CAMF: Context-Aware Message Forwarding in Mobile Social Networks

メタデータ	言語: eng 出版者: IEEE 公開日: 2017-10-06 キーワード (Ja): キーワード (En): mobile social networks, forwarding/receiving capability, knapsack problem, forwarding profit maximization 作成者: WEI, Kaimin, 董, 冕雄, 太田, 香, XU, Ke メールアドレス: 所属:
URL	http://hdl.handle.net/10258/00009478

CAMF: Context-Aware Message Forwarding in Mobile Social Networks

Kaimin Wei, Mianxiong Dong, *Member, IEEE,* Kaoru Ota, *Member, IEEE,* and Ke Xu

Abstract—In mobile social networks (MSN), with the aim of conserving limited resources, egotistic nodes might refuse to forward messages for other nodes. Different from previous work which mainly focuses on promoting cooperation between selfish nodes, we consider it from a more pragmatic perspective in this paper. Be specific, we regard selfishness as a native attribute of a system and allow nodes to exhibit selfish behavior in the process of message forwarding. Apparently, selfishness has a profound influence on routing efficiency, and thus novel mechanisms are necessary to improve routing performance when self-centered nodes are considered. We first put forward a stateless approach to measure encounter opportunities between nodes, and represent forwarding capabilities of nodes by combining the acquired encounter opportunities with node selfishness. We then quantify receiving capabilities of nodes based on their available buffer size and energy. Taking both forwarding and receiving capabilities into account, we finally present a forwarding set mechanism, which could be deduced to a multiple knapsack problem to maximize the forwarding profit. Consequently, we take all the above studies into the design of a context-aware message forwarding algorithm (CAMF). Extensive trace-driven simulations show that CAMF outperforms other existing algorithms greatly. In fact, it achieves a surprisingly high routing performance while consumes low transmission cost and resource in MSN.

Index Terms—mobile social networks, forwarding/receiving capability, knapsack problem, forwarding profit maximization

1 Introduction

Mobile Social Networks (MSN) [1]–[3] is a class of networks in which wireless mobile user of similar interest or commonalities cooperate to establish network connectivity and communicate with each other in the absence of network infrastructure [4]. In such networks, nodes are required to forward messages in a cooperative and selfless way. However, some or all nodes may exhibit various degrees of forwarding willingness (or selfishness), especially when nodes are constrained with battery power and storage space, e.g., a node may refuse to accept and transmit messages for others to conserve limited buffer and energy [5]–[7]. Most researchers argue that this egocentric behavior is harmful for routing efficiency, and pay attention to design incentive mechanisms to promote cooperation between selfish nodes to improve routing efficiency [8]–[12].

However, selfishness is an innate characteristic of human and thus should be given more attention. Moreover, selfish behavior can reduce the total number of message forwarding to effectively conserve limited resources, which are of great importance for resource-constrained MSN. In this paper, we consider the node selfishness from a more pragmatic perspective. Be specific, we regard selfishness as a native attribute of a system and allow nodes to behave selfish behavior in the process of message transmission. Nevertheless, node selfishness has a profound influence on routing efficiency, and will cause a reduced routing performance. Thus, our goal is to improve routing efficiency when nodes are allowed to display selfish behavior.

Social-aware forwarding algorithms [13] [14] have shown their superiority on predicting encounter opportunities between nodes. However, they greatly rely on the state information of an entire network, which always need to take a long-term collecting process and consume a large storage space. These are challenges for MSN. To evaluate encounter opportunities without the collection of network state information, we put forward a stateless approach, in which nodes are described by property profiles, and their similarities are then measured to predict future encounter opportunities. Subsequently, we evaluate forwarding capabilities of nodes by combining the obtained encounter opportunities with node selfishness.

Moreover, we also pay attention to node's receiving capabilities, which are often overlooked in the previous routing protocols. They greedily transmit messages to the encountered node until there is no message left. This will cause some messages are delivered to recipients without any available storage space, and thus these messages will be dropped. To avoid such a situation, we define two concepts, called *reserved buffer size* and *reserved energy*, to denote how many available resources a node can offer to incoming messages, and then adopt them to evaluate receiving capabilities of nodes before

- *K. Wei and K. Xu are with State Key Lab of Software Development Environment, Beihang University, China. Email: {weikaimin, kexu}@nlsde.buaa.edu.cn*
- *M. Dong is with National Institute of Information and Communications Technology, Japan. Email: mx.dong@nict.go.jp*
- *K. Ota is with Department of Information and Electronic Engineering, Muroran Institute of Technology, Japan. Email: ota@csse.muroran-it.ac.jp*

transmitting messages.

Furthermore, we also consider the forwarding set problem to determine which messages are suitable for forwarding. When an encounter opportunity emerges, most MSN routing protocols randomly choose messages from a local buffer and then send them out. However, a receiver may have different forwarding capabilities for various destinations, and thus a random strategy cannot guarantee that all messages are delivered to nodes with an incremental forwarding capability. To prevent this, we present a forwarding set mechanism, which takes forwarding and receiving capabilities of nodes into account and then models the forwarding set problem as a $0/1$ multiple knapsack problem to maximize the sum of forwarding profits in each connection opportunity. Based on the above studies, we develop a context-aware message forwarding algorithm (CAMF), in which messages are transmitted to nodes that not only have an incremental forwarding capability but also poss sufficient receiving capability to serve incoming data. Simulation results show that CAMF achieves a good routing performance with low transmission cost when nodes are permitted to exhibit selfish behavior.

The rest of this paper is structured as follows. After investigating related work in Section 2, we give problem formulation and models in Section 3. Then, we introduce an overview of CAMF in Section 4, following by stating detailed design in Section 5. We also validate the effectiveness of CAMF in Section 6. Finally, we give a brief conclusion in Section 7.

2 Related Work

2.1 Selfishness Behavior in MSN

Previous works on egocentric behavior of nodes focus on designing an incentive mechanism to explicitly or implicitly stimulate selfish nodes to forward message for all others. These mechanisms can be generally classified into two categories: reputation-based schemes [8], [9] and credit-based approaches [11], [12]. In reputation-based schemes, nodes that serve others possess good reputations and receive appropriate services from other nodes. In addition, nodes collectively detect selfish nodes throughout the whole network and broadcast their bad reputations. Eventually, this propagation will lead other nodes to avoid these selfish members in the process of message transmission. In credit-based ways, nodes pay for services provided by others, and get paid for providing services to others. Meanwhile, a digital cash system is implemented in order to encourage cooperation behavior among nodes. Although selfishness is harmful for routing efficiency, it is an innate social tie of humanity which should be given more attention. At present, little work has been done to improve routing performance when selfish nodes are considered.

2.2 Social-aware Forwarding Algorithms

Social-aware forwarding algorithms have shown their superiority on predicting encounter opportunities. Most of them consider the trajectory and/or the contact history of mobile nodes, and then utilize collected state information to predict connection chances between nodes. For example, network topology information is adopted to calculate the betweenness of nodes in [14]; neighbor information is employed to measure the importance of nodes in [13]; node location knowledge [15] is used to the distance between source and destination. Nevertheless, such state information is dynamic and difficult to be acquired without a global and/or long-term collection process. Moreover, it should take a lot of storage space to buffer such state information. However, these are challenges for MSN, especially when nodes have limited resources. Fortunately, a stateless approach [16] has recently been proposed to detect encounter opportunities without a long collection process of state information. This stateless way utilizes an interest profile to describe nodes. However, the interest profile cannot completely figure nodes. In this paper, we adopt a node property profile, not limited to interest, to represent nodes.

2.3 Knapsack Problem

The knapsack problem is a well-known combinatorial optimization problem. Given a set of items, each with a mass and a value, it determines the number of each item to be included in a collection so that the total weight is less than or equal to a given limit and the total value is as large as possible. The most typical one is the $0/1$ knapsack problem, which restricts the number of copies of each kind of item to zero or one. One of its variations is the $0/1$ multiple knapsack problem where there are multiple knapsacks. This problem has been studied [17] extensively in the past.

3 Problem Formulation and Models

3.1 Problem Formulation

To improve routing efficiency in MSN where selfishness is considered, we formulate the message forwarding process as three sub-problems, listed below.

Problem 1. *To evaluate encounter opportunities, social-based routings always take a long-term process of collecting state information and need a large storage space to buffer them, which are challenges for MSN. How to evaluate encounter opportunities of nodes without gathering any state information?*

Problem 2. *In MSN, nodes are resource constrained with limited storage space and battery power. Due to the limitation of available buffer or energy, some incoming messages are easy to be discarded in recipients. How to ensure newly arrived messages are not dropped by receivers?*

Problem 3. *In MSN, a node, e.g., i, stores a set of messages, e.g., $\mathbb{C}_i$, and gains different forwarding profits by delivering* *diverse messages. Moreover, it cannot send all messages out in an encounter opportunity due to the shortage of encounter duration. How to maximize the sum of forwarding profits in each encounter opportunity?*

$$max \sum_{m \in \mathbb{C}_i} P(m)_{i,j} \times x_m$$

where $P(m)_{i,j}$ is the forwarding profit that i gains by forwarding message m to node j, and x_m represents m is selected to be transmitted (x_m = 1) or not (x_m = 0).

3.2 Network Model

MSN is a class of resource-constrained networks where nodes have limited buffer space and energy. Each node has finite buffer for messages from other nodes, but possess unlimited storage space for messages generated by itself. When a buffer is full, some messages from others are discarded by a buffer management mechanism. In addition, each node is supported by a certain amount of battery power. When a node performs some basic operations such as neighbor discovery and message delivery, it needs to consume a specified energy, while other operations are energy free. After all energy is exhausted, the node will die. In order to simplify the model of energy consumption, the total energy consumption is associated with message size. The larger a message is, the more energy it will consume.

In MSN, nodes always exhibit social properties, affecting their movements. More specifically, nodes having identical social properties (e.g., classmate, workmate, etc.) always do similar things and appear in a particular area for a specific period of time. For example, workers from the same company appear in the same work district and frequently encounter their workmates during work time. Therefore, social ties between nodes denote their future encounter opportunities (or probabilities). In addition, we assume that all nodes are very honest and no node is willing to cheat or attack others, e.g., forgery, data modification and eavesdropping.

3.3 Node Selfishness Model

In MSN, nodes also behave various degrees of selfishness, and are interested in forwarding messages from some nodes, rather than all other nodes. In selfishness model, selfish behavior is considered from two aspects [5]–[7]. One is individual selfishness [5], which means a node exhibits the same forwarding willingness to any other nodes in a network. The other is social selfishness [6], denoting nodes are more interested in transmitting messages for nodes with whom they have social ties, but not others. For example, selfish nodes are more interested in receiving and forwarding messages for nodes in the same community, but not willing to do that for nodes outside their communities.

We model a network as a directed weighted graph $G = (V, E)$ where V is a set of nodes and links between them consist E. The weight of edge $e_{i,j}$ represents node i's selfishness to forward messages generated by node j. The weight of $e_{i,j}$ and $e_{j,i}$ may be different. The values of selfishness range from 0 to 1, where 0 denotes nodes refuse to forward any messages from others and 1 indicates nodes are willing to provide communication services for any other nodes. When a selfish value is within (0, 1), this value means a node forward messages with a certain probability. In addition, all destination nodes are quite willing to accept all messages destined to themselves.

4 Overview of CAMF

Here we provide an overview of CAMF and explain how it works. For the convenience of readers, the major notations used in this paper are listed in Table 1.

TABLE 1
List of Notations

Variable	Description
i, j	nodes i and j
m, m_s, m_d	message m, m's source node, m's destination node
PV_i	i's property vector space
$Sim_{i,j}, Sel_{i,j}$	node similarity between i and j, i's selfishness for j
s_m	m's message size
$F(m)_i$	i's forwarding capability for message m
$B(init)_i, E(real)_i$	the initial buffer size of i, the real-time remaining energy of i
$B(m)_i, E(m)_i$	the reserved buffer size of i for message m, the reserved energy of i for m
$P(m)_{i,j}$	forwarding profit that i gains by transmitting m to j
σ	energy consumption used to forward a data with an unit size
$\mathbb{C}_i$	a set of messages buffered by node i
$\mathbb{F}_i$	i's forwarding set

4.1 CAMF Structure

4.1.1 *Receiving Capability Measurement*

Receiving capability measurement is to quantify how many available resources a node can supply for newly arrived messages. In a comprehensive literature review, we notice that packet loss is often incurred by several factors, including no available storage space, no enough battery power, or the instability of wireless channels. Since we ignore the underlying wireless technology in this paper, buffer and energy are only considered to evaluate receiving capabilities of nodes. Here we define two new concepts, called *reserved buffer size* and *reserved energy*, to denote how much available buffer size and energy a receiver can afford to incoming messages, respectively.

In MSN, node behaviors are always affected by their selfishness. For example, selfish nodes are unwilling to afford any resource to those messages created by all or some nodes in a network. Inspired by this, we utilize node selfishness to rank messages and assign priorities to them. The larger the current carrier's selfishness for a message is, the higher the message's priority is. In addition, these messages generated by the current custody node have the largest priority. We then allocate available resources to messages according to their priorities. The higher the priority is, the more resources a message can gain. Detailed information about *reserved buffer size* and *reserved energy* is given in Section 5.1. After two nodes have exchanged their state information, receiving capability measurement is triggered to measure how many available resources a candidate receiver can provide for possible incoming messages buffered in the peer.

4.1.2 Forwarding Capability Measurement

As the name implies, forwarding capability measurement is to quantify the probability that a node delivers a message to its destination. In our forwarding capability measurement, we put forward a stateless approach, which does not need to collect and store any state information, to measure social ties between nodes and further evaluate their future encounter opportunities. Be specific, we adopt *node property profile*, a multi-dimension vector, to describe nodes, and then measure node similarity between them. Since node similarity implies social ties and reflects connection probability, we employ it to quantify encounter opportunities and further evaluate forwarding capabilities of nodes.

Meanwhile, since node selfishness significantly affects the efficiency of data transmission in MSN, we also take it into account to measure forwarding capabilities of nodes. Here two kinds of selfishness are considered, e.g., individual selfishness and social one. Detailed information on how to evaluate the forwarding capability of nodes is introduced in Section 5.2. After a forwarding set is initialized, forwarding capability measurement is triggered to measure forwarding capabilities of nodes.

4.1.3 Forwarding Set Mechanism

Be brief, forwarding set mechanism is employed to determine whether and when a message could be forwarded when a contact opportunity arises. Since a node has different forwarding capabilities for various destinations, a routing protocol that adopts a random or simplified forwarding set strategy is easy to cause some messages to be forwarded to nodes with a relatively lower forwarding capability. Moreover, due to the uncertainty of a contact duration, not all the messages will be sent out in an encounter. It cannot guarantee that all messages that could gain more forwarding opportunities will be forwarded in a connection. In other words, some messages that may acquire more forwarding opportunities to be delivered to destination nodes are retained in the current custody node, while others that may get relatively fewer forwarding chances are transmitted.

To avoid the above-mentioned situations, we first introduce a new metric, called *forwarding profit*, to denote the relative difference of forwarding capabilities that two encountered nodes can provide. Then, we model the forwarding set determination problem as a $0/1$ multiple knapsack problem to maximize the sum of forwarding profits in each encounter opportunity. Meanwhile, we also consider receiving capabilities of nodes to avoid messages being forwarded to those nodes where no buffer or energy is available. Once obtaining forwarding profits of each message in a forwarding set, forwarding set mechanism is triggered to determine the final forwarding set.

4.1.4 Buffer Management

Buffer management is to decide which messages should be dropped when a buffer overflows. It is important for MSN where nodes are constrained with storage space. In our buffer management, each message is associated with a priority, introduced in Section 4.1.1. This priority denotes the forwarding probability that the custody node can supply. When a buffer is full or insufficient for storing any incoming data, it may discard messages with the lowest priority firstly.

4.2 CAMF Algorithm

Here we present a context-aware message forwarding algorithm (CAMF), which exploits context knowledge to quantify forwarding and receiving capabilities of nodes and utilizes them to determine the final forwarding set. When two nodes meet, they first transmit the messages destined to the peer node. Then, they exchange their forwarding and receiving capabilities for messages stored in the peer's buffer. Based on such information, a forwarding set mechanism is employed to choose suitable messages to be included in the forwarding set to maximize the forwarding profit.

In order to better illustrate how CAMF works, we take the message transmission process between nodes i and j as an example.

1). When nodes i and j are in contact, they first deliver messages destined to the peer, and then exchange their state information, including selfishness and forwarding capability and messages buffered.

2). Node j returns its receiving capability for messages in i's buffer.

3). According to the peer's receiving capability, i initializes its forwarding set $\mathbb{F}_i$ by including those messages satisfying the following conditions. 1) It belongs to $\mathbb{C}_i$; 2) the *reserved buffer size* that it obtains from j exceeds itself, e.g., $B(m)_j > s_m$; 3) the *reserved energy* that it gets from j is sufficient to send itself out, e.g., $E(m)_j > s_m \times \sigma$.

4). Based on forwarding capabilities of i and j, i calculates the forwarding profit of each message in $\mathbb{F}_i$, and then removes these messages whose forwarding profit is not greater than zero.

5). i decides the final forwarding set by solving a $0/1$ multiple knapsack problem and determines its forwarding order.

6). i transmits messages in $\mathbb{F}_i$ one by one until no message left in $\mathbb{F}_i$ or the connection is disconnected.

For convenience of explanation, we only introduce how node i determines its forwarding set in the last four steps. In addition, node j does so in a similar way.

5 Detailed Design of CAMF

This section mainly introduces the detailed design of CAMF, including forwarding capability measurement, receiving capability measurement and forwarding set mechanism.

5.1 Receiving Capability Measurement

Definition 1. ***Receiving capability*** *denotes how much storage space or battery power a receiver can offer to messages.*

In a comprehensive literature review, we find that packet loss is often incurred by several factors, including no available storage space in receivers, no adequate battery power in recipients, or instability of wireless channels. Since we ignore the underlying wireless technology, buffer size and energy are only considered to evaluate receiving capabilities of nodes. In this paper, resource assignment is based on the message priority, introduced in Section 4.1.1. The larger the selfishness value is, the higher the priority is and the more resources (e.g., storage space and energy) a message can gain.

In order to better quantify receiving capabilities of nodes, we first gather a number of messages, described as Eq. (1).

$$\mathbb{C}(m)_i = \{n | n \in \mathbb{C}_i,\ Sel(i, n\_s) \geq Sel(i, m\_s)\} \tag{1}$$

Since the priority of messages in $\mathbb{C}_i$ is larger than that of message m, they can acquire more resources than m. In other words, compared with m, they have a high priority to obtain more storage space and battery power in the process of resource allocation.

Definition 2. ***Reserved buffer size*** *denotes the maximum storage space that a recipient can reserve to possible arrived messages, and its definition is expressed below.*

$$B(m)_i = B(init)_i - \sum_{n \in \mathbb{C}(m)_i} s_n$$

where $B(m)_i$ denotes the largest possible buffer capacity that node i can supply to message m. An incoming message may be stored by the possible accepter only when its acquired *reserved buffer size* exceeds its size, described as $B(m)_i > s_m$.

Definition 3. ***Reserved energy*** *is the maximum power resource that a receiver can reserve to possible arrived messages. Reserved energy that node i provides for message m is presented as follows.*

$$E(m)_i = E(real)_i - \sum_{n \in \mathbb{C}(m)_i} s_n \times \delta$$

Reserved energy denotes whether a receiver has ample energy to send a stored message out or not. Node i may accept and store message m only when the following equation is met.

$$E(m)_i > s_m \times \sigma$$

5.2 Forwarding Capability Measurement

Definition 4. ***Forwarding capability*** *is the probability that a node directly delivers a message to its destination.*

When quantifying forwarding capabilities of nodes, we not only consider pure encounter opportunities between nodes, but also take their selfishness into account.

5.2.1 Encounter Opportunity Measurement

Mei et al. [16] notice that people's movement is always affected by their interest, and thus propose a stateless approach to utilize an interest profile to evaluate encounter opportunities between nodes. This approach does not take any state information collected from the whole network. Nevertheless, individual movement is not only guided by their interest, but also affected by other factors, e.g., their affiliation. Thus, in order to better forecast future encounter opportunities, more node properties and characteristics, not limited to interest, should be considered to figure nodes.

Definition 5. ***Node property profile*** *is a set of node properties that can be used to describe nodes. It not only contains all kinds of interest, but also includes other possible meaningful properties describing the main characteristics of individuals, including affiliation, home address, city, colleagues, the prefix of ID cards, nationality, spoken language, and so on.*

We employ node property profile to describe nodes and model each node as a multi-dimension property vector space, described as:

$$PV(pv_1, pv_2, ..., pv_n)$$

where each entry, e.g., $pv_i, i \in \{1, 2, ..., n\}$, denotes a property, and n is the total number of node properties that we consider in this paper. The types of property values vary from a property to a property. Some are a real number, such as the prefix of ID cards. Some others may be a semantic value, such as interest.

Definition 6. ***Node similarity*** *denotes the similar degree between node property profiles.*

In order to quantify node similarity, we utilize the well-known Dice' coefficient [18], which can effectively measure the similarity of vectors. The similar degree between nodes i and j, denoted $Sim_{i,j}$, is expressed below.

$$Sim_{i,j} = \frac{2 \times PV_i \times PV_j}{||PV_i|| + ||PV_j||}$$

where PV_i represents the n-dimension property vector of node i. *Node similarity* denotes social ties between nodes and further implies their connection probabilities. Therefore, we directly utilize node similarity to represent future encounter opportunities.

5.2.2 Forwarding Capability

We consider two factors to assess forwarding capabilities of nodes. One is the similar degree of node property profiles; the other is node selfishness. Forwarding capability that node i provides for message m is defined as follows.

$$F(m)_i = Sel_{i,m\_s} \times Sim_{i,m\_d}$$

where Sel_{i,m_s} represents node i's selfishness for message m. Since node similarity (e.g., Sim_{i,m_d}) and selfishness (e.g., Sel_{i,m_s}) both range from 0 to 1, forwarding capability also belongs to such range. In order to better capture node selfish behavior, we consider two classes of selfishness: individual and social selfishness. Detailed information on how to acquire these selfishness is introduced in Section 6.1.2.

5.3 Forwarding Set Mechanism

When a contact opportunity occurs, it is necessary to make a decision on which candidate messages should be forwarded and the forwarding order. Most existing routing protocols neglect the importance of forwarding set, and adopt a random or simplified strategy to decide the final forwarding set. This cannot guarantee that all messages are delivered to nodes with an incremental forwarding capability.

In order to cope with such a challenge, we introduce a new notion, called *forwarding profit*, to denote the change of forwarding probabilities gained by a message in a forwarding process. In the following, we take an example that node i forwards a number of messages to j to illustrate our forwarding set mechanism.

Definition 7. ***Forwarding profit*** *is the difference of forwarding capabilities of two nodes. More specifically, when node i forwards a message m to node j, the forwarding profit gained by i is described as follows.*

$$P(m)_{i,j} = F(m)_j - F(m)_i$$

The value of $P(m)_{i,j}$ is positive or negative. Node i can acquire its forwarding profit only when a message m is successfully accepted by the peer j; otherwise, $P(m)_{i,j}$ is zero.

We model the forwarding set problem as a $0/1$ multiple knapsack problem to maximize the sum of forwarding profits in each encounter opportunity. To avoid messages being delivered to the node with insufficient storage space or battery power, we initialize the forwarding set by including those messages who satisfy these conditions, described as the following equation.

$$\mathbb{F}_i = \{m | m \in \mathbb{C}_i \ and \ B(m)_j > s_m \ and \ E(m)_j > s_m \times \sigma\}$$

Two encountered nodes may have different forwarding capabilities to the destination of a message. Thus, forwarding profit may be positive or negative. Here we remove those messages in $\mathbb{F}_i$ whose forwarding profits are non-positive, and then sort them in an increasing order of forwarding capabilities provided by node i. We simply use m to denote the m-th message in $\mathbb{C}_i$. x_m represents message m is selected to be transmitted ($x_m = 1$) or not ($x_m = 0$). By incorporating receiving capability of the receiver, the forwarding set problem is formulated as:

$$\begin{aligned} \max \sum_{m \in \mathbb{F}_i} & P(m)_{i,j} \times x_m \\ s.\ t. \quad \forall m, & \sum_{l \le m} x_l \times s_l \le B(m)_j \\ and & \sum_{l \le m} x_l \times s_l \times \sigma \le E(m)_j \end{aligned} \tag{2}$$

We convert this maximum forwarding profit problem into a $0/1$ multiple knapsack problem [17]. Firstly, buffer space and remaining energy are divided into $|\mathbb{F}_i|$+1 knapsacks, and they are assigned as follows.

For the first knapsack, its buffer size and energy are

$$\Delta B(1)_j = B(1)_j$$
$$\Delta E(1)_j = E(1)_j$$

For the nth ($n \in \{2, ..., |\mathbb{F}_i|\}$) knapsack, its buffer size and energy are

$$\Delta B(n)_j = B(n)_j - B(n-1)_j$$
$$\Delta E(n)_j = E(n)_j - E(n-1)_j$$

Then, message m can only put to the knapsacks (e.g., n) whose index is smaller than or equal to m. Let $x_{m,n}$ represents that message m is put to knapsack n ($x_{m,n} = 1$) or not ($x_{m,n} = 0$). If $m < n$, $x_{m,n} = 0$. Eq.(2) can be rewritten as follows.

$$\begin{aligned} \max \sum_{m=1}^{|\mathbb{F}_i|} \sum_{n=1}^{|\mathbb{F}_i|} & P(m)_{i,j} \times x_{m,n} \\ s.\ t. \quad \forall n, & \sum_{m} x_{m,n} \times s_m \le \Delta B(n)_j, \\ & \sum_{m} x_{m,n} \times s_m \times \sigma \le \Delta E(n)_j \\ and \quad \forall m, & \sum_{n} x_{m,n} \le 1 \end{aligned} \tag{3}$$

It is well known that the $0/1$ multiple knapsack problem is NP-hard [17]. In order to maximize forwarding profits of each connection, we design an approximate algorithm, described as follows.

1) we first initialize the forwarding set $\mathbb{F}_i$ by including messages from $\mathbb{C}_i$ who meet the following conditions. (1)The forwarding profit that it acquires exceeds zero; (2) the reserved buffer size that it gets

from the peer node is greater than itself; (3) the reserved energy that it obtains from the peer node is sufficient to send itself out in the next hop.

2) We sort the forwarding set $\mathbb{F}_i$ in a decreasing order of forwarding profit.

The details are shown in Algorithm 1. The time complexity of Alg. 1 is $\Theta(n \log n)$ where n is the number of messages buffered in the local cache of a node.

Algorithm 1 An approximate algorithm for forwarding profit maximization, pseudo-code of node i

Input: $\mathbb{C}_i$ and $\forall m \in \mathbb{C}_i, F(m)_i, F(m)_j, B(m)_j, E(m)_j$
Output: $\mathbb{F}_i$

if node i meets others (e.g., j) and $\mathbb{C}_i \neq \emptyset$ **then**
  initiaze: $\mathbb{F}_i = \emptyset$
  for all $m \in \mathbb{C}_i$ **do**
    if $F(m)_j > F(m)_i$ and $B(m)_j > s_m$ and $E(m)_j > s_m \times \sigma$ **then**
      $\mathbb{F}_i \leftarrow \mathbb{F}_i \bigcup \{m\}$
    end if
  end for
  Sort $\mathbb{F}_i$ in decreasing order of forwarding profit
  return $\mathbb{F}_i$
end if

6 PERFORMANCE EVALUATION

We conduct simulations on the widely-used simulator ONE [19] to evaluate the effectiveness of CAMF.

6.1 Experiment Setup

6.1.1 Mobility Datasets

Since datasets Infocom06 [20] includes node property information, we employ it in our simulations. To better describe nodes, we extract some meaningful property labels from the original data, e.g., nationality, language, affiliation, position, city, country and topics, etc.. However, not all nodes have such property labels in Infocom06. We remove these nodes lacking these property labels.

However, the node scale of Infocom06 is not large enough. In order to draw a more general conclusion, we need a mobile model to create a larger mobility dataset. Since SWIM [21] shows its superiority on generating a synthetic mobility datasets with social characteristics, we adopt it to generate a synthetic mobility datasets with 300 nodes in our simulations.

Since synthetic datasets lack node property labels, we utilize two ways to generate node property profiles for better studying the performance of CAMF.

I: One way is to generate a multi-dimensional node property profile vector for each node, in which entries are chosen independently and uniformly at random in [0,1]. The profile vector is then normalized to 1 for the convenience of calculation.

II: The other is to create a n-dimensional node property profile vector for each node, where n is the total number of nodes in a network and an entry denotes the encounter probability between nodes. Taking the property profile vector of node i as an example, the jth entry is the proportion of the contract frequency between nodes i and j to that between node i and all other nodes in a network. If node i never meet node j, the jth entry is zero. Meanwhile, the ith entry is also zero in the property profile of node i.

CAMF that utilizes node property profiles generated by way I and II are referred as **CAMF-I** and **CAMF-II**, respectively.

6.1.2 Node Selfishness Setting

Individual selfishness has been well studied, and its distribution can be described by a normal distribution model properly [22]. Hence we employ a normal distribution model to figure node's individual selfishness values. These values follow a normal distribution with values normalized between 0 and 1. Since the range of a normal distribution is from negative infinity to positive infinity, we adopt 10% and 90% of its cumulative distribution function (CDF) value to normalize them. For example, if the value of the cutoffs at the 10% point is -5 and at the 90% point is 5, all results will be increased by 5 and then subtracted by -5, described as 5-(-5) =10.

Social selfishness always depends on social ties between nodes. Since mobility datasets do not have explicit social relations between nodes, we construct a weighted social graph and further obtain the distribution of social selfishness values. More specifically, we firstly create power-law distributed node degrees, and assign them to nodes [23]. The largest degree is assigned to the node having the largest contact frequency with others, and repeats it until the remaining degrees are assigned to all nodes. For each node in a network, we connect its ties to other nodes, and generate social selfishness values for edges. These values are uniformly distributed between 0 and 1, consisting with the previous study [24]. Taking node i as an example, we assume its degree is x. We first calculate the contact frequency between i and all other nodes, and connect i with x nodes, having the largest contact frequency with i. Then, we assign selfish values to edges between i and the x nodes. The largest selfishness value is assigned to the edge with the largest contact frequency, and then we repeat it until the remaining selfishness values are assigned to edges. Finally, for those node pairs, e.g., nodes j and k, that have not been connected, their selfishness, e.g., $e_{j,k}$ and $e_{k,j}$, are zero.

6.1.3 Others

In our simulations, each generated message has a certain lifetime, denoted as TTL, and its source and destination are randomly chosen from all nodes in a network. At each round, default energy consumption parameter σ is 4. Each simulation is repeated 20 times with different random seeds for statistical confidence.

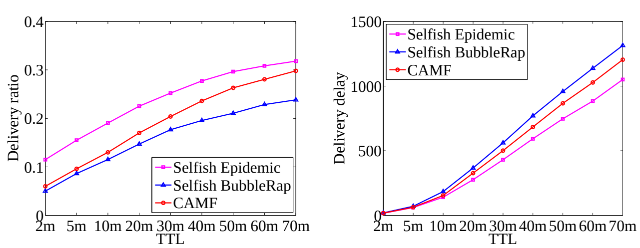
Since most existing routing protocols do not take node selfishness into account, we modify them into selfishness-aware version for a fair comparison. We compare CAMF against the following benchmark algorithms:

- **Selfish Epidemic**: a selfishness-aware version of Epidemic [25].
- **Selfish BubbleRap**: a variant of BubbleRap [14] which considers node selfishness to make forwarding decisions.

We utilize the following criterias to evaluate routing performance.

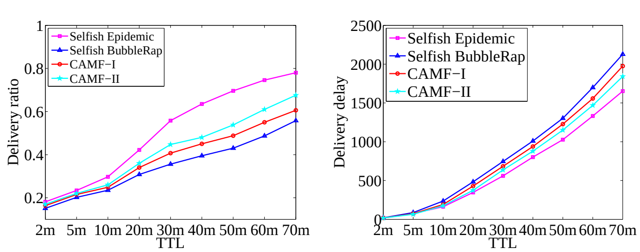
- **Delivery ratio**: it is the proportion of messages that are delivered to destinations out of the total messages generated in the source within a given period (e.g., TTL).
- **Delivery overhead**: it is a measure of the average number of forwarding a network spends to deliver a message to its destination.
- **Delivery delay**: it is defined as the average time that a network spends to deliver a message to its destination.
- **Average energy consumption**: it is the proportion of the consumed energy out of the initial energy. It denotes the average energy consumption level of nodes in the entire simulation.

6.2 Simulation Results on Social Selfishness



(a) Delivery ratio (b) Delivery delay

Fig. 1. Delivery ratio and delivery delay results of all algorithms with social selfishness on datasets Infocom06.



(a) Delivery ratio (b) Delivery delay

Fig. 2. Delivery ratio and delivery delay results of all algorithms with social selfishness on datasets SWIM.

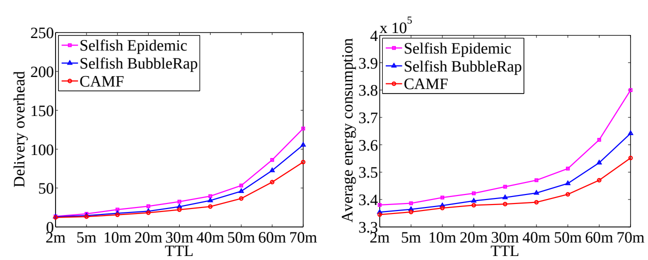
6.2.1 *Delivery Ratio and Delivery Delay Evaluation*

As shown in Fig.1-2, we see that CAMF performs better than Selfish BubbleRap in terms of delivery ratio and delivery delay. Taking datasets SWIM as an example, we observe that CAMF-I's delivery ratio and its delivery delay are 20% greater and 13% lower than Selfish BubbleRap when TTL is 60m, respectively. We argue that two reasons can explain it.

1) CAMF measures receiver's receiving capability before transmitting messages, and delivers messages to nodes having enough receiving capability to serve incoming data. This could avoid the phenomenon that newly arrived messages are dropped in receivers because of insufficient buffer space or battery power. Nevertheless, Selfish BubbleRap ignores receiving capabilities of nodes, and thus cannot avoid such a situation.
2) CAMF employs a forwarding set mechanism to determine the forwarding set to maximize the forwarding profits of each connection. However, Selfish BubbleRap overlooks the importance of such a mechanism. After choosing an appropriate node as a relay node, it randomly selects messages from its buffer and greedily forward them to the relay. This cannot all messages transmitted to the peer node gain an increased delivery capability.

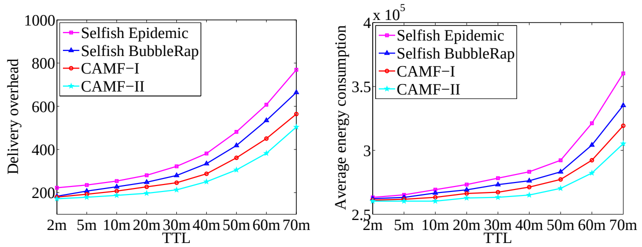
From Fig.2, we can also draw that CAMF-II achieves a higher delivery ratio and a shorter delivery delay than CAMF-I. This is due to the difference of approaches that they utilized to create node property profiles. In CAMF-I, the property profile of each node is created by a random strategy, which is unable to avoid the situation that the similar degrees of some nodes are high, while their contact frequencies are low. This will lead to some messages are delivered to those nodes who may be far from their destination nodes in CAMF-I. However, in CAMF-II, node property profiles are constructed by the contact frequency between nodes. If two nodes always encounter, they shall have a high similar degree of node property; otherwise, they have a low one. Therefore, CAMF-II can avoid the phenomenon emerged in CAMF-I, and outperforms CAMF-I in terms of delivery ratio and delivery delay.

6.2.2 *Delivery Overhead and Average Energy Consumption Evaluation*



(a) Delivery overhead (b) Average energy consumption

Fig. 3. Delivery overhead and average energy consumption results of all algorithms with social selfishness on datasets Infocom06.



(a) Delivery overhead (b) Average energy consumption

Fig. 4. Delivery overhead and average energy consumption results of all algorithms with social selfishness on datasets SWIM.

Fig.3-4 display that CAMF has the least delivery overhead and the minimum average energy consumption, compared with other algorithms. Taking datasets SWIM as an example, delivery overhead and average energy consumption of CAMF-I are much lower than that of Selfish BubbleRap when TTL is 60m, e.g., 73% and 86% of Selfish BubbleRap in terms of delivery overhead and energy consumption, respectively. This can be illustrated by three factors.

1) CAMF measures forwarding capability of a node for a specified message before data transmission, and then delivers this message to a node who has an incremental forwarding capability to deliver it to its destination node. This could shorten the forwarding number (or hop) of messages. However, Selfish BubbleRap adopts node betweenness to make forwarding decisions. A node with a high betweenness means it may frequently encounter some other nodes, rather than all nodes in a network. Thus, it cannot guarantee messages always gain an improved delivery probability during the process of message transmission.
2) CAMF measures receiving capabilities of nodes, and forward messages to those nodes who can offer abundant resources to incoming messages. This could prevent messages to be delivered to nodes where there is no enough buffer or energy. However, Selfish BubbleRap ignores receiving capabilities of receivers and greedily transmit messages to them. Therefore, some newly arrived messages are easy to be dropped due to the limitation of storage space or energy in the recipient.
3) CAMF adopts a stateless approach to measure forwarding capabilities of nodes, without collecting any topology information. However, to calculate betweenness centrality of each node, Selfish BubbleRap needs to collect the global network topology knowledge, which should take limited contact opportunities and energy to transfer such information.

In addition, compared with Selfish Epidemic, we can notice from Fig. 3-4 that CAMF achieves relatively lower delivery overhead and energy consumption. This is because CAMF takes multiple strict conditions to choose suitable relay nodes, and depends on a few nodes to forward messages in the process of message transaction. Nevertheless, Selfish Epidemic only takes node selfishness into account to make forwarding decisions, causing messages are easy to be flooded to every possible part of an entire network. This means the potential cost of finding destinations is dramatically increased. Therefore, delivery overhead and average energy consumption are both improved greatly.

6.2.3 Performance Comparison Between Infocom06 and SWIM

Fig.1-4 reveal that all algorithms have relatively higher delivery overhead and delivery delay on SWIM, compared with Infocom06. In our simulations, the source and destination of each message are chosen randomly among all nodes in a network, and thus the average distance between source and destination changes with the number of nodes in datasets. A dataset including more nodes means it has a long average transmission distance, and thus it should spend more time on transmitting messages from source nodes to destinations. The node scale of Infocom06 is smaller than that of SWIM. Therefore, all algorithms achieve a relatively high delivery overhead and delay on SWIM, compared with Infocom06.

We also find that all algorithms have a higher delivery ratio on SWIM than that on Infocom06. This is because SWIM has a higher average contact frequency of each node pairs than Infocom06. Since all algorithms exploit contact opportunities to forward messages, the high contact frequency denotes that there are more opportunities to transmit messages. Hence all algorithms achieve relatively higher delivery ratio on SWIM.

6.2.4 Others

We can notice clearly from Fig. 1-4 that all results under different performance metrics are growing with the incremental of TTL on both Infocom06 and SWIM. Since TTL denotes the lifetime of messages, a larger TTL indicates messages stay in a network longer. When TTL is increased, messages have more opportunities to encounter others and thus the probability of being delivered to destinations is also improved. Meanwhile, a larger TTL also means more energy will be consumed in the message transmission process.

6.3 Simulation Results on Individual Selfishness

We also evaluate the effects of individual selfishness for these forwarding algorithms. Through compared the simulation results obtained from both individual and social selfishness, we find that the results acquired from individual selfishness are quite similar with that from social selfishness. Therefore, we do not display the results from individual selfishness and analyze and discuss them any more for saving space.

Still we want to stress that the advantage of CAMF over other selfishness-aware routing algorithms becomes even more evident on both Infocom06 and SWIM. This is because it considers receiving and forwarding capability of receivers and further utilizes a forwarding set mechanism to maximize the sum of forwarding profits in each connection opportunity. This is in line with our design goals: improving routing efficiency when nodes are allowed to behave selfish behavior.

7 Conclusion

In this paper, we regard selfishness as a native attribute of a system and seek to design an efficient message forwarding algorithm in which nodes are allowed to exhibit selfishness. We first put forward a stateless way to measure encounter opportunities between nodes, and combine them with node selfishness to evaluate forwarding capabilities of nodes. We then quantify receiving capabilities of nodes according to the maximum available buffer space and energy. Based on these, we formulate the forwarding set problem as a multiple knapsack problem to choose forwarding candidate messages so as to maximize the forwarding profit. To fully explore the benefits of our fundamental studies on forwarding and receiving capabilities measurement and forwarding set mechanism, we propose a context-aware message forwarding algorithm, in which buffer management is particularly considered. Through extensive real-trace driven experiments, the superior efficiency of CAMF is validated as it significantly outperforms other typical selfishness-based routing protocols in terms of delivery ratio, delivery overhead, delivery delay and average energy consumption.

Acknowledgment

We thank the anonymous reviewers for their patient review and valuable comments which significantly improve the quality of this paper. This research was supported by National 863 Program of China (Grant No. 2012AA011005), the fund of the State Key Lab of Software Development Environment (Grant No. SKLSDE-2013ZX-06), NSFC Grant No. 61450110085, the fund of the State Key Laboratory of Industrial Control Technology (Grant No. ICT1407), JSPS KAKENHI (Grant No. 25880002, 26730056) and JSPS A3 Foresight Program.

References

[1] H. Zhou, J. Wu, H. Zhao, S. Tang, C. Chen, and J. Chen, "Incentive-Driven and Freshness-Aware Content Dissemination in Selfish Opportunistic Mobile Networks," in *Proceedings of the 10th International Conference on Mobile Ad-Hoc and Sensor Systems*, 2013, pp. 333–341.

[2] X. Liang, X. Li, T. H. Luan, R. Lu, X. Lin, and X. Shen, "Morality-driven data forwarding with privacy preservation in mobile social networks," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 3209–3222, 2012.

[3] K. Wei, R. Duan, G. Shi, and K. Xu, "Distribution of inter-contact time: An analysis-based on social relationships," *Journal of Communications and Networks*, vol. 15, no. 5, pp. 504–513, 2013.

[4] J. Fan, J. Chen, Y. Du, W. Gao, J. Wu, and Y. Sun, "Geocommunity-Based Broadcasting for Data Dissemination in Mobile Social Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 4, pp. 734–743, 2013.

[5] H. Zhu, X. Lin, R. Lu, S. Member, Y. Fan, and X. (sherman) Shen, "SMART: A Secure Multilayer Credit-Based Incentive Scheme for Delay-Tolerant Networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 4628–4639, 2009.

[6] Y. Li, P. Hui, D. Jin, L. Su, and L. Zeng, "Evaluating the impact of social selfishness on the epidemic routing in delay tolerant networks," *IEEE Communications Letters*, vol. 14, no. 11, pp. 1026–1028, 2010.

[7] Q. Li, W. Gao, S. Zhu, and G. Cao, "A routing protocol for socially selfish delay tolerant networks," *Ad Hoc Networks*, vol. 10, no. 8, pp. 1619–1632, 2012.

[8] L. Wei, Z. Cao, and H. Zhu, "Mobigame: A user-centric reputation based incentive protocol for delay/disruption tolerant networks," in *Proceedings of IEEE International Conference on Global Communications Conference*, 2011, pp. 1–5.

[9] Z. Li and H. Shen, "Game-theoretic analysis of cooperation incentive strategies in mobile ad hoc networks," *IEEE Transaction on Mobile Computing*, vol. 11, no. 8, pp. 1287–1303, 2012.

[10] M. Dong, K. Ota, M. Lin, Z. Tang, S. Du, and H. Zhu, "Uav-assisted data gathering in wireless sensor networks," *The Journal of Supercomputing*, pp. 1–14, 2014. [Online]. Available: http://dx.doi.org/10.1007/s11227-014-1161-6

[11] B. B. Chen and M. C. Chan, "MobiCent: a Credit-Based Incentive System for Disruption Tolerant Network," in *Proceedings of the 29th Conference on Information Communications*, 2010, pp. 875–883.

[12] J. Zhou and Z. Cao, "Tis: A threshold incentive scheme for secure and reliable data forwarding in vehicular delay tolerant networks," in *Proceedings of IEEE International Conference on Global Communications Conference*, Dec 2012, pp. 985–990.

[13] K. Wei, X. Liang, and K. Xu, "A survey of social-aware routing protocols in delay tolerant networks: Applications, taxonomy and design-related issues," *IEEE Communications Surveys Tutorials*, vol. 16, no. 1, pp. 556–578, 2014.

[14] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: social-based forwarding in delay tolerant networks," in *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, 2008, pp. 241–250.

[15] Y. Cao, Z. Sun, N. Wang, F. Yao, and H. S. Cruickshank, "Converge-and-diverge: A geographic routing for delay/disruption-tolerant networks using a delegation replication approach," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 2339–2343, 2013.

[16] A. Mei, G. Morabito, P. Santi, and J. Stefa, "Social-aware stateless forwarding in pocket switched networks," in *Proceedings of the 30th IEEE international conference on Computer Communications*, 2011, pp. 251–255.

[17] L. Zhang and S. Geng, "The Complexity of 0/1 Multi-knapsack Problem," *Journal of Computer Science and Technology*, vol. 1, no. 5, pp. 46–50, 1986.

[18] W. B. Frakes and R. A. Baeza-Yates, Eds., *Information Retrieval: Data Structures Algorithms*. Prentice-Hall, 1992.

[19] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation," in *proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques*, 2009.

[20] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "CRAWDAD cambridge/haggle/imote/infocom2006," 2009.

[21] A. Mei and J. Stefa, "SWIM: A Simple Model to Generate Small Mobile Worlds," in *Proceedings of the 28th IEEE international conference on Computer Communications*, 2009, pp. 2106–2113.

[22] P. Hui, K. Xu, V. Li, J. Crowcroft, V. Latora, and P. Lio', "Selfishness, altruism and message spreading in mobile social networks," in *28th IEEE International conference on Computer Communications*, April 2009, pp. 1–6.

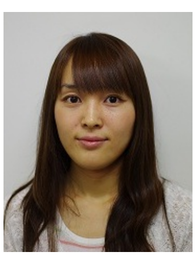
[23] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and Analysis of Online Social Networks," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, 2007, pp. 29–42.

[24] E. Gilbert and K. Karahalios, "Predicting tie strength with social media," in *Proceedings of the SIGCHI Conference on Human Factors in Computing System*, 2009, pp. 211–220.

[25] A. Vahdat and D. Becker, "Epidemic Routing for Partially-connected Ad Hoc Networks," Duke University, Tech. Rep., 2002.

Kaimin Wei joined the State Key Lab of Software Development Environment (SKLSDE) in 2010, and is currently a Ph.D. Student at the Beihang University in Beijing, majoring in computer science. He is a student member of IEEE, and his current research interest is in the areas of wireless mobile networks, with a special emphasis on protocol design and analysis, as well as security and privacy protection.

Mianxiong Dong (M13) received B.S., M.S. and Ph.D. in Computer Science and Engineering from The University of Aizu, Japan. He is currently a Researcher with National Institute of Information and Communications Technology (NICT), Japan. He was a JSPS Research Fellow with School of Computer Science and Engineering, The University of Aizu, Japan and was a visiting scholar with BBCR group at University of Waterloo, Canada supported by JSPS Excellent Young Researcher Overseas Visit Program from April 2010 to August 2011. Dr. Dong was selected as Foreigner Research Fellow (A total of 3 recipients all over Japan) by NEC C&C Foundation in 2011. He is the Best Paper Award winner of IEEE HPCC 2008 and IEEE ICESS 2008. Dr. Dong is currently a research scientist with A3 Foresight Program (2011-2016) funded by Japan Society for the Promotion of Sciences (JSPS), NSFC of China, and NRF of Korea. His research interest include wireless sensor networks, vehicular ad-hoc networks and wireless security.



Kaoru Ota (M12) received M.S. degree in Computer Science from Oklahoma State University, USA in 2008 and Ph.D. degree in Computer Science and Engineering from The University of Aizu, Japan in 2012. She is currently an Assistant Professor with Department of Information and Electronic Engineering, Muroran Institute of Technology, Japan. From March 2010 to March 2011, she was a visiting scholar with BBCR group at University of Waterloo, Canada. Also she was a prestigious Japan Society of the Promotion of Science (JSPS) research fellow with Graduate School of Information Sciences at Tohoku University, Japan from April 2012 to April 2013. Dr. Ota is a recipient of JSPS Grant-in-Aid for Research Activity Start-up in 2013. She has joined JSPS A3 foresight program as one of primary researchers since 2011 which is supported by Japanese, Chinese and Korean government. She serves as a Guest Editor of IEICE Transactions on Information and Systems, Special Section on Frontiers of Internet of Things 2014, Editor of Peer-to-Peer Networking and Applications (Springer), Journal of Cyber-Physical Systems, and International Journal of Embedded Systems. Dr. Ota has actively involved in international conferences in present and past. Currently she is the Publicity Co-chair of the 2014 ICC workshop on Secure Networking and Forensic Computing and 2014 IEEE ICCC Workshop on Internet of Things. Her research interest include wireless sensor networks, vehicular networks, and ubiquitous computing.

Ke Xu is a professor at Beihang University, China. He received his B.E., M.E., and Ph.D. degrees from Beihang University in 1993, 1996, and 2000, respectively. His research interest include algorithm and complexity, data mining, and network.