Effective Data Transmission Based on Path Prediction in Opportunistic Social Networks

Xuan ZHANG

central south university

Jia WU (✉ jiawu5110@163.com)

Central South University <https://orcid.org/0000-0001-9013-0818>

Genghua YU

central south university

Research

Keywords: data transmission, data management, opportunistic social networks, delivery ratio, Overhead

Posted Date: February 26th, 2020

DOI: <https://doi.org/10.21203/rs.2.24596/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Effective Data Transmission Based on Path Prediction in Opportunistic Social Networks

Xuan ZHANG and Jia WU* and GengHua YU

1. School of Computer science and engineering, Central South University, Changsha 410083,China;

Corresponding author email: jiawu5110@163.com

Abstract

With the development of social media, social networks have become an important platform for people to share and communicate. In social network communication, each participant can only pass information correctly if we find the goal of our communication. In other words, when people carry mobile devices for data transmission, they need to find a definite transmission destination to ensure the normal conduct of information exchange activities. In social networks, this is manifested in the process of data transmission by nodes, which requires analysis and judgment of surrounding areas, and finds suitable nodes for effective data classification and transmission. However, the node cache space in social networks is limited, and the process of waiting for the target node will cause end-to-end transmission delay. In order to improve such a transmission environment, this paper proposes a node trajectory prediction method named EDPPM algorithm. This algorithm can guarantee that nodes with high probability are given priority to obtain data information, which realized an effective data transmission mechanism. Through experiments and comparison of opportunistic transmission algorithms in social networks, such as Epidemic algorithm, Spray and Wait algorithm, and PRoPHET algorithm, our algorithm can improve the cache utilization of nodes, reduce data transmission delay, and improve the overall network efficiency.

Keywords data transmission, data management, opportunistic social networks, delivery ratio, Overhead

1. Introduction

With the development of social media, online social networks have become important information dissemination platforms [1-3]. Social networks site such as Facebook, Twitter, and Google plus support the information-sharing activities of billions of users every day. Through these platforms, people can exchange ideas and share information such as albums and videos anytime, anywhere. All information would be converted into data, which can be stored, sent, and received by users. As people move, everyone may become a data center, providing a data source for social communications.

Under these circumstances, using the traditional algorithm paradigm for the Internet and ad hoc networks where routing algorithms are computed based on topological information, is obviously not feasible. The first approach to routing in opportunistic social networks is a variation of controlled flooding. All messages are flooded and are limited by time to live, and then messages are delivered to their destination. This approach contacts the node that is receiving the message during flooding.

Obviously, the traditional approach is not suitable for opportunity transmission in social network scenarios. Large-scale, high-density flooding may cause broadcast storms, and sending or forwarding repeated control messages

in large quantities is also not conducive to node energy conservation. Therefore, in this case, screening nodes for data transmission is an effective method. However, if there is no good method of node selection, there may be situations where there is no suitable transmission target within the transmission range or the transmission target does not respond. This may cause a large amount of buffer occupation and transmission delay. On the one hand, many stranded information is stored in the device, and some of the information may be in a state of no user acceptance and response for a long period of time; On the other hand, new data information is received, emergency information is released in real time, etc. Because a great number of buffer space is occupied, information cannot be updated [25-28].

To solve these problems, this work proposes Effective Data Transmission based on Path Prediction and Management (EDPPM) method. The method performs node trajectory prediction by analyzing the potential regularity of human behavior. In the research of trajectory prediction, we also found that by merging the two records that are most similar in the user's historical trajectory, it is possible to more accurately reflect the user's trajectory. We call it trajectory merge algorithm, and this discovery has contributed to improving the prediction accuracy of our algorithm. The purpose of establishing a trajectory prediction algorithm is to perform fast node screening through the prediction of target nodes, to ensure that nodes with high probability obtain data information first, and achieve the goal of adjusting the cache.

The main contributions of this study include the following:

- (1) A trajectory merge algorithm was found, which we call the Merge algorithm. The algorithm can more accurately reflect the user's trajectory, and significantly improves the accuracy of trajectory prediction.
- (2) A trajectory prediction algorithm is proposed, which we call the EDPPM algorithm. The algorithm guarantees that nodes with high probability obtain data information first and achieve the goal of adjusting the cache.
- (3) Based on the simulation results, the performance of the EDPPM algorithm is analyzed and compared with the Epidemic algorithm, Spray and Wait algorithm, and PRoPHET algorithm, it is found that the EDPPM algorithm performs better in the aspects of Delivery ratio, Energy consumption, Overhead on average, Average end-to-end on delay, etc.

2. Related works

To improve the network environment, many researchers focus on the design of algorithms in social opportunity networks.

Ref [5] Proposed the Direct Delivery algorithm, which is a very simple algorithm. The operation method of the algorithm in social networks is very simple, that is, if and only if the source node encounters the destination node of the data packet, it will forward the data packet. This approach has its advantages; it minimizes network costs. However, this method also has some obvious non-negligible disadvantages: extremely low transmission success rate and significant transmission delay.

In order to improve the problems in the Direct Delivery algorithm, Ref [6] proposed an algorithm with infectious diseases as its core idea, and named it Epidemic. In the Epidemic algorithm, if two nodes meet, they will first check each other's packets, then compare them with the packets they store, and finally exchange what is not in their packets. The core idea of this process is to use the xor operation to ensure that the two meeting nodes can obtain data information that they did not have before. For the Epidemic algorithm, when the network bandwidth, cache capacity, and other resources of the node are sufficient, the transmission delay can be greatly reduced while ensuring a successful propagation rate. This algorithm seems to solve the problem of transmission success rate and transmission delay, but it has an obvious fatal flaw that is that with the increase of network nodes, it is easy to cause network congestion. This method is used in many practical social communication environments, but often cannot achieve good results due to resource constraints.

Ref [7] proposed the Spray and Wait algorithm. The algorithm continues the infection behavior between nodes of the Epidemic algorithm and improves the problem of excessive resource consumption. The Spray and Wait algorithm includes a spray phase and a wait phase. During the Spary phase, the source node continuously broadcasts a copy of the data until N data are diffused out. During the Wait phase, neighbor nodes move with data copies. When they encounter the target node, they will use the Direct Delivery algorithm to pass a copy of the carried data to the target node. This algorithm effectively solves the problem of fast resource consumption in the Epidemic algorithm, but how many data copies the source node needs to broadcast in the Spray stage is still a problem. Therefore, the Spray and Wait algorithm cannot avoid the rapid consumption of resources, especially when the parameter N is manually set. In summary, the Spray and Wait algorithm may cause nodes in the network to die quickly in some cases.

Ref [8] proposed the PRoPHET algorithm. They calculated the target nodes that could be passed through statistical calculations. Then estimate the appropriate transmission node to form a data packet to improve the utilization of the network. Ref [9] proposed a probability-based MV algorithm, which calculates the probability of forwarding by historical statistics of when nodes meet and access the communication area.

Ref [10] proposed the MaxProp algorithm based on priority setting. The algorithm determines the transmission queue by setting the priority matrix when two nodes meet to reduce resource consumption. It also improves efficiency while determining the transmission target queue. It forms a self-organizing transmission area in space to achieve selected communication between nodes.

Ref [11] proposed the Context aware routing algorithm to calculate the transmission rate to the target node. The intermediate node is obtained by calculating the periodic transition probability. The information is directly transmitted to the node at a high transmission rate along with data collection and data packets.

Through the analysis and discussion of the above methods, we may find an effective method that can improve the data transmission in social networks. It guarantees that nodes with a high probability obtain data information first, thereby improving the cache utilization rate of nodes, reducing data transmission delay, and improving the overall network efficiency. The core of implementing this method is the EDPPM algorithm we will propose below.

3. Methods

In this section, we will first give some definitions that will be used later to help understand the algorithms that follow. $CStat_{i,j}$ represents the set of all moments when a mobile user i starts talking on the day j . $CStation_i$ is a collection of all base stations that appear in the historical call history of user i . Record a call record of user i as $\langle Tp, Station_p \rangle$ (x_p, y_p), T_p is the time to start the call, $T_p \in CStat_{i,j}$; (x_p, y_p) is the latitude and longitude of the communication base station during a call, $Station_p \in CStation_i$. The call behavior data of user i on the day j is recorded as $Call_{i,j}$, which expressed as $Call_{i,j} = \{ \langle T_1, Station_1 \rangle, \langle T_2, Station_2 \rangle, \dots, \langle T_n, Station_n \rangle \}$.

$Call_{i,j}$ is an ordered set and $T_1 < T_2 < \dots < T_n$.

Definition 1: Track Let R be a division of the set S_i , and $R_k \in \{Station_{k1}, Station_{k2}, \dots\}$. Then $Call_{i,j}$ can be rewritten as $Track_{i,j} = \{ \langle T_1, R_1 \rangle, \langle T_2, R_2 \rangle, \dots, \langle T_n, R_n \rangle \}$, represented user's trajectory on day j .

Definition 2: Behavior pattern Define point pairs $\langle T_k, T_{k+1}, R_k \rangle$, used to indicate that user i is in the R_k area during the time period $[T_k, T_{k+1}]$. Ordered set $Pattern_{i,j} = \{ \langle T_1, T_2, R_1 \rangle, \langle T_3, T_4, R_2 \rangle, \dots, \langle T_{2n-1}, T_{2n}, R_n \rangle \}$, called as user i 's behavior pattern on day j .

Definition 3: Track prediction Known sequence $\{ \langle T_1, R_1 \rangle, \langle T_2, R_2 \rangle, \dots, \langle T_{n-1}, R_{n-1} \rangle \}$ ($T_1 < T_2 < \dots < T_{n-1}$), time to be predicted T_n ($T_{n-1} < T_n$) and user's position R_n at time T_n . The area R' where the user is located at time T_n is obtained through some prediction algorithm. If $R' = R_i$, the track prediction is considered correct; If $R' \neq R_i$, the track prediction is considered wrong.

Definition 4: Trajectory prediction strategy Known sequence $\{<T_1, R_1>, <T_2, R_2> \dots <T_n, R_n>\}$ ($T_1 < T_2 < \dots < T_n$) and moment to be predicted T' and $|T' - T_i| \leq \forall k(T' - T_k)$, $T_i \in CStat_{i,j}$. Then predict the position of sequence $\{<T_1, R_1>, <T_2, R_2> \dots <T_{n-1}, R_{n-1}>\}$ at T_i ; If $R' = R_i$ is obtained through some prediction algorithm, the prediction is considered correct; If $R' \neq R_i$, the prediction is considered to be wrong.

Based on the above definitions, we use a trajectory prediction to help nodes in social networks quickly find nodes that need to transmit data. In actual social network communication, the base station is fixed, and the equipment used for communication is moved with the user at any time. Therefore, the data in the bill record does not reflect the user's location well. Considering this situation, the dialogue record needs to be perfected and corrected to remove the base station switching that occurs during a call and to obtain the user's behavior pattern based on the area division. Transform user discrete behavior patterns into continuous behavior patterns. Based on this model, EDPPM algorithm is used to predict trajectory. At the same time, considering that the user will not have a large position movement in the case of high probability during communication, a trajectory merge algorithm will be proposed later to further improve the accuracy of our algorithm.

Before introducing our EDPPM algorithm, first introduce our judgment criteria for base station conversion in this experiment, the area division algorithm, user behavior mode algorithm, and trajectory merge algorithm used.

(1) Station Conversion In the actual social network environment, due to various objective problems, communication equipment such as mobile phones do not always choose the nearest base station, so the base station location in the bill record does not accurately reflect the user's location.

We assume the user's two calls are $<T_i, Station_i(x_i, y_i)>, <T_j, Station_j(x_j, y_j)>$ ($Station_i \neq Station_j, T_i < T_j$). The minimum user speed between these two base stations is

$$v_{i,j} = \frac{\sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}}{T_j - T_i} \quad (1)$$

Let the upper limit of human movement speed be $v = 120\text{km} / \text{h}$. If the minimum moving speed between any base stations exceeds the upper limit, a base station conversion is considered to occur, and the data of the base station conversion in the record of the dialog box is corrected by this method.

(2) Region Divide Generally, the active area of a user includes multiple base stations, and a base area of the user can be obtained by combining base stations in the active area. When the signal ranges of two active base stations overlap (here the base station distance is less than 1.5 km as a standard), the two base stations belong to the same area.

Table1 Region Division

Algorithm 1 Region Division

Input: $Station_i$

Output: R

1、for each i, j

2、 If $distance(Station_i, Station_j) < 1.5\text{km}$

3、 $Station_i \in R_k$

4、 $R_k = R_k \cup Station_j$

5、 end

6、end

(3) User Behavior Patterns The user's behavior pattern obtained from the bill data is discrete, and the user's trajectory cannot be predicted from these discrete data. We assume that if the two consecutive call areas of the user are the same, the user is considered to have not moved during these time periods. Transform users' discrete behavioral

patterns into continuous behavioral patterns. When $Track_{i,j}$ is given, the behavior pattern of user i on day j can be obtained according to Algorithm 2.

Table 2 Pattern Extraction

Algorithm 2 Pattern Extraction

Input: $Track_{i,j}$

Output: $Pattern_{i,j}$

```

1、 for each  $k$  do
2、   if  $R_k = R_{k+1}$ 
3、     if  $S$  is NULL
4、        $S = \langle T_k, T_{k+1}, R_k \rangle$ 
5、     else
6、        $S = \langle S(1), T_{k+1}, S(3) \rangle$ 
7、     end
8、   else
9、      $Pattern_{i,j} = Pattern_{i,j} \cup S$ 
10、     $S = NULL$ 
11、  end
12、 end

```

(4)Track Merge Because the user's call behavior is very discrete throughout the day, it is difficult to accurately predict the user's true trajectory from such discrete data. So we select the two records that are most similar in the user's historical trajectory to merge to reflect the user's trajectory more realistically.

scene 1 The real trajectory of user i on day j , k is $\{\langle T_1, R_1 \rangle, \langle T_2, R_2 \rangle, \langle T_3, R_3 \rangle, \langle T_4, R_4 \rangle\}$, expressed as $Track_{i,j} = \{\langle T_1, R_1 \rangle, \langle T_2, R_2 \rangle, \langle T_3, R_3 \rangle\}$, $Track_{i,k} = \{\langle T_1, R_1 \rangle, \langle T_3, R_3 \rangle, \langle T_4, R_4 \rangle\}$. Then merge the trajectories of day j and k . Getting the real trajectory is $Track_{i,j} \cup Track_{i,k} = \{\langle T_1, R_1 \rangle, \langle T_2, R_2 \rangle, \langle T_3, R_3 \rangle, \langle T_4, R_4 \rangle\}$.

The trajectory similarity of the user's two-day trajectory is measured by the trajectory difference.

Theorem 1 In a three-dimensional space (x, y, t) , a continuous curve formed by the sequential connection of sequence point pairs in $Track_{i,j}$ is denoted as $(Line_{d1}^x(t), Line_{d1}^y(t), t)$. The trajectory difference between $d1$ and $d2$ is:

$$Dis_i(m, n) = \frac{\int_{t_{start}}^{t_{end}} \sqrt{(Line_{d1}^x(t) - Line_{d2}^x(t))^2 + (Line_{d1}^y(t) - Line_{d2}^y(t))^2} dt}{t_{start} - t_{end}} \quad (2)$$

And (t_{start}, t_{end}) is the intersection of the domain of the curve $(Line_{d1}^x(t), Line_{d1}^y(t), t)$ and the curve $(Line_{d2}^x(t), Line_{d2}^y(t), t)$.

Proof: For user i , at the same time t , the position on day m is (x_m, y_m) , and the position on day n is (x_n, y_n) . The distance between these two days at time t is:

$$L = \sqrt{(x_m - x_n)^2 + (y_m - y_n)^2} \quad (3)$$

The average distance between two days is:

$$Dis_i(m, n) = \frac{\int_{t_{start}}^{t_{end}} L}{t_{start} - t_{end}} \quad (4)$$

The curve of the track corresponding to m and n is $(f_m^x(t), f_m^y(t), t)$, $(f_n^x(t), f_n^y(t), t)$, The above formula can be written as:

$$Dis_i(m, n) = \frac{\int_{t_{start}}^{t_{end}} \sqrt{(f_m^x(t) - f_n^x(t))^2 + (f_m^y(t) - f_n^y(t))^2}}{t_{start} - t_{end}} \quad (5)$$

If f takes a continuous curve $(Line_j^x(t), Line_j^y(t), t)$ formed by the sequence point pairs in $Track_{i,j}$, then the above formula can be written as:

$$Dis_i(m, n) = \frac{\int_{t_{start}}^{t_{end}} \sqrt{(Line_{d1}^x(t) - Line_{d2}^x(t))^2 + (Line_{d1}^y(t) - Line_{d2}^y(t))^2}}{t_{start} - t_{end}} \quad (6)$$

Complete.

According to the above method of trajectory difference, determine the degree of trajectory similarity between the two days. The method of trajectory merging is to calculate $Dis_i(p, q)$, select the first $th\%$ of the user trajectories with the smallest trajectory difference to merge, and obtain the merged set of user trajectories, $Merge_i$. The trajectory merge algorithm is as follows:

Table 3 Merge

Algorithm 3 Merge

Input: $Track_{i,j}$, n , th

Output: $Merge_i$

1、 for every p do

2、 for every q do

3、 if $p = q$

4、 $Dis_{p,q} = MAXNUM$

5、 else

6、 $Dis_{p,q} = Dis_i(p, q)$

7、 end

8、 end

9、 end

10、 for $j = 1$ to $th * C_n^2$

11、 $Dis_{p,q}$ is the j_{th} smallest num in Dis

12、 $Merge_i = Merge_i \cup \{Track_{i,p} \cup Track_{i,q}\}$

13、 end

14、 $Merge_i = Merge_i \cup Track_i$

Based on the above, we will give the content of the EDPPM algorithm. EDPPM is well-founded. Human behavior is regular. The behavior of future moments often depends on the behavior of the previous moment. In this regard, if the

behavior of the user at the next moment can be predicted, then the node will be quickly find the transmission target node and reduce the data transmission delay.

scene 2 It is known that the user's trajectory sequence before time T_3 is $\{<T_1, R_1>, <T_2, R_2>\}$. The most frequent subsequence containing the sequence contained in the user's historical track $Track_i$ is $\{<T_1, R_1>, <T_2, R_2>, <T_3, R_3>\}$. Then use $<T_3, R_3>$ as the prediction result, this means predicting that the user will be at R_3 at time T_3 .

The law of human behavior has its own uniqueness, and people conduct social activities on the basis of heaven. At certain times or times of the day, human behavior is always similar. For example, people usually work and study during the day and sleep at night. And for most people, they usually work on weekdays and rest on weekends. Based on this feature, we can get the following scenarios.

scene 3 For $Pattern_i$, $\exists j, k \in \{\text{weekdays}\}$, $\exists p \in \{\text{weekends}\}$, let $(Pattern_{i,j} \cap Pattern_{i,p}) \subset (Pattern_{i,j} \cap Pattern_{i,k})$

According to the above rules and scenarios, algorithm 4 (called EDPPM algorithm) is proposed to predict the user's trajectory. In the foregoing, we have given the trajectory merging algorithm. Merge user's trajectory set to $Merge_i$. It turns out that $Merge_i$ can achieve better prediction results instead of $Pattern_i$. Therefore, $Merge_i$ is directly brought into the calculation in the EDPPM algorithm. The EDPPM algorithm is as follows:

Table 4 EDPPM

Algorithm 4 EDPPM

Input: $<T_1, R_1>, <T_2, R_2>, \dots, <T_{n-1}, R_{n-1}> (T_1 < T_2 < \dots < T_{n-1}), T_n, Merge_i$

Output: R'

- 1、 Define T_k in $Merge_{i,j}$ to find a $<T_{2m-1}, T_{2m}, R_m>$ where $T_{2m-1} \leq T_k \leq T_{2m}$
 - 2、 $Satisfy = \{ \}$
 - 3、 for each j do
 - 4、 if T_n in $Merge_{i,j}$
 - 5、 $Satisfy = Satisfy \cup Merge_{i,j}$
 - 6、 end
 - 7、 end
 - 8、 for each j do
 - 9、 find max_i that T_{n-1} in $Satisfy_j$, T_{n-2} in $Satisfy_j$, ..., T_{n-i} in $Satisfy_j$
 - 10、 $EDPPM_j = n-i$
 - 11、 end
 - 12、 $p = \max(EDPPM)$
 - 13、 if $p \neq 0$
 - 14、 $R' = R_q$, that exists q with $<T_{2q-1}, T_{2q}, R_q> \in Satisfy_p$ and $T_{2q-1} \leq T_n \leq T_{2q}$
 - 15、 else
 - 16、 user Algorithm Most Frequent get R'
 - 17、 end
-

Corollary 1: The EDPPM algorithm can reduce overhead costs when the source node transmits messages to its neighbors.

Proof 1: First analyze the Epidemic and Spray and Wait algorithms. These two algorithms must send information to neighbor nodes when transmitting information. The total overhead is as follows:

$$O_{\text{Epidemic}} = O_{\text{SprayandWait}} = n \quad (7)$$

However, when using the EDPPM algorithm, we can find that the algorithm quickly judges the high probability nodes of the received data by predicting the user's next behavior, and improves the accuracy of the prediction through the trajectory merging strategy to avoid invalid overhead. Obviously we can recognize:

$$O_{EDPPM} < O_{Epidemic} = O_{SprayandWait} \quad (8)$$

Using EDPPM algorithm can reduce the delay of data transmission and improve the cache utilization rate of nodes.

Below we use the time complexity to evaluate the performance of the EDPPM algorithm to prove the effectiveness and good performance of our algorithm.

Corollary 2: The time complexity in trajectory prediction in the EDPPM scheme is $O(n)$.

Proof 2: In the EDPPM algorithm, the first for loop finds the set of behavior patterns of all users whose behavior pattern on day j contains the time point T_n to be predicted; the second for loop finds the most frequent subsequence containing the sequence to be predicted, and then uses R' corresponding to the time to be predicted in the subsequence as the result of the trajectory prediction. Trajectory prediction model in EDPPM algorithm is parallel to time complexity, $O(n+n+1)=O(2n+1)$. Obviously, the time complexity of the EDPPM algorithm is $O(n)$. We know that the Epidemic algorithm uses a meet-and-transfer strategy with a time complexity of $O(n^2)$; The Spray and Wait algorithm uses a spray model, and the time complexity of the Spray and Wait algorithm is also $O(n)$. Therefore, based on the time complexity, the EDPPM and Spray and Wait algorithms are better than the Epidemic algorithm.

4. Experimental

OMNET ++ is a powerful simulation software that can simulate the motion trajectories of various types of nodes and record the information transmission process in detail. In this experiment, the OMNET ++ simulation tool was used to simulate the real environment to evaluate the performance of the EDPPM algorithm. In this simulation experiment, we will use three algorithms for opportunistic transmission to compare with the EDPPM algorithm: Epidemic algorithm, PRoPHET algorithm, and Spray and wait algorithm.

The following details the parameter information used in this simulation experiment:

Parameter limit	detail
Simulation time	ranging from one to six hours
Simulation area	about 2500m long and about 1400m wide
Node	The scene contains a maximum of 1000 nodes; All nodes are randomly distributed in the area; The node sends data packets by broadcasting; The node has 100J energy in the initial state; The task of each node is 10 packets; The packet loss threshold of each node is 15%. Nodes will transmit data packets at a speed of 0.5–1.5 m / s; The cache size of each node is 5MB.
Data packet	Every 25 seconds to 35 seconds, a data packet should send one of its copies; each transmission will consume about 0.125J of energy.
Others	The mode of transmission uses a social model

In this section we will analyze the performance of the EDPPM algorithm and compare it with three algorithms mentioned earlier. Before comparing the performance of the algorithms, we must first analyze the four important parameters that affect the performance of the EDPPM algorithm, and analyze the impact of the number of collaborative nodes on these parameters based on the experimental results.

(1)Delivery ratio: This parameter is closely related to how to select the transmission node, which directly affects the performance of the algorithm.

Figure.1 The relationship between the number of cooperation nodes and delivery ratio

According to Figure.1, it can be clearly found that the delivery ratio increases significantly with the increase in the number of collaborative nodes. When the number of cooperative nodes reaches 4, the delivery ratio appears to peak. Obviously, as the number of nodes continues to increase, due to the memory limitation, the node cannot cache all the data and the delivery ratio will decrease.

(2)Energy consumption: This parameter indicates the energy consumption during data transmission, including transmission loss and computational loss.

Figure.2 The relationship between the number of cooperation nodes and energy consumption.

Obviously, according to Figure 2, we can find that the curves of Energy consumption and Delivery ratio show similar rules. As the number of collaborative nodes increases, energy consumption increases, but when the number of nodes reaches 4, Energy consumption peaks.

When the number of nodes exceeds 4, Energy consumption will drop because some nodes are blocked and waiting. But the node can control its own energy consumption. If the node cache allows receiving new information, the node allows to accept new information at the cost of energy consumption.

(3)Overhead on average: This parameter is used to indicate the routing cost of data transmission between any two data.

(4)*Average end-to-end on delay*: This parameter reflects the average delay time when transmitting data between any two points.

Figure.3The relationship between the number of cooperation nodes and overhead on average.

Figure.4 The relationship between the number of cooperation nodes and average end-to-end on delay

It is obvious from Figure.3 and Figure.4 that routing overhead and transmission delay are another case. Obviously, when the number of cooperative nodes gradually increases from 1 to 5, the routing overhead and transmission delay will gradually decrease due to the occupation of the buffer space, and the peak value will be reached when the number of cooperative nodes reaches 5. When the number of cooperating nodes exceeds 5, the performance of the algorithm is affected by the memory size of the nodes, and the routing overhead and transmission delay will gradually increase.

Below we will compare the performance of the above four parameters in the simulation results of the four algorithms to compare their performance.

Figure.5 Delivery ratio and simulation time.

Figure.5 describes the comparison results of the delivery ratio of the four algorithms. Obviously, compared with the other three algorithms, the delivery ratio of the EDPPM algorithm is always the highest. When the simulation time is 6h, the delivery ratio of the EDPPM algorithm reaches about 0.90. The reason why the EDPPM algorithm performs better in the delivery ratio is because the source node can accurately transmit the data packet to the neighbor node in the algorithm. For the prophet algorithm, the nodes need to calculate the probability of encounter before determining the behavior, but because the algorithm lacks a control center, the nodes must collect sufficient effective information before making a decision each time, which results in the nodes must wait long enough to make more accurate predictions. In order to limit the limit flooding behavior of the epidemic algorithm, the spray and wait algorithm uses a strategy of sending a fixed number of copies for each data packet to solve the problem. Although the limit flooding is limited, the spray and wait algorithm has a lower Delivery ratio. The performance of the algorithm generally depends on the choice of parameters. When the parameters are fixed, the algorithm is greatly affected by environmental factors, so the transmission rate is also low.

Figure.6 Energy consumption and simulation time.

Figure.6 depicts the energy consumption comparison results of the four algorithms. From Figure.6, it can be seen that compared with the other three algorithms, the energy consumption of the EDPPM algorithm is the lowest, and the number of sent data packets is the largest. Because the EDPPM algorithm uses mutual exclusion or calculation between nodes, it can save energy when data packets are reduced and transmitted. It can also be seen from the figure that the Epidemic algorithm consumes the most energy, because over-flood may generate duplicate data packets, and the Spray and Wait and PProPHET algorithms perform better than the Epidemic algorithm in terms of energy consumption.

Figure.7 Overhead on average and simulation time

Figure.7 depicts the comparison results of the overhead on average of the four algorithms. The figure shows the relationship between routing cost and time. It can be clearly seen from the figure that the routing overhead of the

EDPPM algorithm is relatively flat with time, and there is no large fluctuation. We can understand that the routing overhead of the EDPPM algorithm is not affected by time. In terms of routing overhead, Prophet algorithm is slightly inferior to EDPPM algorithm, which is superior to the other two algorithms. In the early stage, its routing cost is lower than the other two algorithms, and it is basically stable between 30-60 in the later stage. This is because as time increases, the number of nodes filtered by the algorithm will gradually decrease, so the selection of information transmission nodes will be more accurate, and the number of transmission nodes used for sharing information will also stabilize. Therefore, the routing overhead of the Prophet algorithm is relatively stable. The curve radian obtained by the Spray and Wait algorithm is similar to the algorithm proposed in this paper. In terms of Overhead on average performance, the initial Spray and Wait algorithm has the largest Overhead on average among these four algorithms, but the routing overhead will gradually decrease as time increases. This is because the spray and wait routing algorithm reduces the amount of data transmitted in the network and has good scalability. But overall, the routing cost of the Spray and Wait algorithm is very large and fluctuating, because in order to maximize the success rate of packet transmission, each node carries a duplicate data packet, which leads to there are a large number of data packet copies, the network is down, the network structure is unstable, so the routing overhead is large and unstable.

Figure.8 Average end-to-end on delay

Figure.8 describes the average end-to-end on delay comparison results of the four algorithms. It can be clearly seen from the figure that the Average end-to-end on delay of the EDPPM algorithm is the lowest and relatively stable. For the Epidemic algorithm, the First Contact algorithm used by the algorithm only sends copies to the nodes encountered for the first time, and usually the probability that the nodes encountered for the first time are the target nodes is very small. Since it is necessary to calculate the information for each transmission, the Epidemic algorithm has the highest transmission delay. The Prophet algorithm determines the priority of the message according to the transmission cost, that is, the transmission cost is used to estimate the success rate of the packet successfully transmitted to the destination node, and the incremental averaging method is used to estimate. From the figure, the EDPPM algorithm's Average end-to-end on delay and Spray and Wait Routing algorithm are relatively close. This is because similar to the Spray and Wait Routing algorithm, the EDPPM algorithm uses resources to filter nodes based on different factors. However, we can see from the figure that the transmission delay of the EDPPM algorithm tends to be smoother with time, which is more stable than the Spray and Wait Routing algorithm. And from the perspective of the overall curve change, the transmission delay of the EDPPM algorithm does not increase exponentially with time.

Through the above comparative analysis, it can be noticed that compared with the other three algorithms, EDPPM has better performance in many aspects.

5. Conclusion

This paper proposes an efficient trajectory prediction algorithm named EDPPM, which can guarantee that nodes with high probability can get information first. According to the experimental results, compared with three algorithms for opportunistic transmission, the EDPPM algorithm is superior to the other three algorithms in terms of the performance of Delivery ratio, Energy consumption, Overhead on average and Average end-to-end on delay. Using this algorithm can improve the node's cache utilization, reduce data transmission delay, and improve the overall network efficiency.

In the future works, how to further improve the accuracy of trajectory prediction is the focus of our future research. Also how to ensure the security of data transmission in social opportunity communication networks is still an important issue that remains our concern.

References

- [1] Jia, W. U., Chen, Z., & Zhao, M. (2017). "Effective information transmission based on socialization nodes in opportunistic networks." *Computer Networks*, 129, 297–305.
- [2] Jia WU, Zhigang CHEN and Ming ZHAO, "Information cache management and data transmission algorithm in opportunistic social networks", *Wireless networks*. August 2019, Volume 25, Issue 6, pp 2977–2988, 2019.
- [3] Peiyuan GUAN and Jia WU, "Effective data communication based on social community in social opportunistic networks", *IEEE ACCESS*, Volume: 7, Issue:1, pp. 12405-12414, 2019.
- [4] Wang, Xiaoming , et al. "A novel approach for inhibiting misinformation propagation in human mobile opportunistic networks." *Peer-to-Peer Network and Application* 10.2(2017):377-394.
- [5] Juang P, Oki H, Wang Y, Martonosi M, Peh LS, Rubenstein D. Energy-Efficient computing for wildlife tracking: Design tradeoffs and early experiences with ZebraNet. In: Proc. of the 10th Int'l Conf. on Architectural Support for Programming Languages and Operating Systems. *New York: ACM*, 2002. 96–107.
- [6] Pan H, Chaintreau A, Scott J, Gass R, Crowcroft J, Diot C. "Pocket switched networks and human mobility in conference environments." In: Proc. of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking. *Philadelphia: ACM*. 2005.244–251.
- [7] Hull, Bret , et al. "CarTel: A distributed mobile sensor computing system." *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, SenSys 2006*, Boulder, Colorado, USA, October 31 - November 3, 2006 2006.
- [8] Moreira, Waldir , P. Mendes , and S. Sargento . "Opportunistic routing based on daily routines." *World of Wireless, Mobile & Multimedia Networks IEEE*, 2012.
- [9] Pan, Chenji , et al. "Markov Chain-Based Routing Algorithm in Delay-Tolerant Networks." *Applications and the Internet (SAINT), 2012 IEEE/IPSJ 12th International Symposium on IEEE*, 2012.
- [10] Manam, Mahendran, and Murthy. "Performance modeling of routing in delay-tolerant networks with node heterogeneity." *Fourth International Conference on Communication Systems & Networks IEEE*, 2012.
- [11] Gibran K. The Prophet: a new annotated edition [M] S. 1.] : *Oneworld Publications*, 2012.
- [12] Tie, Xiaozheng , A. Venkataramani , and A. Balasubramanian . "R3: Robust Replication Routing in Wireless Networks with Diverse Connectivity Characteristics." *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking, MOBICOM 2011*, Las Vegas, Nevada, USA, September 19-23, 2011 DBLP, 2011.
- [13] Wu, Dapeng , et al. "Node Service Ability Aware Packet Forwarding Mechanism in Intermittently Connected Wireless Networks." *IEEE Transactions on Wireless Communications* 15.12(2016):1-1.
- [14] WONG, Ka Wai Gary, Chang Y , Jia X , et al. "Performance evaluation of social relation opportunistic routing in dynamic social networks." *International Conference on Computing* 2015.
- [15] Gao, Wei Gao Wei , and G. C. G. Cao . "On exploiting transient contact patterns for data forwarding in Delay Tolerant Networks." *IEEE International Conference on Network Protocols IEEE Computer Society*, 2010.
- [16] Azzuhri, Saaidal Razalli , et al. "An Efficient Hybrid MANET-DTN Routing Scheme for OLSR." *Wireless Personal Communications* 89.4(2016):1335-1354.
- [17] Erdős, Pál, and Rényi, Alfréd. "On random graphs. I. " *Publicationes Mathematicae* 4(1959):3286-3291.
- [18] Jia WU, Xiaoming TIAN and Yanlin TAN, "Hospital evaluation mechanism based on mobile health for IoT system in social networks", *Computers in Biology and Medicine*, Volume 109, June 2019, Pages 138-147.
- [19] Goel, Ashish , S. Rai , and B. Krishnamachari . "Sharp Thresholds for Monotone Properties in Random Geometric

Graphs." *Thirty-sixth Acm Symposium on Theory of Computing ACM*, 2004.

- [20] Fernandez-Marquez, Jose Luis , et al. "The ONE-SAPERE Simulator: A Prototyping Tool for Engineering Self-Organisation in Pervasive Environments." *2014 IEEE 8th International Conference on Self-Adaptive and Self-Organizing Systems (SASO) IEEE*, 2014.
- [21] Hang, Guo , et al. "Adaptive Epidemic Routing Algorithm Based on Multi Queue in DTN." *Journal of Chinese Computer Systems* (2012).
- [22] Wang, Guizhu , B. Wang , and Y. Gao . "Dynamic Spray and Wait Routing Algorithm with Quality of Node in Delay Tolerant Network." *International Conference on Communications & Mobile Computing IEEE Computer Society*, 2010.
- [23] J W and ZG C, "Sensor communication area and node extend routing algorithm in opportunistic networks", *Peer-to-Peer Networking and Applications*. January 2018, Volume 11, Issue 1, pp 90–100. DOI:10.1007/s12083-016-0526-4
- [24] J W, Z C and M Z, "An efficient data packet iteration and transmission algorithm in opportunistic social networks", *Journal of Ambient Intelligence and Humanized Computing*. pp 1-17, 2019. DOI: 10.1007/s12652-019-01480-2
- [25] Gerzaguet R, Bartzoudis N, Baltar L G, et al. The 5G candidate waveform race: a comparison of complexity and performance. *Eurasip Journal on Wireless Communications & Networking*, 2017, 2017(1):13.
- [26] Mao Y, Clerckx B, Li V O K. Rate-splitting multiple access for downlink communication systems: bridging, generalizing, and outperforming SDMA and NOMA. *Eurasip Journal on Wireless Communications & Networking*, 2018, 2018(1):133.
- [27] He S, Dong X, Lu W S. Localization algorithms for asynchronous time difference of arrival positioning systems. *Eurasip Journal on Wireless Communications & Networking*, 2017, 2017(1):64.
- [28] Tan W, Jin S, Wen C K, et al. Spectral efficiency of multi-user millimeter wave systems under single path with uniform rectangular arrays. *Eurasip Journal on Wireless Communications & Networking*, 2017, 2017(1):181.

Abbreviations

EDPPM: Effective Data Transmission based on Path Prediction and Management

Availability of Data and Materials

Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Competing interests

The authors declare that they have no competing interests (All financial).

Funding:

This work was supported in The National Natural Science Foundation of China(61672540); Hunan Provincial Natural Science Foundation of China (2018JJ3299, 2018JJ3682).

Authors contributions:

Jia WU, Xuan ZHANG and Genghua YU designed the project and drafted the manuscript, collected the data, wrote the code and performed the analysis. All participated in finalizing and approved the manuscript.

Brief introduction of the author

Xuan ZHANG is bachelor student in computer science and engineering from Central South University, Changsha, Hunan, China. Her research interests include wireless communications and networking, big data research and medical informatics.

Jia WU received the Ph.D. Degrees in software engineering Central South University, Chang-sha, Hunan, P.R.China, in 2016. He is associate professor in School of computer and engineering, Central South University. Since 2010, he has been Algorithm engineer in IBM company in Seoul, Republic of Korea and in Shang-hai, P.R.China. He is a senior member of CCF(China Computer Federation), a member of IEEE and ACM. His research interests include wireless network, big data research, Medical Informatics.

GengHua YU received his Ph.D. degree from Central South University, School of Computer Science, Changsha, Hunan Province in 2019. She studied for a master's degree at Central South University in 2017. She is an outstanding graduate of Nanchang University during her undergraduate degree. Her research interests include wireless communications and networking, medical informatics, big data research.

Figures

Figure 1

The relationship between the number of cooperation nodes and delivery ratio

Figure 2

The relationship between the number of cooperation nodes and energy consumption.

Figure 3

The relationship between the number of cooperation nodes and overhead on average.

Figure 4

The relationship between the number of cooperation nodes and average end-to-end on delay

Figure 5

Delivery ratio and simulation time.

Figure 6

Energy consumption and simulation time.

Figure 7

Overhead on average and simulation time

Figure 8

Average end-to-end on delay