating costs [1] and provide a foundation for large-scale data transmission. 5G derivatives, such as mobile devices (mobile phones, tablets, and other human-computer interaction devices), sensing devices (applications for transmission and collection), vehicle networks (autonomous driving, vehicle equipment) [2]–[4] continue to spread. Various wireless communication networks have penetrated our daily lives, such as handheld device networks and communication networks in remote areas. The continuous expansion of the application scale of 5G derivatives and wireless data transmission networks, such as knowledge networks, protein molecules, military communications [5], medical fields [6] field tracking [7], and disaster rescue [8] have increased the demand for data transmission.

In the context of human activities, complex interpersonal relationships will change the evolution direction of social groups. Due to human social behavior and activities' characteristics, the device has specific community attributes and dynamic features, thereby forming an opportunistic social network [9], [10]. Opportunistic network nodes have typical mobility, openness, and sparseness [11]. It selects neighboring mobile devices dynamically as neighbor nodes to construct a communication path based on the "storage-save-forward" mechanism [12]. The widespread use of opportunistic networks and the rapid increase in information requires us to pay attention to the control of Fraud Information [13], [14] and the spread of negative news [15].

At the same time, further research on opportunistic network data caching and transmission is needed. The "storage-save-forward" mechanism in opportunistic networks makes waiting for information stored in the device. It makes there is long-term storage, but there is no user acceptance and

The associate editor coordinating the review of this manuscript and approving it for publication was Wenjie Feng.

response problem. As a result, the node cache space is insufficient, and it is difficult for the node to update data. Considered the mobility and sociality of the communication network user, data transmission has the characteristics of small data packets and high connection volume. The massive amount of data makes the data transmission volume of the communication node become very large. The node calculation task is complicated. What is worse is the use of flooding technology for data transmission [16], and multi-copy transmission strategy. It is easy to cause node death, communication network blockage, and data transmission delay [17]. There will also be a problem of packet loss, which significantly limits network performance. Edge computing, a new computing paradigm that pushes computing and storage resources to the edge of cloud computing [18], [19]. Also, in the network information, the edge node cache information is used. These nodes are close to the central information node, and through collaboration, the pressure on the central node can be relieved. Simultaneously, in remote areas or at the edge of the device network, it is often impossible to access the Internet. So, the communication connection is unstable and the data delivery rate is low in these situations [20]. In addition, the pervasive edge of the global communication cost is high, but the effectiveness is low [21].

Users pay attention to obtaining information, hoping to obtain data in a short time. To achieve comprehensive and stable communication and support forward-looking decision-making. Edge collaboration community opportunity routing communication can effectively improve data transmission efficiency and reduce transmission energy consumption. To this end, we choose community communication methods and apply efficient transmission paths and edge collaboration cache routing strategies to optimize message delivery efficiency and alleviate traffic congestion on the backbone network. Thereby, the communication network's data transmission efficiency is further improved, and central node storage pressure is relieved. The final network service quality and user experience quality are improved [21].

In response to the problems mentioned above, this paper proposes a complex social network trust routing algorithm. The contributions of this research are as follows:

(1) Propose the concept of edge nodes based on data transmission characteristics, define the measurement method of node trust, and further propose clustering to form a local trust module community. Divide the community based on the trust relationship, obtain the community module with high structural strength accurately and clearly, and realize efficient community retrieval.

(2) Join the trust community relationship and propose an edge node cache cooperative transmission routing algorithm based on community division. Make information transmission more in the community, improve delivery efficiency, relieve communication node congestion, and reduce network overhead.

(3) The node data transmission network's simulation experiment proves that the edge cache collaboration trust community routing algorithm can significantly improve delivery efficiency and reduce network overhead.

The rest of the paper is arranged as follows: Section II reviews the latest research work and opportunistic network results. Section III of this paper proposes edge cache collaboration trust community routing algorithm (ECTC). In Section IV, we conducted simulation comparison experiments of 5 routing algorithms ECTC, EMPIDEMIC, ICTM, BSW, ETNS to evaluate the algorithms' performance. Section V summarizes the full text and expounds on potential research directions in the future.

II. OVERVIEW OF COMMUNITY OPPORTUNISTIC NETWORKS

At present, academic circles at home and abroad have many research methods on opportunistic community networks, which mainly focusing on community division and routing strategies. The proposed different algorithms can effectively improve the existing network communication and data transmission. The following will focus on the existing opportunistic network community division and routing strategy algorithm.

Nodes in the same community are relatively similar, and nodes communicate more within the community than between them [22]. Community trust consensus communication can effectively reduce the search space of information transmission and improve transmission efficiency [23]. Comprehensive trust evaluation is better than a single trust [24]. The literature [25] proposed node similarity as an implicit trust index, which improves prediction and recommendation effectively, but time information is not considered. The research [26] proposed a time series to predict the future similarity and used the network changes over time to better complete the prediction. Node similarity link prediction is useful in community division, such as fuzzy communities [27], which provides new ideas for community division.

In addition to the trust relationship between nodes, community trust relationship should also be considered. Starting from the trust relationship, integrating edge fitness and community fitness detection in non-overlapping communities [27], [28]. However, it is not suitable for existing social networks with considerable overlap. Users are clustered in the research [29] before trusting, so that trust can be spread in the user-specific community. The redundant transmission of information is reduced. The research [30] reorganized the community according to the degree of relevance of the node, and the new community assists the source node in transmitting information. The literature [31] proposed maximum and minimum modularity, uses hierarchical clustering to fuse connectivity rules, mines, detects communities, realizing community mining and division efficiently and straightforward.

Flood routing [16] sends a message to every node encountered, resulting in multiple copies, quickly causing node death, data packet loss, and reducing the delivery rate. It can be used based on social networks, which make certain improvements. The proposal of edge nodes can effectively solve the problem of node congestion. Data caching in the edge computing environment [32]–[35] have focused on how to cache data across edge to achieve different optimization objectives, to minimize caching cost [35], to minimize latency [36], to guarantee the quality of transmissions [33], etc. The research [37] reorganize the community, duplicate filter nodes, and improve the delivery rate of the opportunity network through node contribution evaluation and community reconstruction. SAW [38], [39] broadcast a certain number of copies to the relay node, and then wait until one of them reaches the destination node, and further control the routing and forwarding volume by receiving feedback. Gao *et al.* proposed a novel scheme to support cooperative caching in social opportunistic networks. Wu *et al.* [40] propose a hierarchical cooperative caching scheme in mobile opportunistic social networks. In terms of cache cooperative data transmission, it performs well to relieve node congestion and speed up data updates. The literature [17] proposed an adaptive spray waiting for routing algorithm based on node quality QON. The relationship between nodes allocates the number of message copies, avoiding the blindness of forwarding relay selection, but still only considers relay nodes Quality, ignore the connection of source node, relay node, and destination node.

Based on the research of the above methods and problems, we propose a more effective method to solve data transmission during the edge collaboration cache trust community in complex networks.

III. TRUST COMMUNITY ROUTING ALGORITHM

In this section, the research will introduce the proposed the edge collaboration cache trust community in detail. The algorithm includes three main steps: (1) node trust measurement; (2) trust community division; (3) Cooperative cache routing. Specifically, first, the node used in this study carries information through the cache. It is proposed to directly mark explicit factors and predict that possible future links will be implicitly trusted, thereby gaining complete node trust. Then build a trusted network based on the node trust, calculate the module's trust value, and divide the trust community through clustering. Finally, under the concept of edge nodes, trust nodes are found in trust community, and edge caches cooperate to share the transmitted data. Network congestion is reduced. Data update efficiency is significantly improved. That is an Efficient cooperative transmission strategy based on the cache of the edge in trust community opportunistic social network.

Computing node trust is the basis for the division of trust communities. Information dissemination is more spread in trusting communities. Choosing a collaborative cache at the edge of a trusted community for data transmission can further reduce network congestion and speed up information dissemination. Node trust measurement, trust community division, trust cooperative cache three algorithms are progressively coordinated to achieve efficient data transmission.

A. NODE TRUST DEGREE MEASUREMENT

Information dissemination under the condition of an unclear path will consume much time, causing node death and network congestion. Calculating with node trust degree and constructing a node trust network is the basis for efficient information transmission. For this reason, this paper proposes to use link prediction to quantify the calculation index of node trust to obtain the user information transmission path. In data transmission, the source node information cannot directly reach the destination node. The trusted relay node is selected to assist in the transmission of information. The cache space of trusted nodes is limited. The article proposes a collaborative trust cache of edge nodes. As shown in FIGURE 1. Node trust *trust* comprehensively considers the explicit and implicit direct trust of nodes. Explicit trust s can be directly marked while implicit trust means that the link may be obtained in the future.

Definition 1 Node Trust:

This project defines the relationship between the possibility of information transfer between a node and its neighbors. For the graph $G(V, E)$, V is the vertex set, and E is the edge set. The trust between nodes is shown in formula (1).

$$\begin{cases} trust^{+}_{(u,v)} = \alpha s + (1-\alpha) sim(u, v) \\ trust^{-}_{(u,v)} = sim(u, v) \end{cases} \tag{1}$$

among them, $trust^{+}_{(u,v)}$ represents the comprehensive trust when there is explicit trust between nodes, s is the direct label of the trusted node. If there is explicit trust $s = 1$, α is the weighting factor. $trust^{-}_{(u,v)}$ means that there is only implicit trust between nodes.

Definition 2 Node Similarity:

In the study, the local similarity index AA and the global index LP is comprehensively considered to find. The user's comprehensive trust index is calculated. Considering all nodes with fixed communication cost, the proposed similarity index $sim(u, v)$ is defined as follows:

$$sim(u, v) = \sum_{\substack{z \in V \setminus \{u,v\} \\ 1<d=\delta(u,z)+\delta(v,z)<L}} \frac{\beta^{d-2}}{\log|\Gamma(z)|} \tag{2}$$

among them, β is the damping factor for the distribution of weights, so that the weight decreases with the increase of the number of hops. d represents the shortest path between (u, v), L is the longest reachable distance.

Through formula (2) to get node $sim_{(1,2)}$ and node 1 to 2 directly trust, s = 1, we can get $trust_{(1,2)}$.

According to the above definition, the trust value between node (u, v) is related to s and $sim_{(u,v)}$. The larger the value, the more likely the node will transmit data. Algorithm 1 details the process of calculating the path between nodes.

The graph $G(V, E)$ takes the adjacency matrix as input and also accepts the parameter hop count attenuation coefficient and (the longest path) as the threshold.

Step1: Calculate the shortest path d between all pairs of nodes in the network.

$$d = \delta(u, z) + \delta(v, z) \tag{3}$$

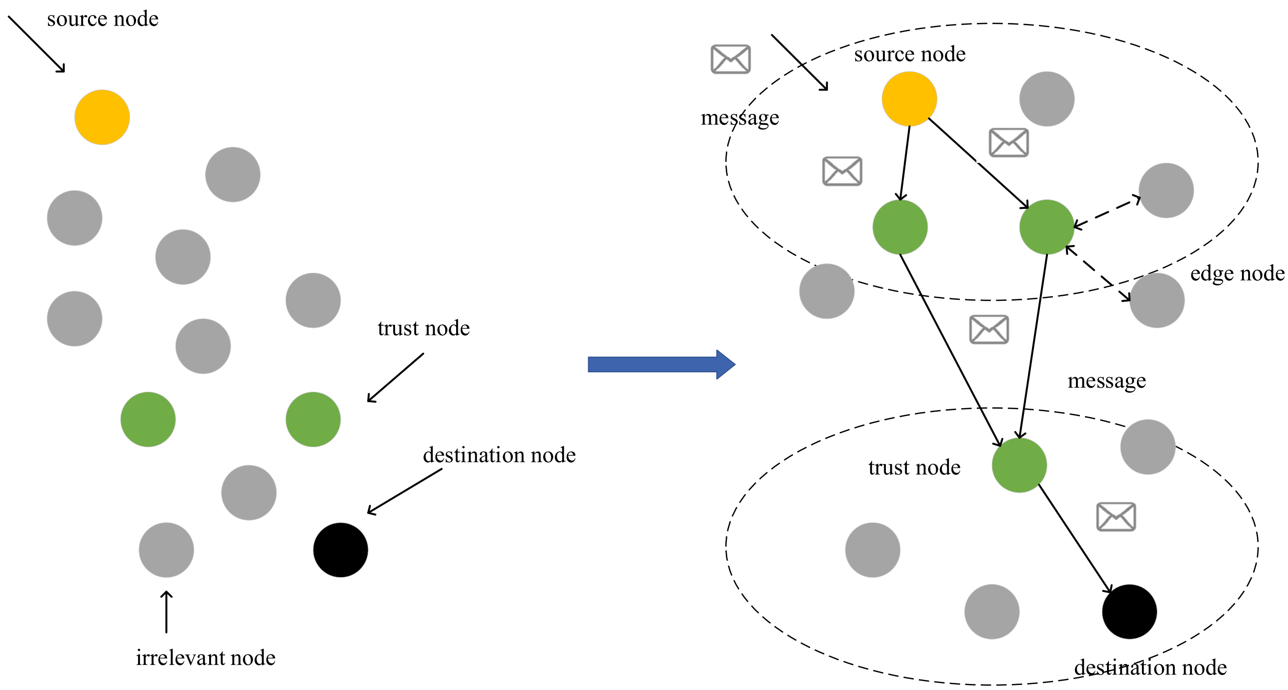

FIGURE 1. Information transmission process based on trust node and edge node.

TABLE 1. Algorithm node relay.

Algorithm 1: $node_{rel}$		
1.	Input	Network G (V, E), the value L and damping parameters B, S
2.	Output	similarity score trust M
3.	Trust $\leftarrow$ 0	
4.	$\sigma \leftarrow$ SHORTPATHS (G)	
5.	for $(u,v) \in E \setminus E^T$ d	
6.	for $z \in V \setminus \{u,v\}$ d	
7.	d= $\sigma(u,z)+\sigma(v,z)$	
8.	if $d \le L$ then	
9.	$sim(u,v) \leftarrow sim(u,v)+\beta^{d-2}(\log\|z\|)^{-1}$	
10.	end if	
11.	end for	
12.	if s=0	
13.	$trust(u,v) \leftarrow sim(u,v)$	
14.	Else	
15.	$trust(u,v) \leftarrow \alpha s_{(u,v)}+(1-\alpha)sim(u,v)$	
16.	end if	
17.	TRUST.NORMALIZON	
18.	return M	
19.	end for	

Step2: Traverse all pairs of unconnected nodes. Consider the paths of all nodes in the network (only consider the paths with connection value less than L), and calculate the similarity score.

Step3: Finally, add direct trust, and normalize to get the total node trust value *trust*.

See TABLE 1 for specific steps.

B. NODE COMMUNITY DIVISION

The community is further divided based on node trust so that more information is transmitted within the community, which can reduce transmission energy consumption to a certain extent. The trust degree clustering method is further used to divide the overlapping community network, and the partial module trust degree is used as the basis for division.

$$Trust_c = \frac{1}{m_c}\sum_{i,j\in c} trust_{(i,j)} \tag{4}$$

among them $Trust_c$ is the module trust degree, m is the number of module nodes, and (i,j) is the current measured node pair.

A vast community will weaken the effectiveness of community division. Set the module size threshold M the subordinate value b. The range of the number of communities is $[1, n]$, where n is the maximum number of communities that the set user can have $b < n$.

Step1: From the community network to be divided, a virtual node with a larger node degree $k(i)$ is selected, the initial core community [41]. Starting from the core node, use formula (3) to calculate the node trust. Calculate the trust degree of the module when there is a core node, $m = 1$.

$$Trust_c = \frac{1}{m}\sum_{m}^{M}[\alpha s_{(i,j)} + (1-\alpha) \times \sum_{\substack{i,j\in m, z\in V\setminus\{i,j\}\\ 1<d=\delta(i,z)+\delta(j,z)<L}} \frac{\beta^{d-2}}{\log|\Gamma(z)|}] \tag{5}$$

Step2: In the current community's reachable nodes, choose to add a node. Increase the clustering module for this node's

TABLE 2. Algorithm community relay.

	Algorithm 2 $Community_{relay}$
1.	Input Network $G(V, E)$ *trust* matrix, the value of *M, n*
2.	Output Community $C=(c_1, c_2, c_3 ......c_n)$
3.	$C \leftarrow \phi$
4.	for each $i \in G$ do
5.	get $k(i)$
6.	end for
7.	for $V \neq Null$ do
8.	C $\leftarrow i$, m $\leftarrow$ 1
9.	$Trust_c$
10.	for $(i, j) \in E \setminus E^T, m < M$ do
11.	$b_i \leftarrow 0$, $P \leftarrow \phi$, $i_b \leftarrow 0$
12.	for $b<n$ do
13.	$P \leftarrow j$
14.	$Trust_c' \leftarrow Trust_c$
15.	if $Trust_c' > Trust_c$ then
16.	$V \leftarrow V - j$, $C \leftarrow C + j$,b $\leftarrow$ b+1
17.	$P \leftarrow \phi$
18.	else
19.	goto label 12
20.	end if
21.	end for
22.	// all nodes are assigned to the community
23.	end for
24.	return Community $C=(c_1, c_2, c_3 ......c_n)$
25.	end for

community affiliation value.

$$Trust_c' = \frac{1}{m+1} \sum_{m+1}^{M} [\alpha s_{(i,j)} + (1-\alpha) \times \sum_{\substack{i,j \in m, z \in V \setminus \{i,j\} \\ 1<d=\delta(i,z)+\delta(j,z)<L}} \frac{\beta^{d-2}}{\log|\Gamma(z)|}] \quad (6)$$

Step3: Determine the degree of trust in the local module. When the local modularity starts to decrease, $Trust_c' < Trust_c$, stop clustering. The division of the community c is completed. Return to Step 1, and start to divide the next community $c + 1$.

When the local trust modularity increases, $Trust_c' > Trust_c$, return to Step2 to find the next node. Until the local modularity value $Trust_c^x$ of the executed x word becomes smaller, return to Step2.

Step4: When the local module exceeds the huge community threshold M or all node V in the network, stops the algorithm.

FIGURE 2 shows clustering $community_1$, $community_2$, $community_3$ represents the three communities after clustering.

C. EDGE COOPERATIVE CACHE TRANSMISSION

Nodes in the community have high trust, stable information transmission and high frequency. In the transmission process, with the trusted node's high transfer rate, the node's cache may overflow due to excessive information. The traditional caching strategy releases the cache space by directly deleting the message, which is easy to cause the information delivery ratio drop. In the research, we proposed the edge cooperative cache trust transmission to reduce buffer overflow. As shown in the FIGURE 2, when the nodes in the cache of the trusted relay node are complete, the neighbor nodes (edge nodes) that have excess cache space should be transmitted.

There are two communities $community_1$, $community_2$ in the community network. Nodes will move between community communication networks to transfer information. When the trusted node cache is full, the edge node cache will delete or schedule the information. and the According to the current network conditions, the messages' importance in the node buffer space is distinguished. The appropriate messages are selected for priority storage to avoid network congestion.

The article proposes a trusted network cache cooperative transmission strategy when nodes meet. This strategy adjusts the order of messages according to the priority of message forwarding. If the buffer is saturated during the forwarding process, the strategy can discard the message queue to improve efficiency. Simultaneously, the strategy will be that the node periodically deletes expired messages to ensure that it has enough cache space. FIGURE 3 shows the routing process of information. The routing algorithm between communities is further described below.

When two nodes meet, messages are sent in order of the practical level. Priority forwarding of messages with a small percentage of cache. The trust value *trust* with the destination node reflects the node's ability to deliver messages to the destination node. The higher the node's intimacy is, the more chance the message will be delivered to the destination node.

Definition 3 Messagetr Life Ratio:

The message's life ratio is the ratio of the existence time of the message to the total lifetime of the message. The greater the life-to-life ratio of a message, the closer it is to die. It needs to be forwarded as soon as possible.

$$TR = \frac{T_{\text{current}} - T_{\text{create}}}{TTL} \quad (7)$$

Definition 4 Message Cache:

The message cache space ratio to the total cache space of the node. The smaller the proportion of cached messages, the smaller the time required for forwarding.

$$MR = \frac{m}{M_t} \quad (8)$$

Definition 5 Message Priority:

The utility function of message priority is:

$$TL_m = \alpha trust + \beta TR + \gamma(1 - MR) \quad (9)$$

Among them, $\alpha + \beta + \gamma = 1$.

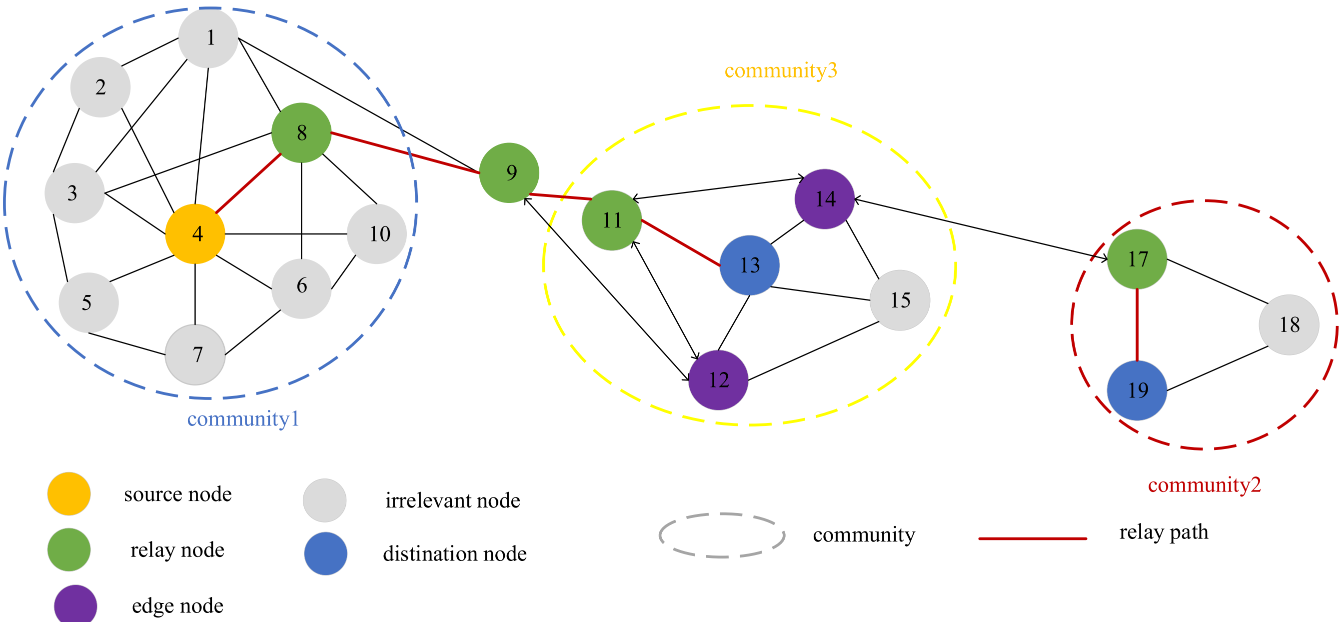

FIGURE 2. **Community module trust cluster division and edge cache.**

Based on the above method, we propose a method for collaborative node caches in trusted social networks. The process is as follows:

Step 1: In each simulation period T, the nodes in the network check their own message list ML and delete expired message m.

Step 2: If the destination node D is directly connected to the source node S, it will be passed directly to exchange each other's message lists. That is, i, j nodes directly exchange information.

Step 3: If the destination node D and the source node S are not directly connected, and the trusted node has enough buffer space. Then the trusted node path is selected for transmission. Choose suitable marginal neighbors in the trusted community.

Step 4: If the edge node has enough buffer space, accept all the SL messages of i. If space is saturated, the messages are sorted. According to the messages' importance, the messages with the SL priority more significant than the ML threshold are received first.

Step 5: If the trusted node's cache space is full, i preferentially copy important message information to the edge node ML and deletes the information m in the trusted communication node.

Step 6: If the node does not find a suitable cooperative node within the communication range, sort the cached messages. Then, according to the importance of the message priority, the node with the low level of essential messages in the local buffer is deleted.

Step 7: After all node messages are transmitted, the connection is disconnected.

See TABLE 3 for specific algorithms.

TABLE 3. **Algorithm community routing with edge cache collaboration.**

```
Algorithm 3 edge cooperative cache transmission
input: source node S , destination node D , node list V
output: ML
if S , D are in common community
for each node in V
  for message m in ML
    if TTL = 0
      delete m
    end for
  end for
end for
node i meet j return H.forwardlist
i , j exchange ML
  for message m in ML
    if m exists in ML
      delete m
    end if
  end for
i send SL to j and receive SL from i
sort ML and SL with TL
insertion sort SL to ML
interrupt connection
```

This section defines node trust and edge node as first, and explicit trust is added based on considering local and global similarities. Further clustering is based on the node trust degree and the local modules with high trust degrees are divided into the same community. Finally, under the concept of edge nodes, trust nodes are found in Trust Community. Edge caches cooperate to share the transmitted data. Network congestion is reduced. Data update efficiency is significantly improved. That is an Efficient cooperative transmission

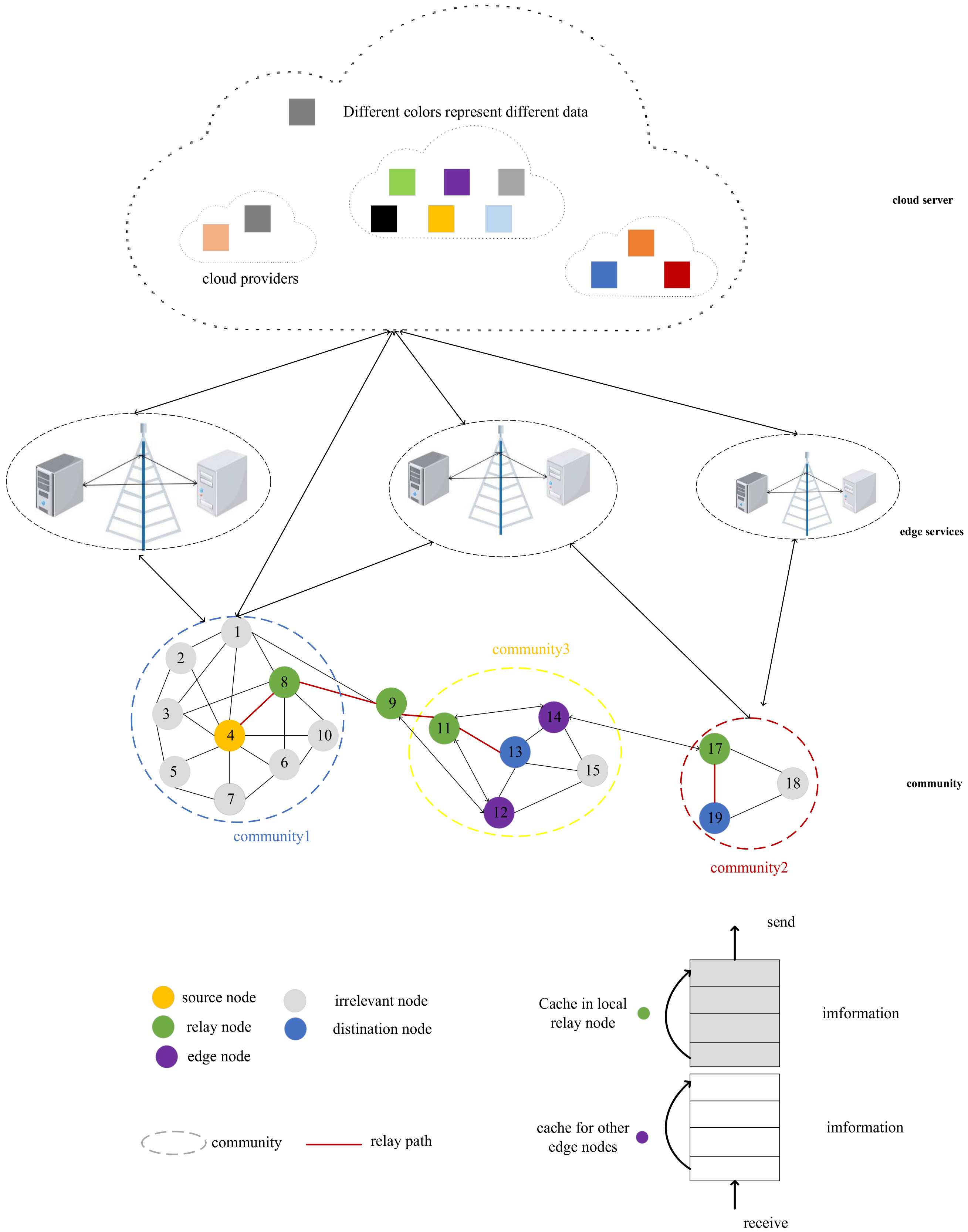

FIGURE 3. Community routing with edge collaboration community routing with edge collaboration.

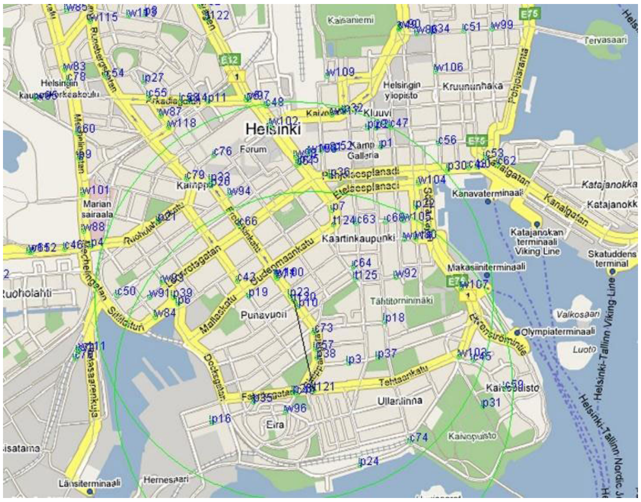

FIGURE 4. Simulation map in Helsinki.

TABLE 4. Simulation parameters.

simulation parameters	parameter range
simulation time	1 h - 6 h
node motion model	SHORESTPATHMAPBASEDMOVEMENT
simulation area	4500 m × 3400 m
nodes numbers	1000
movement speed	0.8 m/s - 1.8 m/s
message categories	12
maximum transmission range	10 m
packet sending interval	25-35 s
packet sending Speed	250 kb /s
message size	200 k-1M
node initial energy	100 J

strategy based on the cache of the edge in trust community opportunistic social network. In the following, the real map in Helsinki will be selected to conduct simulation experiments on the above edge cooperative cache trust community routing strategy.

IV. TCR ALGORITHM PERFORMANCE ANALYSIS RESULTS

This paper uses simulation THE ONE SIMULATOR experiments on the algorithm to evaluate the edge cooperative cache trust community routing (ECTC) understudy from three aspects: delivery rate, transmission delay, and network overhead. In this research, traditional routing algorithms are selected: EPIDEMIC (epidemic algorithm), BSW (two-hop spray waiting for routing algorithm), ICMT (information cache management and data transmission algorithm), and routing algorithm suitable for communities ETNS(community reconstruction routing algorithm). Compare and analyze with (ECTC)the edge cooperative cache trust community routing. Considering the data information required by the algorithm, select an open street map to edit city maps. Different parks, streets, and shops are established in the map. They can exhibit a real environment. As shown in FIGURE 4. It adopts real map in Helsinki, which suitable for node trust information transmission activities.

The experimental environment is i7-7700K 3.5-GHz 8 (CPU), TITAN 12GB (GPU), 16GB × 4 memory (DDR4), and Ubuntu 16.04 (OS). The simulation parameters set in the experiment are summarized in TABLE 4. The simulation time is 1 h to 6 h, and the simulation area is 4500 m × 3400 m in the map. The involved nodes are 1000. The transmission pattern is broadcast. The node motion model is SHORESTPATHMAPBASEDMOVEMENT. The movement speed is 0.8 - 1.8 m/s. The message categories are 12. The maximum transmission range 10 m. The packet sending interval range from 25 to 35 s. The packet sending Speed is 250 kb /s. The message size is 200 k - 1M. The node initial energy is 100 J. According to the data collected in the simulation report, the specific correlation analysis between time and three measurement parameters.

In THE ONE SIMILAR simulation experiment focus on the ECTC and EPIDEMIC, BSW, ICTM, ETNS five routing algorithms. For each algorithm, Delivery ratio, Network overhead, Average delay and five algorithms for analysis are compared.

Delivery ratio: It also known as transmission success rate. This parameter refers to the ratio of successfully delivered messages to the total messages sent in the network during transmission.

$$deliveryratio = \frac{message_{del}}{message_{all}} \times 100\% \tag{10}$$

Average delay: This parameter refers to the average time elapsed between when a message is sent to the target node is received. Where is the total number of messages.

$$latency\_avg = \frac{1}{n}\sum_{i=1}^{n} delivered_t \tag{11}$$

Network overhead: This parameter reflects the average overhead of a successful message routing or forwarding between a pair of mobile nodes, that is, the average of energy consumption and storage overhead.

FIGURE 5 shows the relationship between delivery rate and simulation time with ETCT, EPIDEMIC, BSW, ICTM and ETNS. e. The algorithm ECTC and ETNS similar both divide the structure of the community. Each pair of nodes with high similarity in the community can frequently communicate with each other, eliminate inefficient and increase the delivery rate. Algorithms ETCT and ICTM use cache collaboration strategies to reduce node data packet congestion, speed up the update of trusted node information, and increase the delivery ratio. Furthermore, compared to the late delivery ratio, the advantage of ECTC is obvious. The division and cooperative cache effect are good. The post-delivery rate reaches about 68%. The EPIDEMIC algorithm has many

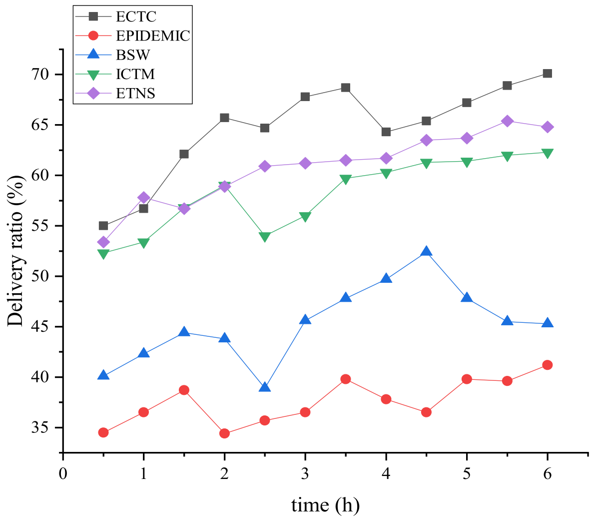

FIGURE 5. **Delivery ratio and simulation time.**

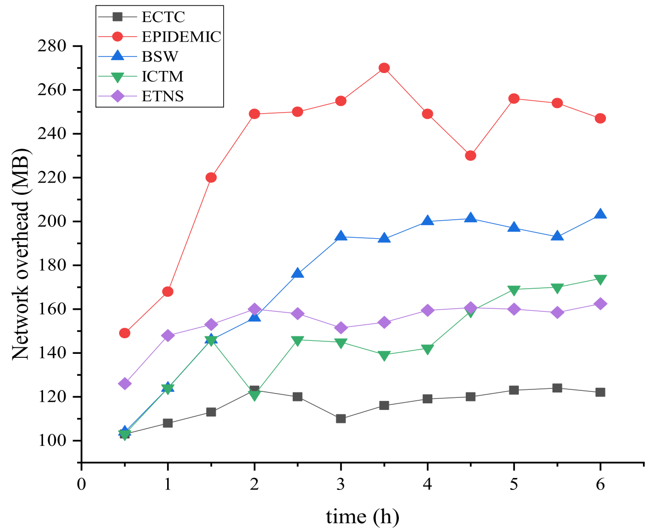

FIGURE 7. **Network overhead and simulation time.**

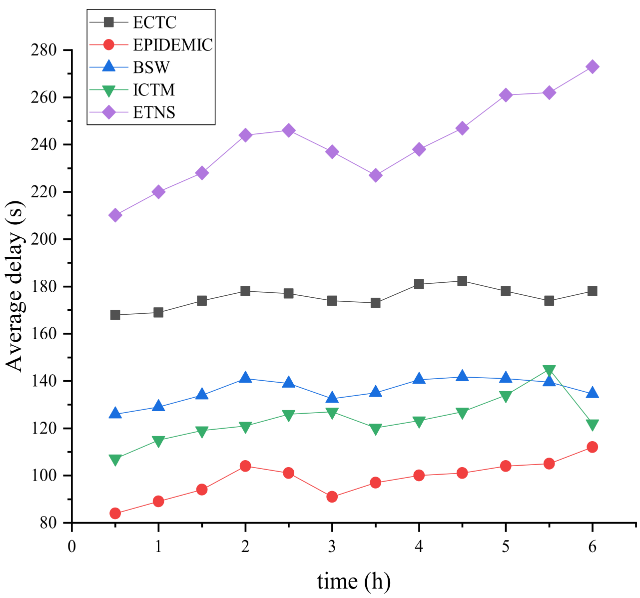

FIGURE 6. **Average delay and simulation time.**

copies of messages. It uses flooding methods will cause a large amount of information to be lost. Therefore, compared with other algorithms, the delivery rate is the lowest, 33%-39%. The number of forwarding relay copies of BSW is limited. The delivery rate is much improved compared to EPIDEMIC.

FIGURE 6 shows the Average delay of EPIDEMIC, BSW, ICTM, ETNS and ETCT routing transmission. The experimental results show that the community's division takes up many network resources, causing delays. Therefore, the transmission delay of ETNS and ECTC are slightly higher than that of EPIDEMIC, BSW, ICTM routing algorithm, about 206-267 s. Since EPIDEMIC and BSW are direct broadcast replicas, they have strong diffusion capabilities and low transmission delay. The ICTM algorithm distributed cache management strategy and node encounter probability effectively reduce the transmission delay, and the average delay is lower than BSW, reaching 104-138 s. The ECTC algorithm effectively combines community division and specific edge node cache cooperation, and the transmission delay is 167-173 s.

It can be seen from FIGURE 7 that the network overhead of EPIDEMIC and BSW routing algorithms have been at a relatively high level. BSW routing algorithm controls the number of relay nodes, which is slightly lower than EPIDEMIC. However, due to the higher algorithm complexity of BSW, the consumption of network storage and computing resources is also relatively high. The ECTC and ETNS algorithms use community optimization methods to eliminate the most unreliable irrelevant nodes and save network resource overhead. ECTC algorithm, except for simulation time at 2 h the average network overhead is below 120 MB. In the later stage of the simulation experiment, the transmission was stable. The ECTC network overhead was more evident than other advantages, only about 43.2% of the EPIDEMIC network overhead.

In social networks, node caching is an important parameter, which affects transmission performance. So the simulation compares cache parameters.

The FIGURE 8 shows the relationship between the delivery ratio and the node cache. In this figure, as the cache increases, the delivery rate of each algorithm increases. The community division can increase the delivery rate. The ECTC, ICTM and ETNS has the higher delivery rate. Especially for ECTC, when the cache is 60MB, the delivery rate is about 84%. The BSW routing algorithm and EPIDEMIC have the worst performance because when data packets are transmitted between nodes, many data packets will be lost in transmission. The Delivery ratio reaches 32%-63%.

The FIGURE 9 shows the overhead and cache. The cache will affect the network overhead in the simulation

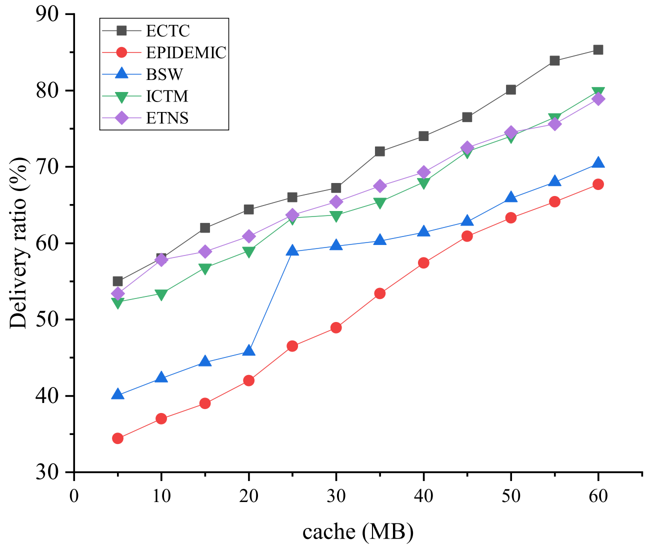

FIGURE 8. Delivery ratio and cache.

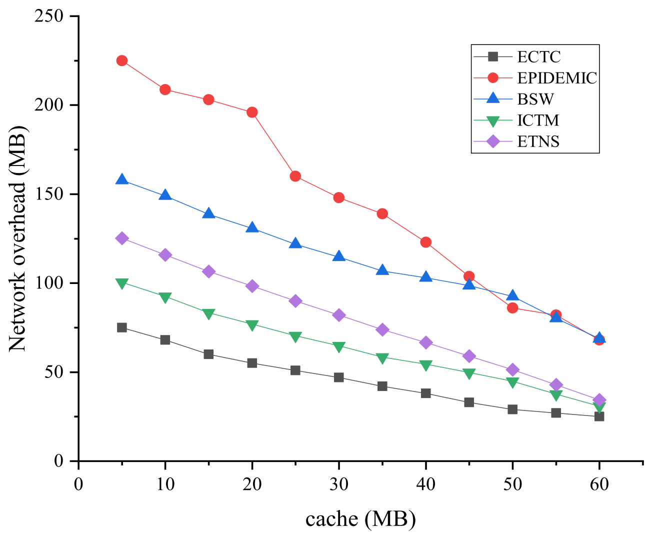

FIGURE 9. Network overhead and cache.

experiment. Increase the cache, and the network overhead will soon decrease. In EPDIMEMIC, the network overhead dropped from 226 to 84 MB. In the BSW algorithm, the overhead dropped from 163 to 83 MB. In the ECTC algorithm, the overhead minimum reached 23 MB. In summary, the ECTC algorithm is better than other algorithms.

FIGURE 10 explains the average overhead and cache. The average delay decreases as the cache increases. As the cache increases, the epidemic algorithm has the lowest latency. ECTC, ICTM delay is kept between 130-50 s. In the ETMS algorithm, the community is divided, and the relay node needs to receive all the copy information, and node congestion may consume the opportunistic network's overhead. The prediction results show that ECTC has better reduced the transmission delay caused by communities' division and node congestion.

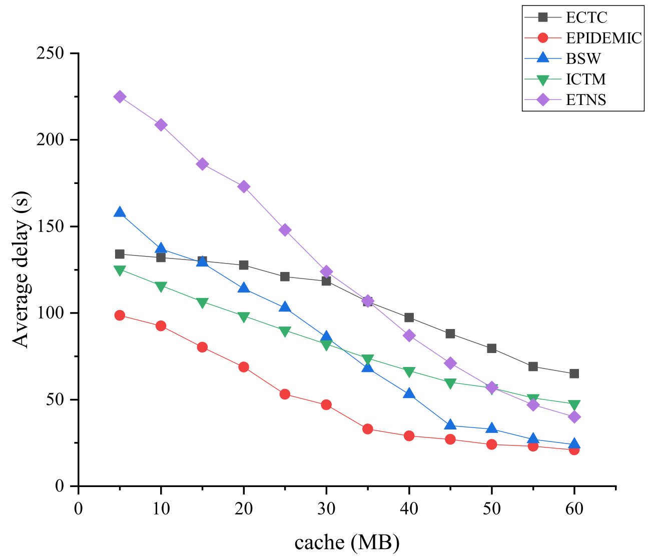

FIGURE 10. Average delay and cache.

In summary, the algorithm ECTC is superior to other routing algorithms in terms of delivery ratio and network overhead. Due to the division of communities, the delay is longer, and the delay overhead is longer than the BSW and EPIDEMIC algorithms. However, edge cache is used so that the transmission delay is much lower than the ETNS algorithm. As the cache increases, the algorithms ECTC, EPIDEMIC, BSW, ICTM, ETMS delivery ratio all increases the network overhead and average delay decrease. Overall. ECTC is superior to other routing algorithms. Combining the three indicators: delivery ratio, network overhead and average delay, within 48 - 93s of time delay, the delivery rate increases about 173% than EPIDEMIC. The network cost is reduced by about 73.6%., which is significantly improved compared to other algorithms. When the node cache is 60 MB, the delivery rate is up to 89.7%. The results show that the ECTC algorithm performs well in improving the information delivery rate and reducing network overhead. It can effectively reduce the redundant replication of messages, alleviate the backbone network's congestion, and improve routing performance.

V. CONCLUSION

This paper designed an effective edge cache collaboration routing strategy of opportunistic social networks in the trust community The Edge Collaborative Cache Trusted Community Routing Algorithm (ECTC). The algorithm proposes that the nodes are used to carry information through the cache. Based on the node trust degree, the efficient trust relay node is identified. Build a trusted network based on the trust degree of nodes. Divide local trust community modules into clusters to realize efficient community retrieval. Under the concept of edge nodes, trust nodes are found in the trust community. Edge caches cooperate to share transmitted data, reducing backbone network congestion. Combining the three indicators: delivery ratio, network overhead and average delay. The simulation results show that The Edge Collaborative Cache Trusted Community Routing Algorithm, within 48 - 93s of time delay, the delivery rate is increased about 173% than EPIDEMIC, and the network cost is reduced by about 73.6%. When the node cache is 60 MB, the delivery rate is up to 89.7%. The ECTC algorithm effectively reduce the redundant replication of messages, alleviate the backbone network's congestion, and improve routing performance.

In the community, the edge collaborative Cache improve the efficiency of data transmission. Reducing energy consumption will be the focus of future work. It is necessary to explore node relationships, reduce network energy consumption, transmission strategies further and further explore routing strategies with low latency, low energy consumption, and high efficiency.

REFERENCES

[1] G. Yu and J. Wu, "Content caching based on mobility prediction and joint user prefetch in mobile edge networks," *Peer Peer Netw. Appl.*, vol. 13, no. 5, pp. 1839–1852, Sep. 2020.

[2] W. Yang, J. Wu, and J. Luo, "Effective data transmission and control based on social communication in social opportunistic complex networks," *Complexity*, vol. 2020, Jun. 2020, Art. no. 3721579, doi: 10.1155/2020/3721579.

[3] X. Zhang, Y. Yuan, Z. Zhou, S. Li, L. Qi, and D. Puthal, "Intrusion detection and prevention in cloud, fog, and Internet of Things," *Hindawi*, vol. 2019, May 2019, Art. no. 4529757.

[4] R.-C. Marin, R.-L. Ciobanu, C. Dobre, C. X. Mavromoustakis, and G. Mastorakis, "A context-aware collaborative model for smartphone energy efficiency over 5G wireless networks," *Comput. Netw.*, vol. 129, pp. 352–362, Dec. 2017.

[5] A. Y. Lokhov, M. Mézard, H. Ohta, and L. Zdeborová, "Inferring the origin of an epidemic with a dynamic message-passing algorithm," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 90, no. 1, Jul. 2014, Art. no. 012801.

[6] J. Wu, L. Chang, and G. Yu, "Effective data decision-making and transmission system based on mobile health for chronic disease management in the elderly," *IEEE Syst. J.*, early access, Sep. 17, 2020, doi: 10.1109/JSYST.2020.3024816.

[7] S. Misra, T. Ojha, and A. Mondal, "Game-theoretic topology controlfor opportunistic localization in sparse underwater sensor networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 5, pp. 990–1003, May 2015.

[8] X. Li and J. Wu, "Node-oriented secure data transmission algorithm based on IoT system in social networks," *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2898–2902, Dec. 2020, doi: 10.1109/LCOMM.2020.3017889.

[9] G. Wang, B. Wang, and Y. Gao, "Dynamic spray and wait routing algorithm with quality of node in delay tolerant network," in *Proc. Int. Conf. Commun. Mobile Comput.*, vol. 3, Apr. 2010, pp. 452–456.

[10] A. Keranen, "Opportunistic network environment simulator," Dept. Commun. Netw., Helsinki Univ. Technol., Espoo, Finland, Special Assignment Rep., 2008.

[11] W. Chen, K. B. Letaief, and Z. Cao, "Opportunistic network coding for wireless networks," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2007, pp. 4634–4639.

[12] J. Wu, Z. Chen, and M. Zhao, "An efficient data packet iteration and transmission algorithm in opportunistic social networks," *J. Ambient Intell. Humanized Comput.*, vol. 11, pp. 3141–3153, Sep. 2020, doi: 10.1007/s12652-019-01480-2.

[13] Y. Lin, X. Wang, F. Hao, Y. Jiang, Y. Wu, G. Min, D. He, S. Zhu, and W. Zhao, "Dynamic control of fraud information spreading in mobile social networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, early access, Aug. 9, 2019, doi: 10.1109/TSMC.2019.2930908.

[14] Y. Lin, Z. Cai, X. Wang, and F. Hao, "Incentive mechanisms for crowdblocking rumors in mobile social networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 9220–9232, Sep. 2019.

[15] X. Wang, X. Wang, F. Hao, G. Min, and L. Wang, "Efficient coupling diffusion of positive and negative information in online social networks," *IEEE Trans. Netw. Service Manage.*, vol. 16, no. 3, pp. 1226–1239, Sep. 2019.

[16] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Duke Univ., Durham, NC, USA, Tech. Rep. CS-200006, 2000.

[17] P. Guan and J. Wu, "Effective data communication based on social community in social opportunistic networks," *IEEE Access*, vol. 7, pp. 12405–12414, 2019, doi: 10.1109/ACCESS.2019.2893308.

[18] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[19] X. Xia, F. Chen, Q. He, J. C. Grundy, M. Abdelrazek, and H. Jin, "Cost-effective app data distribution in edge computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 1, pp. 31–44, Jan. 2021.

[20] M. Zhang, C. Zhang, V. S. Pai, L. L. Peterson, and R. Y. Wang, "Planetseer: Internet path failure monitoring and characterization in wide-area services," in *Proc. OSDI*, vol. 4, 2004, p. 12.

[21] J. Wu, J. Qu, and G. Yu, "Behavior prediction based on interest characteristic and user communication in opportunistic social networks," *Peer Peer Netw. Appl.*, vol. 14, no. 2, pp. 1006–1018, 2021, doi: 10.1007/s12083-020-01060-8.

[22] P. Hui, J. Crowcroft, and E. Yoneki, "BUBBLE Rap: Social-based forwarding in delay-tolerant networks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 11, pp. 1576–1589, Nov. 2011.

[23] X. Liu, Y. Du, M. Jiang, and X. Zeng, "Multiobjective particle swarm optimization based on network embedding for complex network community detection," *IEEE Trans. Comput. Social Syst.*, vol. 7, no. 2, pp. 437–449, Apr. 2020.

[24] Y.-M. Li, C.-T. Wu, and C.-Y. Lai, "A social recommender mechanism for e-commerce: Combining similarity, trust, and relationship," *Decis. Support Syst.*, vol. 55, no. 3, pp. 740–752, Jun. 2013.

[25] W. Yuan, L. Shu, H.-C. Chao, D. Guan, Y.-K. Lee, and S. Lee, "ITARS: Trust-aware recommender system using implicit trust networks," *IET Commun.*, vol. 4, no. 14, pp. 1709–1721, 2010.

[26] S. Yin, J. Wu, and G. Yu, "Low energy consumption routing algorithm based on message importance in opportunistic social networks," *Peer Peer Netw. Appl.*, vol. 14, no. 2, pp. 948–961, 2021, doi: 10.1007/s12083-021-01072-y.

[27] H. Jiang, Z. Liu, C. Liu, Y. Su, and X. Zhang, "Community detection in complex networks with an ambiguous structure using central node based link prediction," *Knowl.-Based Syst.*, vol. 195, May 2020, Art. no. 105626.

[28] J. Wu, S. Yin, Y. Xiao, and G. Yu, "Effective data selection and management method based on dynamic regulation in opportunistic social networks," *Electronics*, vol. 9, p. 1271, Aug. 2020, doi: 10.3390/electronics9081271.

[29] D. Zhang, C.-H. Hsu, M. Chen, Q. Chen, N. Xiong, and J. Lloret, "Cold-start recommendation using bi-clustering and fusion for large-scale social recommender systems," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 2, pp. 239–250, Jun. 2014.

[30] J. Luo, J. Wu, and Y. Wu, "Advanced data delivery strategy based on multiperceived community with IoT in social complex networks," *Complexity*, vol. 2020, Feb. 2020, Art. no. 3576542, doi: 10.1155/2020/3576542.

[31] J. Chen, O. R. Zaïane, and R. Goebel, "Detecting communities in social networks using max-min modularity," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, Apr. 2009, pp. 978–989.

[32] J. Wu, Z. Chen, and M. Zhao, "Information cache management and data transmission algorithm in opportunistic social networks," *Wireless Netw.*, vol. 25, no. 6, pp. 2977–2988, Aug. 2019, doi: 10.1007/s11276-018-1691-6.

[33] X. Zhang and Q. Zhu, "Hierarchical caching for statistical QoS guaranteed multimedia transmissions over 5G edge computing mobile wireless networks," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 12–20, Jun. 2018.

[34] K. Zhang, S. Leng, Y. He, S. Maharjan, and Y. Zhang, "Cooperative content caching in 5G networks with mobile edge computing," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 80–87, Jun. 2018.

[35] X. Cao, J. Zhang, and H. V. Poor, "An optimal auction mechanism for mobile edge caching," in *Proc. IEEE 38th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2018, pp. 388–399.

[36] R. Halalai, P. Felber, A.-M. Kermarrec, and F. Taiani, "Agar: A caching system for erasure-coded data," in *Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2017, pp. 23–33.

[37] J. Wu, Z. Chen, and M. Zhao, "Community recombination and duplication node traverse algorithm in opportunistic social networks," *Peer Peer Netw. Appl.*, vol. 13, pp. 940–947, Jan. 2020, doi: 10.1007/s12083-019-00833-0.

[38] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: An efficient routing scheme for intermittently connected mobile networks," in *Proc. ACM SIGCOMM Workshop Delay-Tolerant Netw. (WDTN)*, 2005, pp. 252–259.

[39] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, ''Efficient routing in intermittently connected mobile networks: The multiple-copy case,'' *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 77–90, Feb. 2008.
[40] J. Wu, X. Tian, and Y. Tan, ''Hospital evaluation mechanism based on mobile health for IoT system in social networks,'' *Comput. Biol. Med.*, vol. 109, pp. 138–147, Jun. 2019, doi: 10.1016/j.compbiomed.2019.04.021.
[41] J. Wu, G. Yu, and P. Guan, ''Interest characteristic probability predicted method in social opportunistic networks,'' *IEEE Access*, vol. 7, pp. 59002–59012, 2019, doi: 10.1109/ACCESS.2019.2915359.

XIAOMIN WU is currently pursuing the bachelor's degree with Central South University, Changsha, Hunan, China. Her research interests include wireless communications and networking, and big data research.

LIU CHANG is currently pursuing the master's degree with the School of Computer Science and Engineering, Central South University, China. Her research interests include wireless networks, big data research, and medical informatics.

JINGWEN LUO received the B.E. degree from Xidian University, Xi'an, China, in 2018. He is currently pursuing the master's degree with the School of Computer Science and Engineering, Central South University, China. His research interests include wireless networks, big data research, and medical informatics.

JIA WU (Member, IEEE) received the Ph.D. degree in software engineering from Central South University, Changsha, Hunan, China, in 2016. He is currently an Associate Professor with the School of Computer and Engineering, Central South University. Since 2010, he has been an Algorithm Engineer with IBM Company, Seoul, Republic of Korea, and IBM Company, Shanghai, China. His research interests include wireless networks, big data research, and medical informatics. He is also a Senior Member of China Computer Federation (CCF), and a member of ACM.