WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2015; **15**:1633–1648
DOI: 10.1002/wcm.2451

RESEARCH ARTICLE

Non-cooperative game-based packet ferry forwarding for sparse mobile wireless networks

Xin Guan[1], Min Chen[2] and Tomoaki Ohtsuki[1]*

[1] Department of Information and Computer Science, Keio University, Yokohama, Japan
[2] School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, 430074, China

ABSTRACT

In sparse mobile wireless networks, normally, the mobile nodes are carried by people, and the moving activity of nodes always happens in a specific area, which corresponds to some specific community. Between the isolated communities, there is no stable communication link. Therefore, it is difficult to ensure the effective packet transmission among communities, which leads to the higher packet delivery delay and lower successful delivery ratio. Recently, an additional ferry node was introduced to forward packets between the isolated communities. However, most of the existing algorithms are working on how to control the trajectory of only one ferry work in the network. In this paper, we consider multiple ferries working in the network scenario and put our main focus on the optimal packet selection strategy, under the condition of mutual influence between the ferries and the buffer limitation. We introduce a non-cooperative Bayesian game to achieve the optimal packet selection strategy. By maximizing the individual income of a ferry, we optimize the network performance on packet delivery delay and successful delivery ratio. Simulation results show that our proposed packet selection strategy improves the network performance on packet delivery delay and successful delivery ratio.

KEYWORDS
sparse mobile wireless networks; non-cooperative game; packet selection strategy; ferry node

***Correspondence**
Tomoaki Ohtsuki, Department of Information and Computer Science, Keio University, Yokohama, Japan.
E-mail: ohtsuki@ics.keio.ac.jp

1. INTRODUCTION

In traditional wireless ad hoc and sensor networks, any pair of nodes communicates with each other via single hop or multi-hop mode [1,2]. As for any two nodes, if they communicate with each other by using multi-hop mode, at least one permanent connected path, which is organized by other forwarding nodes, should exist. However, in some real application scenarios, because of the limited wireless communication range or sparse deployed wireless nodes, it is difficult to guarantee that there is always an existing consistent connected path between any pair of nodes [3,4]. This results to traditional routing and forwarding protocols of wireless networks not being able to work in the sparse mobile wireless networks.

In sparse mobile wireless networks, how to guarantee the effective packet forwarding is the hot topic in research areas [5–10]. As for the moving activity of nodes distributed in sparse wireless networks, the moving activity of nodes is similar to that of human being who usually forms the special community [11]. There are some research results working on how to forward packets between isolated communities [12,13]. In some previous research results, source node prefers to select the neighbor nodes that have higher probability to reach the destination [14] or some nodes that have the higher cluster co-efficient [15]. However, there are some obvious limitations in these solutions, and the network performance is difficult to be guaranteed.

To reduce the hop counts of packet forwarding and improve the performance of sparse wireless networks, a special node is introduced to establish the connected path between source and destination nodes. The only duty of a special node is to forward a packet, and we call it a ferry node. In real application scenarios, a ferry node could be acted by a robot, a bicycle, and a vehicle. As the host of a ferry node is easy to be controlled by people, it can easily be recharged.

As the packet is forwarded by a ferry node, the behavior of a ferry significant impacts the performance of a network.

From the point of view of a single ferry work in the network, the ferry mobility control has been deeply studied [16]. Currently, how to optimize the moving trajectory of a ferry node is still the hot topic for ferry-based forwarding schemes. As for the scenario of multiple ferry nodes working together in the same network area, the main research topic has two sub-branches: cooperative and non-cooperative behaviors of multiple ferry nodes. In the cooperative behavior of ferry nodes, it means that a packet is allowed to be exchanged between ferry nodes [17]. In the non-cooperative behavior of ferry nodes, it means that there exists a competition relationship among all ferry nodes [18]. Our paper focuses on this topic.

In this paper, we consider the network scenario in which multiple ferries are working together. All the ferry nodes are rational, and their only target is how to obtain the maximum income by finishing forwarding task for normal nodes in sparse networks. We define the income of a ferry and the residual time of packets as a linear relationship. It means that if the ferry forwards the packet to its destination before the packet expires, the ferry can obtain an income. If the packet has more residual time when it reached the destination, the ferry obtains more income. Meanwhile, we set fixed expired time (TTL) for each packet to ensure the packet delivery delay and restrict the behavior of the ferry. In the scenario where multiple ferries are working together, by using the a Bayesian game [19,20], we make all ferries work more effectively. In the main sections of this paper, we make a deep analysis and conclude that the non-cooperative Bayesian game is the appropriate model for the proposed packet selection strategy. How to optimize the packet selection strategy is the main work of this paper. To the best of our knowledge, this paper is the first one working on optimizing the packet selection strategy for sparse wireless networks.

When multiple ferries distribute in the network area for forwarding packets, the mutual impact between ferries is serious, which affects the forwarding performance. Our work considers this first. We formulate the optimal packet selection strategy issue as a non-cooperative Bayesian game to make all ferries obtain the maximum income. Furthermore, our solution achieves the higher packet delivery ratio and lower delivery delay. First of all, we introduce a novel concept, hot degree to measure the packet generation ratio for each community. We calculate the hot degree for each community. As the number of nodes in each community is different, under the condition of same packet generation ratio, the more nodes in one community, the more packets would be generated in one time unit. We define the community that generated more packets in one time unit as higher hot degree, in contrast to the lower hot-degree community. Meanwhile, the ferry in the higher hot-degree community can obtain more income within a given time unit. In our work, the hot degree of community is dynamic, and it needs to be updated according to the time passing. All ferries can obtain the newest hot-degree information when they are visiting. As the rational character, a ferry prefers to travel to the higher hot-degree community to do the forwarding task. This results to more ferries grabbing the packets in a higher hot-degree community while no ferries would travel to a lower hot-degree community. Therefore, we introduce the non-cooperative Bayesian game to solve this issue.

Each packet records its own expired time (TTL) and destination location. The expired time of a packet determines the income of a ferry, and the destination location determines the moving trajectory of a ferry. Therefore, to improve the network performance in terms of packet delivery delay and successful delivery ratio, how to optimize the packet selection strategy is the core issue. When the ferry reaches some community, firstly, it would check the hot degree of destination of all candidate packets and also checks the behavior of the ferries before it reached a community. By using the non-cooperative Bayesian game, the ferry concludes the optimal packet selection strategy. Finally, we adopt the traveling salesman problem (TSP) algorithm to formulate the moving trajectory of a ferry. By optimizing the packet selection strategy for all ferries, our solution achieves a better performance in terms of packet delivery ratio and delivery delay.

The rest of this paper is organized as follows: Section 2 surveys the related work of ferry-based forwarding schemes. Section 3 formulates the problem. Section 4 introduces the big picture for a non-cooperative game. Section 5 describes the detailed information of our proposed solution. Section 6 presents the simulation results, and Section 7 concludes the paper.

2. RELATED WORK

Currently, several solutions are proposed to forward a packet between isolated communities in sparse mobile wireless networks, such as message ferry [17,21], data mule [22], and wireless agent [23]. Generally speaking, these solutions are called ferry-based forwarding schemes. In terms of applications, there are some recent projects adopting packet ferry solution to achieve an effective performance, such as the DakNet project [24], underwater monitoring by using vehicle [25], and the ZebraNet project [26].

Packet ferry issue was first proposed in [21], and the authors only considered one ferry node work in sparse mobile wireless networks. In [27], the authors proposed the solution for controlling moving behavior of a single ferry node in the predicted trajectory. In [17], the authors considered multiple ferry node works in the sparse networks and made all ferries exchange packets among each other.

In [21], a ferry node provides service to a normal node directly instead of providing service to a community. It does not consider the community scenario of social proximity. In the scenario of small number of nodes or low packet generation rate, a ferry can forward packets with low delay. However, when the number of nodes is enormous or the packet generation rate is high, the ferry cannot provide service for every node in one time. In [28], a ferry calculates the fixed trajectory by following TSP according to the parameters of a community such as the coordinate. Then a ferry moves along the concluded trajectory and forwards packets. As in different time slots, a community has different packet generation rates, and the request of a forward service is dynamic in each community. However, a ferry only moves along the concluded trajectory and neglects the dynamic request. Thus, this algorithm wastes much forwarding ability of the ferry and has very low efficiency. In this paper, we compare our proposed solution with that in [21,28], called as FIMF and HRC, respectively. Normally, only a small number of ferry nodes can achieve an effective forwarding task between normal nodes; in such a case, this is not too much additional cost for using this ferry solution. However, existing solutions still have some limitations and shortcomings. For instance, some ferry solutions only consider the scenario of stationary node [29], or only consider one ferry working in network [30], or assume that the ferry node moves according to the fixed routine. Some existing research results consider the issue of how to optimize the trajectory of a ferry [31]. However, these solutions do not consider the issue on how to optimize the packet selection strategy. Based on our detailed analysis, if the solutions do not optimize the packet selection strategy, a ferry cannot make the optimal decision on forwarding a packet. Therefore, we put our main focus on how to optimize the packet selection strategy when ferry nodes come into a community.

3. PROBLEM FORMULATION

In this paper, we adopt the MIT university campus as a network model. With regard to the active area of the people who belong to the university campus, we divide the entire campus into several sub-areas or communities. We introduce one gateway with fixed position for each community. Its mission is only to store the packets that come from other nodes and forward the packets to the ferry. We assume that the gateway has the same communication ability as a normal mobile node. As the number of communities is limited, the implementation of gateway node would not lead to too much additional cost for the entire network.

Regarding the normal mobile nodes in a community, the communication radius is r and the buffer size is b. A mobile node generates a packet to a random destination within a fixed time interval (e.g., ΔT). We set a fixed number of ferries in the network. By considering the real application scenario, we assume that the buffer of a ferry is finite and that the communication radius is r. To describe our work more clearly, we make the following constraints for our network model.

Firstly, all the packets are sent to the gateway of each community. In our work, we ignore the community-detection issues, which means that each node knows which community it belongs to. Secondly, a ferry knows which community that a packet should be forwarded to. Note that this assumption needs a unique ID for each node. Within time interval ΔT, a mobile node generates a packet and assigns a destination randomly. If the packet reaches a destination before a deadline, it is deemed as a positive transmission. Otherwise, the transmission is negative, and the packet would be erased without re-transmission. When a ferry reaches a gateway node, it communicates with the gateway node to exchange information.

4. BIG PICTURE

4.1. Definition of community hot degree

In the real application scenarios of social networks, normally, the mobile nodes belong to some specific community. Such as in the campus of a university, the mobile node carried by students normally distribute in the library, canteen, and dorm. Meanwhile, in the different time period given, the number of nodes is different in each community. This also means that within a different time period given, the number of generated packets is different in the same community. In such a case, we define the production rate of packets as the hot degree for the given community. Based on this definition, we put our focus on how to achieve the better forwarding performance under the condition that the hot degree of community is dynamically changing according to the different given time periods.

To achieve the better performance of packet delivery ratio and lower delivery delay in sparse mobile wireless networks, we implement multiple ferry nodes to help normal nodes forward packets, and also, we set some gateway nodes for each community. The gateway nodes cannot move and are located in the fixed location within each community. There are several ferries moving in the network, and their only duty is to forward packets. We assume that all the ferries are selfish and rational. In each community, there is one unique gateway, and all the nodes that belong to this community send their packets to the gateway node. As for a ferry node, they move to the gateway to collect the packets that need to be forwarded. According to the destination of packet, a ferry would move to the special gateway of the destination community. Furthermore, we assume there is no communication activity between normal node and ferry, and also, as a ferry node is selfish and rational, ferries cannot communicate with each other.

In this paper, the community hot degree is dynamic, and it needs to be updated according to the time passing. However, in distributed mobile intermittent wireless networks, a ferry node can obtain the latest hot-degree information only when they access the gateway for each community. We consider that the community hot degree is relatively stable in a given time period. In such a case, we assume that the given time period is denoted by T, it is divided into n equal time slots, and each time slot is denoted by Δt, $T = n\Delta t$. A gateway node would save the number of packets received in current and previous time periods. Under the condition of time period t passing, the total number of packets received in current and previous time periods is denoted by N. Then we define the community hot degree as Equation (1). In this equation, we set the community ID as n.

$$H_n = N/(\Delta t + t) \tag{1}$$

When a ferry node reaches a gateway of the community, it obtains the latest hot-degree information of this gateway and updates its own hot-degree information. Meanwhile, a ferry node calculates the mean value of the hot degree that it had reached. If the hot degree of some gateway nodes is less than the mean value, we call such communities as unpopular communities. On the contrary, if the hot degree of some gateway nodes is larger than the mean value, we call such communities as popular communities.

In this paper, we first propose the definition of hot degree of a community, and it is important for routing and forwarding issues in sparse mobile wireless networks. If the community has a higher hot degree, it means that in a given time unit, this community generates more packets. As for the ferry node, when it reaches such higher hot-degree community, it has more opportunity to obtain more income. All ferries are assumed to be selfish and rational. A ferry pursues the most valuable packets to achieve the highest income. As the value of a packet and the residual lifetime of a packet have a linear relationship, the most valuable packet means the longest residual lifetime of a packet. We also use this relationship to reduce the packet delivery delay. In such a case, the higher hot-degree community attracts more ferries that forward the packets with lower delivery delay within this community. However, as the number of ferries is fixed, if too many ferries come to the higher hot-degree community, it results to each ferry obtaining less income, because the number of packets is limited in one community. Meanwhile, only a few number of ferries would travel to the lower hot-degree community, as it leads to higher delivery delay and lower successful delivery ratio. Therefore, in this paper, we propose a non-cooperative game to solve these issues. As for a ferry node, the packet destination determines the trajectory of its own. We optimize the packet selection strategy by using the non-cooperative game theory. We also optimize the distribution of ferry nodes between the higher and lower hot-degree communities.

4.2. Overview for non-cooperative game

In this paper, we set multiple ferries to help normal nodes forward packets. Our main work is to optimize the activity and strategy of a ferry to achieve better performance in terms of delivery delay and successful delivery ratio by using the non-cooperative game theory.

In our work, packet delivery delay has a linear relationship with the income of a ferry; when ferry nodes finish the forwarding action and carry the packet to the destination, the ferry would obtain the income. The income is reflected by the TTL of packet. It means that the much earlier the packet is forwarded to its destination by the ferry, the more income the ferry gains. In this way, the ferry node tries to forward the packet as soon as possible to achieve more income; meanwhile, the packet delivery delay is reduced by this activity. Ferry nodes are selfish and rational, and the objective of a ferry is only to take the packets having the longest TTL to obtain the most income in a given time. In our model, we set multiple ferry nodes that are working at the same time; if some ferry nodes take too much higher valuable packets, it results to some other ferry nodes taking less packets. Meanwhile, owing to their selfish characteristic, all ferry nodes are pursuing higher income, which inevitably leads to the competitive conflict among all ferry nodes. The competitive conflict would impact the network performance, such as making the income of all ferry nodes lower and lower. How to eliminate the influence of competition is our main goal for a multiple ferry nodes scenario. By using the non-cooperative game theory, we make each ferry adopt the strategy of logical action of optimizing the packet taken. As for ferry competition, the main reason is the distribution of ferries. In this paper, we propose the non-cooperative game to formulate the packet selection strategy. As the packets in the buffer of a ferry determine the moving trajectory of a ferry, when the higher hot-degree community possesses too much ferries, some ferries would move to the lower hot-degree community to take packets by using the game relationship to avoid further competition. Owing to the distributed network scenario, some important information that can impact the income cannot be observed by ferries. It means that during the game process, a ferry lacks the necessary information of other ferries. Therefore, we introduce a Bayesian game to solve such incomplete information game to optimize the packet selection strategy and moving trajectory.

As for the game behavior of ferries, the ferries that participated in the competition have their own target or interest. To achieve their own targets, all ferries have to consider the possible activity schemes of the adversary and try to find the most logical solution to maximize their own income. Game theory is a subject that focuses on how to determine whether the optimal behavior exists among all participators and concludes the reasonable mathematical method. Therefore, by using the non-cooperative game theory, we can conclude a way to reduce the competition expense among all ferries.

Game theory is one branch of applied mathematics. Currently, it is widely used in many fields, such as biology, economics, computer science, and electronic engineering. It considers the prediction behavior and actual behavior of participators and optimizes the strategy. Normally, game theory is composed of cooperative and non-cooperative games. The difference between cooperative and non-cooperative games is whether there exists at least one bound rule for all participators. If there exists a bound rule, the game is cooperative. Otherwise, the game is non-cooperative. As for the non-cooperative game, it is the game model in which participators cannot come to the bound rule and the participators try to make the decision on how to maximize their own income under the condition of all participators having mutual influence.

In game theory, cooperative is one kind of an arithmetic zero-sum game. It means that during the gaming process, at least one participator would gain the more income, and no participator loses income; it increases the total income of the whole system. Cooperative game is working on how to allocate the society income when participators collaborate to finish one task and cooperative game also needs mutual compromise among all participators. By using the compromise way, the income of participators and also the income of society can be increased, because cooperative game can generate collaborating surplus. As for the issue of how to allocate the collaborating surplus, it is determined by the ability of all participators and the method utilized.

The main difference between cooperative and non-cooperative games is that the cooperative game emphasizes the information exchange and requires the participators to be able to form binding commitments. Information exchange is the necessary foundation, and it makes all the participators form a coalition to achieve the same target. However, whether the coalition can gain the net earnings and how to distribute the net earnings in the coalition are all determined by the contract needed to be ordered for enforcement. Therefore, the contract needed to be ordered for enforcement is the main point for the cooperative game.

In this paper, we need to solve the core issues on how to handle the competition among all participators and increase the income for all participators by using the way of mutual compromise. From this aspect, our work conforms to the cooperative game model. However, during the process of packet ferrying, it is difficult to exchange the private information of a ferry. Furthermore, in distributed networks, the coalition cannot form binding commitments and distribute the net earning. By considering these descriptions, in this paper, cooperative game is not the appropriate tool. Therefore, we use the non-cooperative game to solve the packet ferry issue for mobile intermittent wireless networks.

5. OPTIMAL STRATEGY OF A FERRY

5.1. Packet selection strategy

In sparse mobile wireless networks, there are n ferries to help normal nodes forward packets. Each ferry has the following two categories of packet selection strategy.

- Grab packets from the higher hot-degree community. In case the competition is not too fierce, as the higher hot-degree community generates more packets, it makes the ferry nodes obtain higher income. If a ferry can forward many packets in a single trip, the average packet delivery delay could be reduced significantly.
- Abandon higher hot-degree community and carry the packets from the lower hot-degree community. This strategy can make the income stable for a ferry when the competition is fierce and also can avoid the situation that the packets in the lower hot-degree community wait for too long.

We construct the following game relationship, shown in Table I. Here, θ_a, θ_b are the grabbing convictions of ferries A and B, respectively. $F(x, y)$ is the expected utility of a ferry that grabs the packets from a higher hot-degree community. $G(x, y)$ is the expected utility of a ferry that abandons the packets from a higher hot-degree community. x is the ferry's own probability of grabbing, and y is the probability of grabbing of an adversary. The grabbing behavior would result to packets in the lower hot-degree community being not forwarded. In such a case, the grabber needs to bear the risk. C is the needed cost when the grabbing behavior is existing. Because of the selfishness of ferries, they would not announce their own grabbing conviction to others. Therefore, the grabbing conviction of each ferry is incognizable for other ferries.

During the process of forwarding activity of a ferry, multiple ferries cannot reach the same gateway simultaneously. We assume that ferry A reaches the gateway first; then it makes a decision prior to node B. We formulate the aforementioned game relationship into a dynamic game.

The game relation shown in Figure 1 is the public knowledge for all ferries. In such game relation, as for a ferry node, how to select packets to reduce the cost led by the competition is the main point of our game model. Each ferry adopts its packet selection strategy according to the game relation to make the game play in the Nash equilibrium. In case all participators play in the Nash equilibrium, all participators would not deviate from the equilibrium point. Therefore, the cost led by the competition decreased to the lowest level for all ferries, and the network performance would be improved.

When node A reaches the gateway prior to node B, as there is no *a priori* knowledge to help it to make decision, therefore, $y = 0$, which is in $F(x, y)$ and $G(x, y)$ of node A. If node A had made a decision and B is reached, the knowledge of A affects the income of B. In such a case, the payoff function of game is mainly determined by the grabbing probability θ_a of A. As A knows θ_a and B does not know, we have the incomplete information game and formulate it into a Bayesian game. We introduce the Bayesian rule conviction to make the strategy play in the

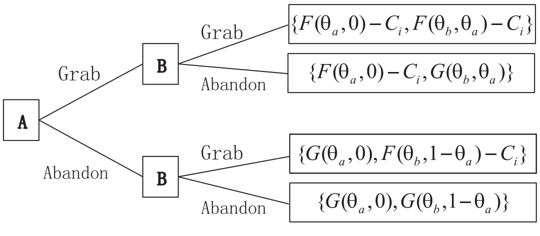

Figure 1. Formal game process for ferries.

Table I. Non-cooperative game between ferries *A* and *B*.

	B grabbing	*B* abandoning
A grabbing	$\{F(\theta_a, \theta_b) - C_i, F(\theta_b, \theta - a) - C_i\}$	$\{F(\theta_a, \theta_b) - C_i, G(\theta_b, \theta_a)\}$
A abandoning	$\{G(\theta_a, \theta_b), F(\theta_b, 1 - \theta_a) - C_i\}$	$\{G(\theta_a, 1 - \theta_b), G(\theta_b, 1 - \theta_a)\}$

Nash equilibrium, referred to also as Bayesian equilibrium; furthermore, optimize the packet selection strategy, reduce the competition cost, and improve the performance of network.

5.2. Detailed process of the Bayesian game

We formulate the previous game relation into the tetrad mode: $\langle N, (A_i), (\Theta_i), (u_i) \rangle$. $N \triangleq \{1, 2, \ldots, n\}$ is the participator set. This set contains all ferries that plan to participate in the packet forwarding task, and they are denoted by $N_1, N_2, \ldots, N_n$. $A_i \triangleq \{f_1, f_2, g_1, g_2\}$ is the feasible activity set of participator i and $\forall i \in N$.

In this paper, we define two strategies for a ferry node. We describe the detail in this section. The first strategy is to grab the packets from the higher hot-degree community, and we denote it by f. The second strategy is to abandon the higher hot-degree community and carry the packets from the lower hot-degree community, and we denote it by g.

We expand each strategy into two sub-strategies; then each participator i has four activity schemes. They are denoted by $f_1, f_2, g_1, g_2 (f_1, f_2 \in f; g_1, g_2 \in g)$.

> f_1: Only consider packets from the higher hot-degree community as a candidate.
> This strategy can make the ferry obtain more packets from multiple higher hot-degree communities and reserve the idle buffer for the hot-degree packets when the current community does not have enough packets. Moreover, this strategy makes the ferry obtain more income within the same time unit.
> f_2: Consider all packets as the candidate and treat the packets from the higher hot-degree community as the first priority. Only carry the packets from the lower hot-degree community until all the higher hot-degree packets have been carried away.
> g_1: Consider all packets as the candidate and treat the packets that are from the lower hot-degree community as the first priority. Only carry the packets from the higher hot-degree community after all the lower hot-degree packets have been carried away.
> g_2: Only consider packets from the lower hot-degree community as a candidate.

As the buffer size of a ferry is limited, when the competition is too fierce in the higher hot-degree community, this strategy can improve the expected utility.

As for the other important elements of the Bayesian game, we denote the conviction set of participators as $\Theta_i \triangleq \{\theta_1, \ldots, \theta_n\}$ and $\forall i \in N$. In this set, θ_i means to execute the conviction of $f(f_1, f_2 \in f)$ when N_i is in the same condition. The different conviction can affect the utility.

In this paper, we set the buffer size as b units for each ferry, the number of ferries as n, the mean distance between any two communities to l, and the ferry velocity to v. The forwarding efficiency of a ferry is given by

$$r_f = \frac{n \times b}{l/v} \tag{2}$$

We assume that the total packet generating efficiency is r_m. If $r_m > r_f$, it means that the carrying ability cannot support the network requirement. As time passes, more and more packets would be accumulated in a network and finally leads the result on lower successful delivery ratio and longer delivery delay. Therefore, the relation between r_f and r_m should be $r_f > r_m$, and we have $R = r_m / r_f$. If R is larger, the network needs more grabbing behavior to carry the packets. Therefore, the number of mighty ferry is directly proportional with R. Then the grabbing behavior ratio of a ferry is $R\alpha$ and $0 < \alpha < 1$. In case $R > 0.5$ and the number of ferries n is an odd number, we equally divide R to $x = (n-1)/2$ parts. Then we conclude the following type space, and it is given by

$$\begin{cases} \theta_n = R\alpha \\ \theta_i = \left[R + (-1)^i \frac{1-R}{x}\right] (i \backslash 2)\alpha \; (i = 1, \ldots, n-1) \end{cases} \tag{3}$$

If the number of ferries n is an even number, we conclude the following type space, and it is given by

$$\theta_i = \left[R + (-1)^i \frac{1-R}{x}\right] (i \backslash 2)\alpha \; (i = 1, \ldots, n) \tag{4}$$

In case $R < 0.5$, we convert Equations (3) and (4) to the following expressions.

$$\begin{cases} \theta_n = R\alpha \\ \theta_i = \left[R + (-1)^i \frac{R}{x}\right] (i \backslash 2)\alpha \; (i = 1, \ldots, n-1) \end{cases} \tag{5}$$

$$\theta_i = \left[R + (-1)^i \frac{R}{x}\right] (i \backslash 2)\alpha \; (i = 1, \ldots, n) \tag{6}$$

To reduce the level of competition, we assign a different grabbing conviction to all ferries. And the bigger the conviction value, the more income is obtained by a ferry. We let the grabbing conviction of a ferry follow the arithmetic distribution to make the activity of a ferry more flexible. When R is smaller or bigger than 0.5, and the number of ferries is odd or even, and it needs a different equation to make sure that the conviction of ferries is an arithmetic distribution. Equations (3)–(6) are used to handle each of the situations whereby R is greater or less than 0.5 and n is even or odd.

By using the preceding equations, we conclude the type set consisting of n types, and it is a public knowledge for all ferries. However, a ferry cannot know the type for other ferries. As we described in the previous section, as for one ferry, the grabbing conviction is θ_i and the abandon conviction is $\left(1-\frac{\theta_i}{\alpha}\right)\alpha$. We express these two strategies respectively by the following equations.

$$\begin{cases} \text{Prob}(f|\theta_i)=\theta_i \\ \text{Prob}(g|\theta_i)=\left(1-\frac{\theta_i}{\alpha}\right)\alpha \end{cases} \tag{7}$$

In our game model, the utility function of a participator determines the strategy directly. The target of our model is to achieve the equilibrium point. As for any participator, if it deviates from this point, the expected utility cannot be increased. It also means that by using our game model, the total income of ferries could be increased. In the aspect of a network, the delivery delay would be increased and the performance on a successful delivery ratio would be improved. In such a case, we define the utility function based on these requirements: lower delivery delay of a packet and more income of a ferry. Meanwhile, the lower the successful delivery ratio, the less income is obtained by a ferry. In this paper, when a ferry forwards a packet to the destination, it obtains an income. The income has a linear relation with residual time of a packet. If the TTL of a packet expired before it was forwarded to the destination, the ferry needs to pay the cost of losing the packet. The utility function of our model reflects the expected income in a given time unit.

$u_i(S_1,\ldots,S_n,\theta_1,\ldots,\theta_n)$ is the utility of ferry i under the situation of $S \triangleq (S_1,\ldots,S_n)$, $S_i \in A_i, i=1,\ldots,n$. Its type set is $(\theta_1,\ldots,\theta_n)$, $\theta_i \in \Theta_i$. Combine the previous utility function $u_a(S_a,S_b,\theta_a,\theta_b)$ and $u_b(S_a,S_b,\theta_a,\theta_b)$ of a game relation, and we have $(a,b \in 1,\ldots,n)$, shown by

$$\begin{cases} u_a(f,f,\theta_a,\theta_b)=F(\theta_a,0)-C_i \\ u_a(g,f,\theta_a,\theta_b)=G(\theta_a,0) \\ u_a(f,g,\theta_a,\theta_b)=F(\theta_a,0)-C_i \\ u_a(g,g,\theta_a,\theta_b)=G(\theta_a,0) \end{cases} \tag{8}$$

$$\begin{cases} u_b(f,f,\theta_a,\theta_b)=F(\theta_b,\theta_a)-C_i \\ u_b(g,f,\theta_a,\theta_b)=G(\theta_b,\theta_a) \\ u_b(f,g,\theta_a,\theta_b)=F(\theta_b,\theta_a)-C_i \\ u_b(g,g,\theta_a,\theta_b)=G(\theta_b,\theta_a) \end{cases} \tag{9}$$

As we described in the previous section, $F(x,y)$ is the expected income of a ferry that grabs packets in the higher hot-degree community and $G(x,y)$ is the expected income of a ferry that abandons the packets in the higher hot-degree community. C_i is the cost of the grabbing behavior of a ferry. We extend the utility function to multiple participators mode, which is shown by

$$u_i(S_1,\ldots,S_n,\theta_1,\ldots,\theta_n)=\begin{cases} F(\theta_i,\theta_{-i})-C_i & \text{if } S_i=f \\ G(\theta_i,\theta_{-i}) & \text{if } S_i=g \end{cases} \tag{10}$$

$\theta_{-i} \triangleq (\theta_1,\ldots,\theta_{i-1},\theta_{i+1},\ldots,\theta_n)$ denotes the type combination of $n-1$ participators besides i. m is the ferry that reached the gateway prior to ferry i, i considers θ_m as the type of m, and θ_i is the type of i. Here, $F()$ and $G()$ are expressed by

$$F(\theta_i,\theta_1,\ldots,\theta_m)=\left(\sum_{j=c_1}^{c_n}\frac{N_j\times s}{\frac{L_{c1\to j}}{v_i}}+E_{buf}-C_i\right)\times\theta_i \tag{11}$$

$$G(\theta_i,\theta_1,\ldots,\theta_m)=\left(\sum_{j=c_1}^{c_n}\frac{N_j\times s}{\frac{L_{c1\to j}}{v_i}}+E_{buf}\right)\times\left(1-\frac{\theta_i}{\alpha}\right)\alpha \tag{12}$$

s is the stable income and $c1$ to cn are the optimal order of gateways that a ferry would travel according to the TSP algorithm. N_j is the number of packets in the buffer where the destination is j. $L_{c1\to j}$ is the distance of $c1$ to j. v_i is the velocity of ferry i.

E_{buf} is the expected income in the next community and it depends on the buffer size of a ferry. In a community where the competition is fierce, as the expected income is caused by a small buffer size, a ferry would skip there and travel to the other community, which can bring the maximum utility function.

$$E_{buf}=N_{buf}\times s\times\frac{\sum\limits_{j=x}^{y} z_j}{N_f} \tag{13}$$

Here, if $\theta_i > \theta_j$, then $Z_j=1$; otherwise, $Z_j=0$. During the process of strategy adopted in community com_i, ferry A predicts which ferries would come to com_i in the future time interval T_{buf}. T_{buf} means the time to fill up the residual buffer of N_{buf} for ferry A, and it is denoted by $T_{buf}=N_{buf}/H_n$. In this paper, H_n denotes the hot degree of community n.

C_i is the punitive cost of losing packets, and it is led by a grabbing behavior.

$$C_i=\sum_{w=w_1}^{w_z}\sum_{m=wm_1}^{wm_k}\frac{s\times(1-\varphi(TTL-T_{mw}-T_{mt}))}{TTL-T_{mw}-T_{mt}} \tag{14}$$

In this paper, we denote W as the gateway set of the lower hot-degree community, $W=\{w_1,\ldots,w_z\}, W \in N$. $M_w=\{wm_1,\ldots,wm_k\}$ is the packet set of gateway w. T_{mw} is the pasted time for packet, and T_{mt} is the needed time for forwarding to the destination community. $\varphi(TTL - T_{mw} - T_{mt})$ is the probability that a new weak ferry comes to the current gateway in the future $TTL - T_{mw} - T_{mt}$ time period.

$$\varphi(TTL - T_{mw} - T_{mt}) = \varepsilon_i \times R_j \times (TTL - T_{mw} - T_{mt}) \quad (15)$$

The parameter C_i of Equation (14) means the expected punishment for the grabbing behavior of ferry i because the grabbing behavior can make some packets expire.

Equation (15) calculates the probability that a weaker ferry comes to this gateway in time point $TTL - T_{mw} - T_{mt}$. $1 - \varphi(TTL - T_{mw} - T_{mt})$ of Equation (14) denotes the probability that in time point $TTL - T_{mw} - T_{mt}$, no weaker ferry comes to this gateway. If the value of Equation (15) is less than 1, some packets in the buffer of this gateway will expire before they are forwarded to the destination. Therefore, the grabber has to be punished, and the value of punishment has a linear relationship with the value of Equation (15). If the value of Equation (15) is greater than 1, the grabber do not have to be punished. Therefore, we convert 1 instead of Equation (15) into Equation (14) if Equation (15) is greater than 1.

In the forwarding process, a gateway node would record the ID and times for the ferries that had reached this community. According to these records, we can obtain R_j and ε_i. MODIFIED: R_j is the number of times that the gateway has been visited by a ferry within time Δt. ε_i is the ratio between the number of weak ferries and that of total ferries within Δt.

5.3. Transforming incomplete information to complete but imperfect information

In our game model, the grabbing probability θ_a of ferry A is known by itself; however, other ferries cannot know this probability information. In such a case, when ferry B is trying to adopt a strategy, as it knows nothing about the type and information of ferry A, B cannot determine its own utility function, and thus, it is difficult to find the strategy. Based on the Harsanyi transformation [32], we introduce *Nature* into the game process as one participator. *Nature* valued an initial grabbing probability $\text{Prob}(\theta)$ to each ferry, and this probability is the public information. The process of the Harsanyi transformation is shown in Figure 2.

Before adopting a strategy, according to the observed behavior of ferry A, B updates its own initial grabbing probability. By using this updated grabbing probability, B concludes the more accurate utility function to adopt the strategy.

5.4. Optimal packet selection strategy for multiple ferries

By using the Harsanyi transformation for a former game relation, we convert the tetrad to the following quintuple

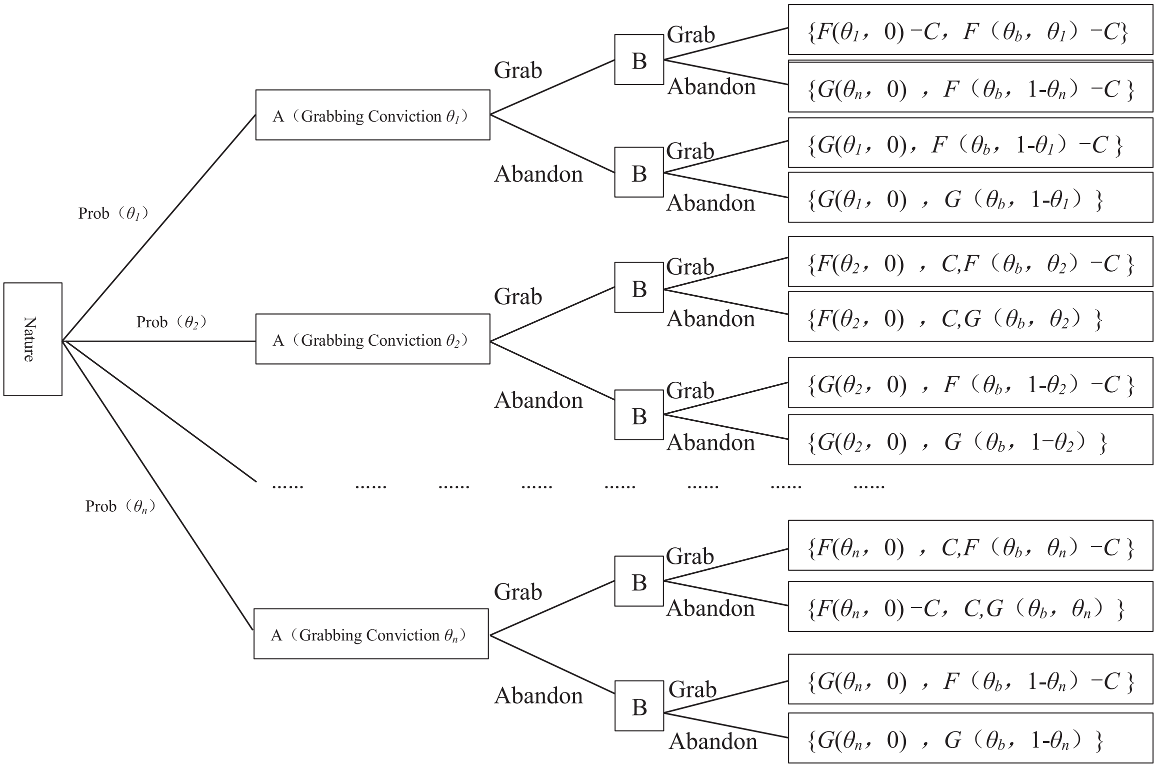

Figure 2. Transforming incomplete information to complete but imperfect information.

form: $\langle N, (A_i), (\Theta_i), (\Psi_i), (u_i)\rangle$. Here, $\Psi_i \triangleq \{P_i | P_i = \text{Prob}(\theta_i)\}$ and P_i is the probability θ_i of ferry i considered by ferry A. As set Θ_i is public knowledge, each participator has a different type. In that case, there are n ferries in the network; if the ferries have the same type, their initial value of P_i is zero; otherwise, it is $P_i = 1/n - 1$. When ferry i reaches the community, P_i is its *a priori* conviction. During the gaming process, according to the behavior of a former ferry, P_i would be updated by a Bayesian rule.

When ferry A reaches the gateway, it observes the packet selection strategy of the former ferries. In order to obtain the more accurate utility function, ferry A updates its *a priori* conviction according to the strategy of a former ferry. As for the Bayesian rule, it adopts two kinds of information. One is under the condition of which *Nature* chooses θ_i as the scenario conviction; the probability of which the ferries decide to adopt the grabbing strategy is $\text{Prob}(f|\theta)$. The other one is under the condition of which *Nature* does not choose θ_i as the scenario conviction. Based on these analysis, we conclude the grabbing probability of an adversary and denote it as $\text{Prob}(f)$ and also conclude the abandoning strategy and denote it by $\text{Prob}(g)$. They are the marginal likelihood probabilities for the strategies and are shown by the following equations.

$$\text{Prob}(f) = \sum_{i=1}^{n} \text{Prob}(f|\theta_i)\text{Prob}(\theta_i) \tag{16}$$

$$\text{Prob}(g) = \sum_{i=1}^{n} \text{Prob}(g|\theta_i)\text{Prob}(\theta_i) \tag{17}$$

In order to make $\text{Prob}(\theta_i|f)$ and $\text{Prob}(\theta_i|g)$ more clear, we conclude the following equations.

$$\text{Prob}(\theta_i|f) = \frac{\text{Prob}(f|\theta_i)\text{Prob}(\theta_i)}{\text{Prob}(f)} \tag{18}$$

$$\text{Prob}(\theta_i|g) = \frac{\text{Prob}(g|\theta_i)\text{Prob}(\theta_i)}{\text{Prob}(g)} \tag{19}$$

In our work, according to the behavior of ferries that reached the same gateway, we update the conviction for all these ferries by using a Bayesian rule. By introducing the conviction space, a ferry can use the relative complete information to guide it to take a strategy. Therefore, a ferry can obtain the maximum expected income to optimize the packet selection process, reduce the cost of competition among ferries, and improve the network performance.

When a ferry reaches the gateway of a community, it checks the information saved in the gateway and predicts the packet selection strategy of former ferries of this gateway. According to the strategy set (four packet selection strategies), the ferry determines four packet sets as its candidate. Based on the packet selection strategy, a ferry calculates the expected income for all the four packet set. The strategy that obtains the maximum income is the final decision of a ferry. According to the rule of the decided strategy, ferries select the packets and order their destination based on the TSP algorithm (Figure 3).

The preceding algorithm makes the ferry obtain the maximum expected utility function and meanwhile guarantees that each packet could be forwarded to the destination within the TTL. In this process, all ferries select their own strategies, and all these strategies form the strategy combination, denoted by $X(x_1(\theta_i), \ldots, x_n(\theta_n))$. In a Bayesian game, the existence of equilibrium means that according to the Bayesian rules, the participators refresh their convictions to achieve the Nash equilibrium status.

Lemma 1. *Our strategy combination based on the former game relation is a Bayesian equilibrium.*

Proof. (i) The strategy combination $X(x_1(\theta_i), \ldots, x_n(\theta_n))$ is formed by the strategy selected by all ferries. (ii) When each ferry observes the behavior of other ferries, it updates its own probability $p_i(\theta_{-i})$, which is for other ferries' type. (iii) The Bayesian equilibrium requires that all participators, under the given conviction, need to select their own optimal response. We convert this issue to the calculation of maximum expected utility, which is shown as follows:

$$\max_{a_i \in A_i} \{E_{p(\theta_{-i})}[u_i(a_i, a_{-i}(\theta_{-i}), \theta_1, \ldots \theta_n)]\} \tag{20}$$

□

The strategy combination $X(x_1(\theta_i), \ldots, x_n(\theta_n))$ is for each ferry to select the strategy that can make a ferry obtain the maximum expected income. Therefore, $X(x_1(\theta_i), \ldots, x_n(\theta_n))$ is the optimal solution for $\max_{a_i \in A_i} \{E_{p(\theta_{-i})}[u_i(a_i, a_{-i}(\theta_{-i}), \theta_1, \ldots \theta_n)]\}$. Formally, for all i, θ_{-i} and a_i,

$$E_{p(\theta_{-i})}[u_i(x_i, x_{-i}(\theta_{-i}), \theta_1, \ldots \theta_n) \geq E_{p(\theta_{-i})}[u_i(a_i, a_{-i}(\theta_{-i}), \theta_1, \ldots \theta_n) \tag{21}$$

Therefore, based on the preceding process, we prove that the game relation achieves a Bayesian equilibrium. It makes all ferries optimize the packet selection process by using our Bayesian game mechanism. In our work, during the process of packet forwarding, our proposed algorithm reduces the cost caused by the competition to the minimum level for all ferries and obtain the maximum income in a given time unit. It also improves the network performance on packet delivery delay and successful delivery ratio.

6. PERFORMANCE EVALUATION

In this section, we compare our proposed protocol game theory-based multiple ferry forwarding (GMFF) with the other existing classical protocols by using the Reality Mining from the MIT campus. The reason we use Reality Mining metadata is that we want to verify that the proposed protocol, GMFF, has better performance than the

```
{
        if(b2=0)
        {       if(a>=b)
                {       Put all candidate packets into buffer
                        value= The maximum value after executing the TSP algorithm
                        return value
                }
                else
                {       take a packets from candidate packets
                        for(For each packet selection way ci=c1→cn)
                        {       put the packets which is using ci way into buffer
                                vi= the maximum value after executing the TSP algorithm
                                take out the packet which are put into the buffer
                        }
                        value=max{v1,…,vn}
                        return value
                }
        }
        else
        {       if(a>b1)
                {       if(a-b1>=b2)
                        {       put all candidate packets into buffer
                                value=the maximum value after executing the TSP algorithm
                                return value
                        }
                        else
                        {       take a-b1 packets from the second- priority set
                                for(each packet selection way cj=c1→cm) each packet selection way cj=c1→cm
                                {       put the packets which is using cj way and packets from the first-priority set into buffer
                                        vj= the maximum value after executing the TSP algorithm
                                        take out the packet which are put into the buffer
                                }
                                value=max{v1,…,vm}
                                return value
                        }
                else
                {       take a packets from the first- priority set
                        for(each packet selection way ck=c1→cz)
                        {       put the packets which is using ck way into buffer
                                vk= the maximum value after executing the TSP algorithm
                                take out the packet which are put into the buffer
                        }
                        value=max{v1,…,vz}
                        return value
                }
        }
}
```

Figure 3. Algorithm process of packet selection strategy.

other protocols in real community networks. The simulation map is shown in Figure 4. We use the ONE as the simulation platform. The two compared forwarding protocols refer to those in [16,23], and they are called FIMF and HRC, respectively. As FIMF does not consider the scenario of more than one ferry work at the same time, when one ferry cannot undertake a large number of packets, we adopt multiple ferries in the network. Each ferry works without consideration of the influence of other ferries. Therefore, in a performance evaluation section, FIMF has the same number of ferries, and each ferry follows its own forwarding rule.

6.1. Reality Mining data analysis

The Reality Mining project was conducted by MIT Media Laboratory. It consists of 94 people using mobile phones pre-installed with several pieces of software. The pre-installed software would record and send the researcher

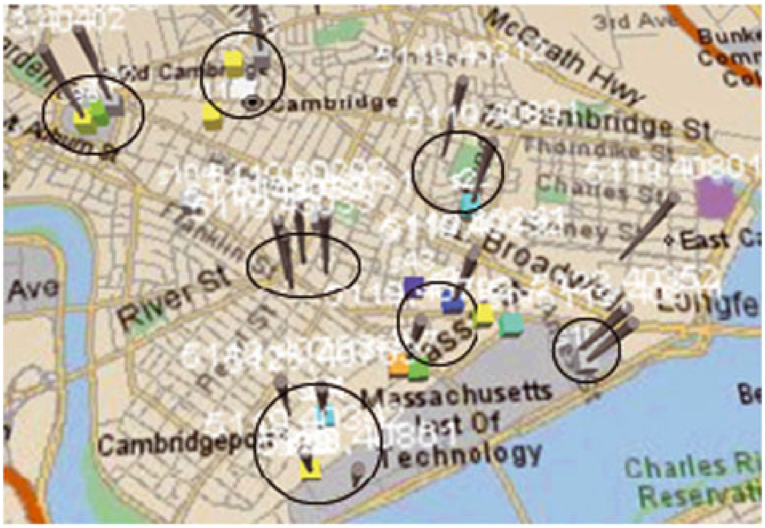

Figure 4. Map of MIT campus for Reality Mining.

data about call logs, Bluetooth devices in proximity of approximately 5 m, cell tower IDs, application usage, and phone status.

Our main focus is working on the scenario of which the nodes belong to different communities and how to make the ferries forward packets more effectively. Therefore, we filter the encountering time and ID of cellular towers recorded by subjects between all cellular towers and subjects. Through the location of cellular towers, we can obtain the location information of subjects. Unfortunately, in Reality Mining, only the cellular towers that are associated with MIT have location information. Therefore, we choose the areas covered by the cellular towers, and these areas' location information are known in our work. In Figure 4, the areas marked by circles are for the simulation.

6.2. Simulation setting

In MIT Reality Mining, our simulation runs 24 h to collect data. We choose 50 subjects that have complete information and typical community character as the normal mobile nodes among all 94 subjects. According to the locations of cellular towers in the campus map, we divided the whole map into seven communities. By using the information of contacts with cellular towers of Reality Mining, we conclude the belonging relation between subjects and community in the given time period.

In our simulations, we set one gateway for each community, and the buffer size of a gateway has no limitation. We put five ferries working in the network to forward packets between seven isolated communities. According to Reality Mining from MIT, almost all the people gathered in seven fixed areas. Therefore, we employ these seven areas to denote seven communities. Because of the limitation of community number and geographic region, which we found during the simulation, if the number of ferries is less than 3, ferries stay in the work state without competition. On the other hand, if the number of ferries is more than 7, some ferries would stay in the idle state. Also, as for our proposed solution, when the number of ferries exceeds 5, the performance improvement is not obvious. In such a case, we employ the mean value 5 as the ferry number. As the moving activity in a community is unknown to us, we assume that the normal node adopts the random waypoint model as its moving rule. When normal nodes encounter the gateway, they send their packets to the gateway. If some nodes move to other communities, they execute the same packet sending action after determining which community they belong. The packets are generated randomly, and also, the destinations of packets are also randomly assigned. We set the buffer size as the variable, and the buffer size is three and five packet units. The total simulation time is 24 h (86 900 s). We set the initial 400 s as the warming up time period and the last 500 s as the terminal time period. In these two special time periods, we do not collect data for final simulation results. The detailed simulation parameters are as follows: the velocity of a normal node is 5 m/s; the velocity of a ferry node is 15 m/s; the communication range of normal node, ferry, and gateway is 150 m; the expired time (TTL) of a packet is 60 min; and the packet generation time interval is expanded from 20 to 200 s. As for the normal node, we set the velocity of a normal node as faster than people who live in the real scenario. The main reason is that we want to describe the high topology change within a community. As for the velocity of a ferry, as we assume that the only task of a ferry is to forward a packet among communities, we assume that the velocity of a ferry is similar with the velocity of a bicycle. In the simulation section, 5 m/s is the upper bound velocity of a node and 15 m/s is the upper bound velocity of a ferry node. The simulator randomly set the velocity for ferry and normal nodes from intervals of 1–15 and 1–5 m/s periodically.

6.3. Simulation results

In Figures 5 and 6, we show the successful packet delivery ratio and average delivery delay of the three compared algorithms where the buffer size is five units. In Figure 6, as time passes, the successful delivery ratio of the three algorithms all increases. We can see that compared with FIMF, our proposed GMFF and HRC are all better in all time intervals. When the packet generation time interval is less than 100 s, our proposed GMFF has significantly better performance; when the time interval is larger than 100 s, they are in the same level for successful delivery ratio. However, when time interval is larger than 100 s, from Figure 5, we can see that in the aspect of packet delivery delay, HRC has larger delay, which is around 300 s. Also from Figure 5, we can see that only when packet generation time interval is less than 40 s, compared with other two algorithms, that GMFF has a higher delivery delay. However, as the time interval is less than 40 s, the successful delivery ratio of the two algorithms has declined to lower than 75%, and only our proposed GMFF keeps the delivery ratio larger than 88%.

In our proposed GMFF, the algorithm selects packets according to the non-cooperative game and optimizes the trajectory of a ferry by using the TSP algorithm. Therefore, the efficiency of GMFF is definitely higher than that of HRC, as the moving trajectory of a ferry in HRC is according to the fixed routine. In such a case, when the packet generation time interval is small, GMFF has a higher successful delivery ratio than HRC. In addition, GMFF has a lower packet delivery delay compared with HRC. As for the FIMF, as it does not consider the normal assembly in some specific communities, the ferry provides services to a normal node directly.

According to the normal understanding of a simulation, packet delivery delay should change with the packet generation time interval, the time interval is reduced, and delivery delay is increased. As when we count up the average delivery delay, only the successful transmission packets could be counted within the TTL. When time interval is less than 80 s, according to time passing, the time interval declines, and the successful delivery ratio of HRC and FIMF significantly declined. It means that the higher frequent packet generation would cause the congestion for networks, and the system would discard the packets that have exhausted the expired time. Therefore, when packet generation time interval is less than 80 s, HRC and FIMF have the lower average packet delivery delay.

In Figures 7 and 8, we show the successful packet delivery ratio and average delivery delay of the three compared

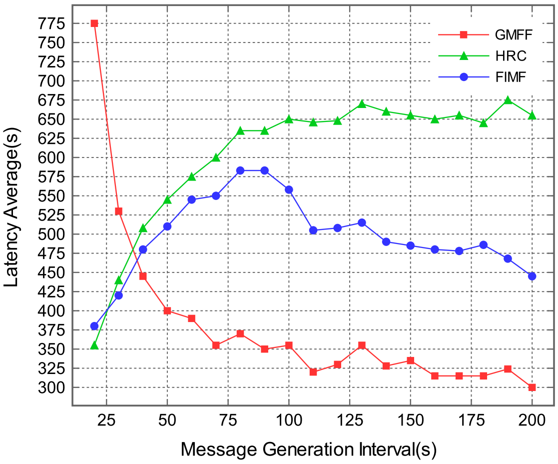

Figure 5. Illustration of average packet delivery delay in which the buffer size of a ferry is 5.

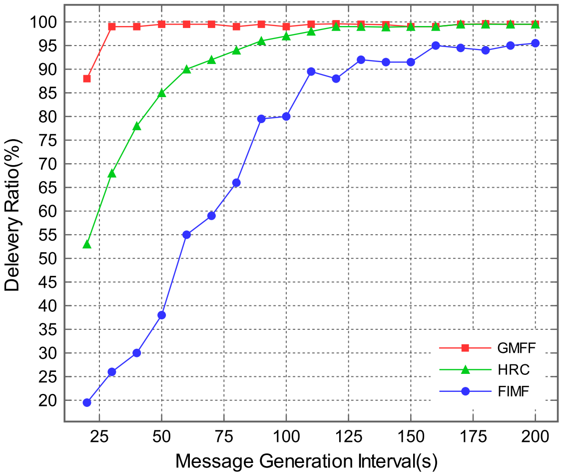

Figure 6. Illustration of successful packet delivery ratio in which the buffer size of a ferry is 5.

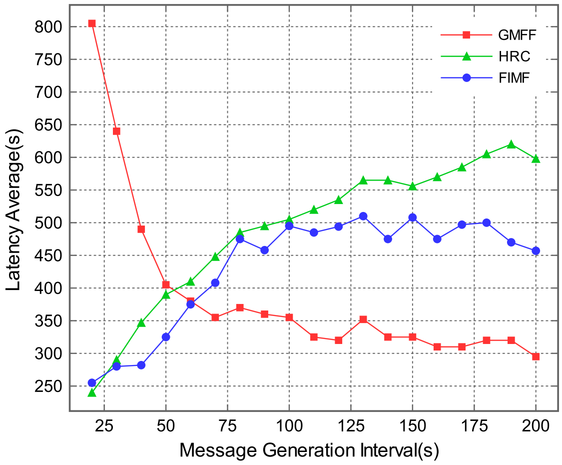

Figure 7. Illustration of average packet delivery delay in which the buffer size of a ferry is 3.

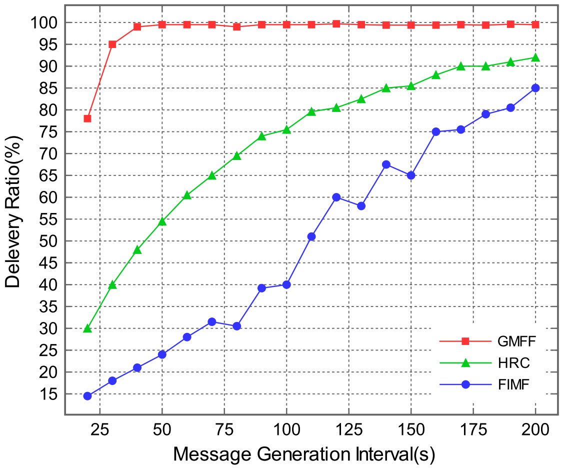

Figure 8. Illustration of successful packet delivery ratio in which the buffer size of a ferry is 3.

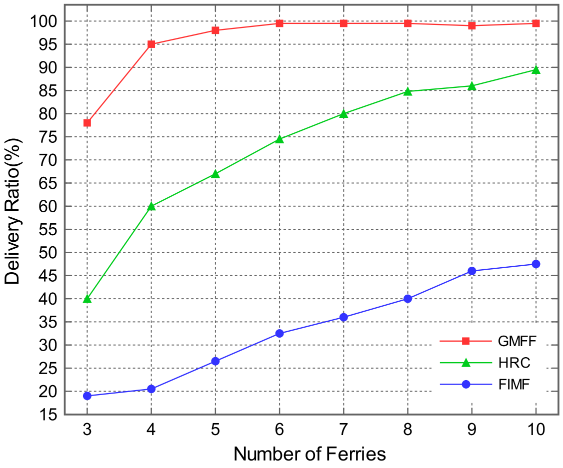

Figure 9. Illustration of successful packet delivery ratio versus the number of ferry nodes when the packet generation time interval is 30 s.

algorithms where the buffer size is three units. By comparing Figure 5 with Figure 6, according to the decreasing buffer size, the successful packet delivery ratio of HRC and FIMF significantly declined, while our proposed algorithm GMFF would be affected only when the time interval is less than 40 s. We can also see that when time interval is larger than 50 s, under the same higher successful delivery ratio, the delivery delay of GMFF has not been affected by the decrease of buffer size. Owing to the network congestion, the delivery delay is decreased.

In order to verify our proposed solution comprehensively, we compare our proposed solution with two other ferry-based solutions under the condition of packet generation time interval of 30 s. The number of ferry nodes changes from 3 to 10. In Figure 9, we show the compared performance in terms of packet delivery ratio according to the change of number of ferry nodes. In this figure, we can see that our proposed solution performs better compared with the other two solutions. When the number of ferry nodes is 6, the performance in terms of packet delivery ratio is in the saturated level, and it means that even when the number of ferry nodes is increasing, the packet delivery ratio cannot be improved. On the other hand, the other two compared ferry-based solutions are still in the lower level even when the number of ferry nodes is 10. In such a case, we know that compared with the other two solutions, our proposed solution only needs less number of ferry nodes.

In the aforementioned figures of simulation results, we set the buffer size of a ferry as fixed, three or five units. In Figures 10 and 11, we compare our proposed solution with the other two ferry-based solutions under the condition of a buffer size increasing from 10 to 50 units. In these two figures, we show the performance in terms of

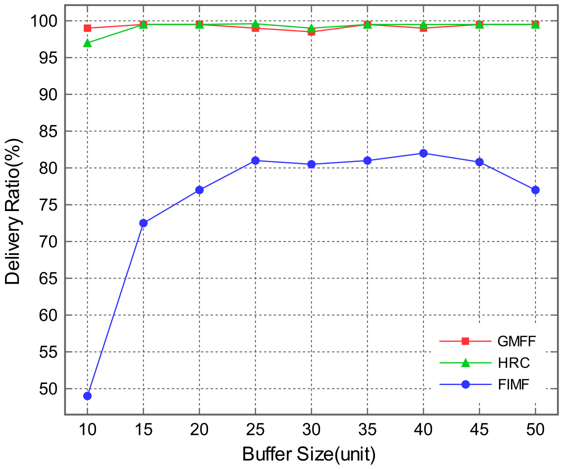

Figure 10. Illustration of successful packet delivery ratio versus buffer size change of a ferry node when the packet generation time interval is 30 s.

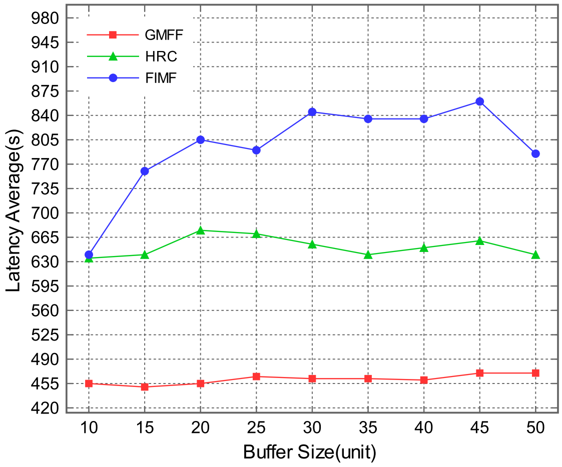

Figure 11. Illustration of average packet delivery delay versus buffer size change of a ferry when the packet generation time interval is 30 s.

packet delivery ratio and delivery delay. As for HRC, a ferry node moves according to the fixed trajectory. In this way, if we increase the buffer size, we would obtain a similar result with increase in the number of ferry nodes. In Figure 10, we can see that packet delivery ratio increases with the increase of buffer size from 10 to 50 units. In Figure 11, we can see that for HRC, as the trajectory of a ferry node is fixed, even though the number of ferry nodes or the buffer size of a node is increased, the average delivery delay cannot be improved. As for FIMF, when a ferry node provides forwarding service for a normal node, it considers the buffer residual buffer units to make a forwarding decision. In such a case, if we increase the buffer size of a ferry node, the waiting time for forwarding service would be prolonged. Therefore, according to the increasing buffer size, packet delivery delay of FIMF would be increased. In Figure 10, when the buffer size of a ferry node reaches 25 units, the packet delivery ratio cannot be improved. In our proposed solution, when buffer size is five units, packet delivery ratio has reached the saturated level. Even though the buffer size is more than 20 units, HRC has a similar packet delivery ratio with our proposed solution. However, in Figure 11, our proposed solution shows better performance in terms of packet delivery delay compared with HRC.

We analyzed the overhead data of all the three protocols in the simulation. In the analysis results, we found that all the three protocols have almost stable overhead ratio. Furthermore, the overhead ratio mainly varies with the delivery ratio. The reason is that all of the three protocols only use a ferry to forward packets. The proposed protocol uses ferry forward packets for communities along the optimal trajectory. The HRC uses ferry forward packets for communities along the fixed trajectory. The FIMF uses ferry forward packets for every node when they buffered a certain number of packets. Thus, each packet is delivered through two hops. Therefore, the network does not have other redundant data besides cost by a ferry. With the consideration of the aforementioned situation, we did not show the comparison of overhead ratio in this paper.

7. CONCLUSION AND FUTURE WORKS

In this paper, we analyzed the activity behaviors of multiple ferries working in the same network scenario to forward packets between the isolated communities. We found that as for a ferry node, how to optimize the packet selection strategy is a core issue for improving network performance in terms of packet delivery delay and successful delivery ratio. We introduce the non-cooperative Bayesian game to the packet selection strategy for ferries to conclude the optimal strategy and prove that the proposed algorithm can achieve the Bayesian–Nash equilibrium. To the best of our knowledge, this is the first research result working on how to optimize the packet selection strategy for sparse wireless networks. As for the moving trajectory of a ferry, we adopted the TSP algorithm to optimize the moving behavior to achieve better performance for networks. By using the simulation results, we prove that our proposed algorithm performs better than other existing schemes in terms of delivery delay and delivery ratio.

Our proposed solution is designed to deal with the situation that a single ferry is not enough to forward the large number of packets. As for the disadvantage of our solution, we use a game theory to handle the competition among multiple ferries. However, a network does not always stay in the busy state; it also has an idle state when there are few packets that need to be forwarded. In the idle state of a network, some ferries are unnecessary and waste much energy. In our future work, we will employ a dynamic ferry system. We adjust the number of ferries according to the amount of packets and communities to save the unnecessary waste of energy. From the aspect of cost, we set one gateway point for each community. It is the additional cost to reduce the packet delivery delay within a community. In our future work, we will try to use a normal node to be the packet collection point to save this additional cost. From the aspect of energy consumption, as multiple ferries work in the network area, it leads some additional energy consumption. In our future, we will propose a novel algorithm to optimize the number of ferries to save energy.

REFERENCES

1. Chen M, Leung V, Mao S, Xiao Y, Chlamtac I. Hybrid geographical routing for flexible energy-delay trade-offs. *IEEE Transactions on Vehicular Technology* 2009; **58**(9): 4976–4988.
2. Zhang Y, Yu R, Yao W, Xie S, Xiao Y, Guizani M. Home M2M networks: architectures, standards, and QoS improvement. *IEEE Communications Magazine* 2011; **49**(4): 44–52.
3. Gao W, Li Q, Zhao B, Cao G. Social-aware multicast in disruption tolerant networks. *IEEE/ACM Transactions on Networking* 2012; **20**(5): 1553–1566.
4. Gao W, Cao G, La Porta T, Han J. On exploiting transient social contact patterns for data forwarding in delay tolerant networks. *IEEE Transactions on Mobile Computing* 2013; **12**(1): 151–165.
5. Liu C, Wu J. On multi-copy opportunistic forwarding protocols in nondeterministic delay tolerant networks. *IEEE Transactions on Parallel and Distributed Systems* 2012; **23**(6): 1121–1128.
6. Wu J, Wang Y. Social feature-based multi-path routing in delay tolerant networks, In *Proceedings of the 31st IEEE International Conference on Computer Communications (IEEE INFOCOM 2012)*, Orlando, FL, USA, March 2012; 1368–1376.
7. Liu C, Wu J. An optimal probabilistic forwarding protocol in delay tolerant networks, In *Proceedings of MobiHoc*, New York, NY, USA, 2009; 105–114.

8. Xiao Y, Zhang Y. Divide- and conquer-based surveillance framework using robots, sensor nodes, and RFID tags. *(Wiley) Wireless Communications and Mobile Computing (WCMC), John Wiley & Sons, special issue Emerging Techniques for Wireless Vehicular Communications* 2011; **11**(7): 964–979. DOI: 10.1002/wcm.863.
9. Xiao Y, Zhang Y. A critical line based boundary surveillance strategy in wireless sensor networks. *Telecommunication Systems Journal (Springer), Special issue on: Challenges in Next-Generation and Resource-Constrained Networks* 2013; **52**(2): 423–434. DOI: 10.1007/s11235-011-9453-0.
10. Xiao Y, Zhang Y, Liang X. Primate-inspired communication methods for mobile and static sensors and RFID tags. *ACM Transactions on Autonomous and Adaptive Systems*; **6**(4, Article 26): 2011. DOI: 10.1145/2019591.2019595.
11. Hui P, Crowcroft J, Yoneki E. BUBBLE rap: social-based forwarding in delay-tolerant networks. *IEEE Transaction on Mobile Computing* 2011; **10**(11): 1576–1589.
12. Henkel D, Brown TX. Towards autonomous data ferry route design through reinforcement learning, In *Proceeding of IEEE-WoWMoM*, Newport Beach, CA, 2008; 1–6.
13. Li Y, Hui P, Jin D, Su L, Zeng L. Performance evaluation of routing schemes for energy-constrained delay tolerant networks, In *Proceeding of IEEE ICC 2011*, Kyoto, Japan, June 2011; 1–5.
14. Yuan Q, Cardei I, Wu J. Predict and relay: an efficient routing in disruption-tolerant networks, In *Proceedings of ACM MobiHoc'09*, New Orleans, Louisiana, USA, 2009; 95–104.
15. Daly E, Haahr M. Social network analysis for routing in disconnected delay-tolerant MANETs, In *Proceedings of ACM MobiHoc'07*, Montreal, Quebec, Canada, 2007; 32–40.
16. He T, Swami A, Lee KW. Dispatch-and-search: dynamic multi-ferry control in partitioned mobile networks, In *Proceedings of MobiHoc'11*, Paris, France, 2011; 188–197.
17. Zhao W, Ammar M, Zegura E. Controlling the mobility of multiple data transport ferries in a delay-tolerant network, In *IEEE-INFOCOM*, Miami, FL, USA, 2005; 1407–1418.
18. Estanjini RM, Wang J, Paschalidis IC. Scheduling mobile nodes for cooperative data transport in sensor networks. *IEEE/ACM Transactions on Networking* 2013; **21**(3): 974–989.
19. Liang X, Xiao Y. Studying bio-inspired coalition formation of robots for detecting intrusions using game theory. *IEEE Transactions on Systems, Man, and Cybernetics, Part B, Special Issue on Game Theory* 2010; **40**(3): 683–693. DOI: 10.1109/TSMCB.2009.2034976.
20. Liang X, Xiao Y. Game theory for network security. *IEEE Communications Surveys & Tutorials* 2013; **15**(1): 472–486. DOI: 10.1109/SURV.2012.060912.00182.
21. Zhao W, Ammar M, Zegura E. A message ferrying approach for data delivery in sparse mobile ad hoc networks, In *Proceedings of ACM MobiHoc'04*, Roppongi, Japan, 2004; 187–198.
22. Shah R, Roy S, Jain S, Brunette W. Data MULEs: modeling a three-tier architecture for sparse sensor networks. *Ad Hoc Networks* 2003; **1**(2): 215–233.
23. Saad W, Han Z, Basar T, Debbah M, Hjerungnes A. Hedonic coalition formation for distributed task allocation among wireless agents. *IEEE Transactions on Mobile Computing* 2011; **10**(9): 1327–1344.
24. Pentland A, Fletcher R, Hasson A. DakNet: rethinking connectivity in developing nations. *Computer* 2004; **37**(1): 78–83.
25. Dunbabin M, Corke P, Vasilescu I, Rus D. Data muling over underwater wireless sensor networks using an autonomous underwater vehicle, In *Proceedings of IEEE ICRA*, Orlando, FL, USA, 2006; 2091–2098.
26. Juang P, Oki H, Wang Y, Martonosi M, Peh L, Rubenstein D. Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with Zebranet, In *Proceedings of International Conference on Architectural Support for Programming Languages and Operating Systems*, San Jose, California, USA, 2002; 96–107.
27. Chen Y, Yang J, Zhao W, Ammar M, Zegura E. Multicasting in sparse MANETs using message ferrying, In *Proceedings of Wireless Communications and Networking Conference*, Las Vegas, NV, USA, 2006; 691–696.
28. Chen Y, Zhao W, Ammar M, Zegura E. Hybrid routing in clustered DTNs with message ferrying, In *Proceeding of ACM-MobiOpp'07*, San Juan, Puerto Rico, USA, June, 2007.
29. He T, Lee KW, Swami A. Flying in the dark: controlling autonomous data ferries with partial observations, In *Proceedings of MobiHoc'10*, Chicago, Illinois, USA, 2010; 141–150.
30. Mansy A, Ammar M, Zegura E. Deficit round robin based message ferry routing, In *IEEE-GLOBECOM'2011*, Houston, USA, 2011; 1–5.
31. Guan X, Chen M, Chen H, Liu C, Ohtsuki T. Trajectory optimization of packet ferries in sparse mobile social networks, In *Proceeding of IEEE-GLOBECOM'2011*, Houston, TX, USA, 2011; 1–5.
32. Binmore K. *Playing for Real: A Text on Game Theory*. Oxford University Press: US, 2007.

AUTHORS' BIOGRAPHIES

Xin Guan received his bachelor degree in School of Computer Science and Technology from Heilongjiang University, China, in 2001 and his master degree from Harbin Institute of Technology, China, in 2007. In 2012, he received his PhD degree from the Graduate School of Science and Technology, Keio University, Japan. His research interests include topology control, performance evaluation, and routing algorithm in wireless networks. Currently, he is an associate professor in Heilongjiang University, China. He was the technical committee member of IEEE-GLOBECOM, PIMRC, HPCC, IWCMC, and so on. He served as a reviewer in *IEEE-NETWORK, KSII Transactions on Internet and Information Systems* (TIIS), *EURASIP Journal on Wireless Communications and Networking, Security and Communication Networks* (Wiley), *International Journal of Ad Hoc and Ubiquitous Computing, ACM/Springer Mobile Networks & Applications* (MONET), and so on. He is a member of the IEEE.

Min Chen (minchen@ieee.org) [M'08-SM'09] is a professor in School of Computer Science and Technology at Huazhong University of Science and Technology. He was an assistant professor in the School of Computer Science and Engineering at Seoul National University (SNU) from Sep 2009 to Feb 2012. He worked as a Post-Doctoral Fellow in Department of Electrical and Computer Engineering at the University of British Columbia (UBC) for 3 years. Before joining UBC, he was a Post-Doctoral Fellow at SNU for 1.5 years. He has more than 170 paper publications. He received Best Paper Award from IEEE ICC 2012 and Best Paper Runner-up Award from QShine 2008. He is a Guest Editor for *IEEE Network, IEEE Wireless Communications*, and so on. He is Symposium Co-Chair for IEEE ICC 2012 and IEEE ICC 2013. He is General Co-Chair for IEEE CIT 2012. He is a TPC member for IEEE INFOCOM 2014. He is Keynote Speaker for CyberC 2012 and Mobiquitous 2012. He is an IEEE Senior Member since 2009 and has received the Centennial Award.

Tomoaki Ohtsuki received his BE, ME, and PhD degrees in Electrical Engineering from Keio University, Yokohama, Japan, in 1990, 1992, and 1994, respectively. From 1994 to 1995, he was a Post Doctoral Fellow and a Visiting Researcher in Electrical Engineering at Keio University. From 1993 to 1995, he was a Special Researcher of Fellowships of the Japan Society for the Promotion of Science for Japanese Junior Scientists. From 1995 to 2005, he was with Science University of Tokyo. In 2005, he joined Keio University. He is now a Professor at Keio University. From 1998 to 1999, he was with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley. He is engaged in research on wireless communications, optical communications, signal processing, and information theory. Dr. Ohtsuki is a recipient of the 1997 Inoue Research Award for Young Scientist, the 1997 Hiroshi Ando Memorial Young Engineering Award, Ericsson Young Scientist Award 2000, 2002 Funai Information and Science Award for Young Scientist, IEEE the 1st Asia-Pacific Young Researcher Award 2001, the 5th International Communication Foundation (ICF) Research Award, 2011 IEEE SPCE Outstanding Service Award, the 28th TELECOM System Technology Award, and ETRI Journal's 2012 Best Reviewer Award.

He has published more than 125 journal papers and 285 international conference papers.

He served as a Chair of IEEE Communications Society, Signal Processing for Communications and Electronics Technical Committee. He served as a technical editor of the *IEEE Wireless Communications* magazine. He is now serving as an editor of the IEEE Communications Surveys and Tutorials, IEICE Communications Express, and Elsevier Physical Communications. He has served as symposium co-chair of many conferences, including IEEE GLOBECOM 2008, SPC, IEEE ICC2011, CTS, and IEEE GCOM2012, SPC. He gave tutorials and keynote speech at many international conferences including IEEE VTC and IEEE PIMRC. He is a senior member of the IEEE and the IEICE.