Incentive-Driven and Freshness-Aware Content Dissemination in Selfish Opportunistic Mobile Networks

Huan Zhou, Jie Wu, *Fellow, IEEE*, Hongyang Zhao, Shaojie Tang, *Member, IEEE*, Canfeng Chen, *Member, IEEE*, and Jiming Chen, *Senior Member, IEEE*

Abstract—Recently, the content-based publish/subscribe (pub/sub) paradigm has been gaining popularity in opportunistic mobile networks (OppNets) for its flexibility and adaptability. Since nodes in OppNets are controlled by humans, they often behave selfishly. Therefore, stimulating nodes in selfish OppNets to collect, store, and share contents efficiently is one of the key challenges. Meanwhile, guaranteeing the freshness of contents is also a big problem for content dissemination in OppNets. In this paper, in order to solve these problems, we propose an incentive-driven and freshness-aware pub/sub Content Dissemination scheme, called *ConDis*, for selfish OppNets. In *ConDis*, the Tit-For-Tat (TFT) scheme is employed to deal with selfish behaviors of nodes in OppNets. Moreover, a novel content exchange protocol is proposed when nodes are in contact. Specifically, during each contact, the exchange order is determined by the content utility, which represents the usefulness of a content for a certain node, and the objective of nodes is to maximize the utility of the content inventory stored in their buffer. Extensive realistic trace-driven simulation results show that *ConDis* is superior to other existing schemes in terms of total freshness value, total delivered contents, and total transmission cost.

Index Terms—Publish/subscribe, content dissemination, opportunistic mobile networks, selfish behavior, freshness

1 Introduction

The content-based publish/subscribe (pub/sub) paradigm is a promising technology offering content-based services for nodes in opportunistic mobile networks (OppNets), which consist of a diversity array of portable devices capable of ad hoc wireless communication, e.g., smart phones, PDAs, and laptops [1], [2], [3], [4], [5]. Specifically, end-to-end communication paths are hard to guarantee in OppNets, due to the time-varying network topology in OppNets. Thanks to the decoupling of the binding relationship between source and destination pairs, the pub/sub scheme has high flexibility and adaptability when dealing with highly dynamic network topologies, which bring a tremendous advantage for content dissemination in OppNets. Therefore, there is a great demand for studying such a scheme in OppNets.

Contents in the pub/sub scheme are organized into a set of channels, while a channel is a set of labels describing a type of content, and the goal is to deliver contents from publishers to subscribers [6], [7], [8], [9]. The subscribers, namely the content consumers, express their interest without knowledge of the content generators' specific ID(s), and their interest is always stable over a long period of time; the publishers generate contents to the network without specifying the destination ID(s). Fig. 1 provides an example of pub/sub content dissemination in OppNets. Contents are organized into a set of channels, such as traffic news, sports news, weather forecast, and so on, and nodes can obtain subscribed contents through their opportunistic contacts with each other. For example, node 1, interested in job advertisements and financial news publishes "traffic jam in Temple University" to the network. Since node 2, subscribed to the traffic news channel, is in contact with node 1, and because "traffic jam in Temple University" belongs to the traffic news channel, node 2 can obtain this content from node 1.

The above example only illustrates the situation in which nodes in the network can obtain subscribed contents from each other without giving anything back to the counterpart, which means they are cooperative. In this paper, we argue that nodes in the network are to be inherently selfish rather than cooperative. This is because nodes in OppNets are controlled by rational humans, who by nature often exhibit selfish behavior in reality [10], [11], [12], [13]. If a node is selfish, it only aims to maximize its own revenues, and will not be willing to share its resources (e.g., memory space, transmission bandwidth, energy) to provide network services to others. Therefore, appropriate incentive schemes need to be designed for selfish OppNets, in order to stimulate cooperation between nodes in selfish OppNets.

- *H. Zhou is with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China, and the College of Computer and Information Technology, China Three Gorges University, Yichang 443002, China. E-mail: zhouhuan117@gmail.com.*
- *H. Zhao and J. Chen are with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China. E-mail: leaderzhy@gmail.com, jmchen@iipc.zju.edu.cn.*
- *J. Wu and S. Tang are with the Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19128. E-mail: jiewu@temple.edu, tangshaojie@gmail.com.*
- *C. Chen is with the Nokia Research Center Beijing, China. E-mail: canfeng.chen@ieee.org.*

Manuscript received 2 Mar. 2014; revised 16 Aug. 2014; accepted 26 Sept. 2014. Date of publication 30 Oct. 2014; date of current version 7 Aug. 2015.
Recommended for acceptance by H. Wu.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPDS.2014.2366474

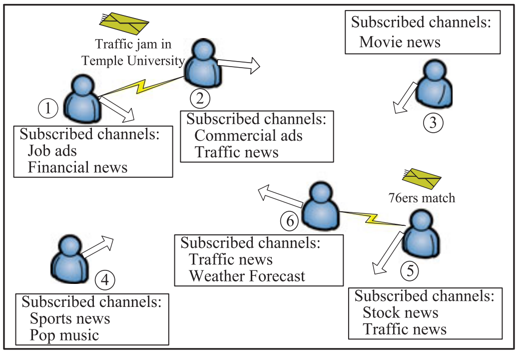


Fig. 1. An example of Pub/sub content dissemination in OppNets.

Incentive schemes have been extensively studied in the Internet, mobile ad hoc networks (MANETs), and peer-to-peer (P2P) networks [14], [15]. Most existing works addressing selfishness can be classified into three categories: reputation-based [14], credit-based [15], and Tit-For-Tat (TFT)-based schemes [16], [17]. Note that the TFT-based scheme does not require the detection of trusted nodes, secure hardware, or a centralized credit center; it only requires the principle of equal amounts of service, which is easy to achieve in intermittently-connected OppNets. Therefore, in this paper, we propose an incentive-driven and freshness-aware pub/sub content dissemination scheme, *ConDis*, for selfish OppNets, in which TFT is chosen as the incentive scheme to deal with selfish behaviors of nodes. It is worth noticing that *ConDis* also takes the freshness of content into consideration; this is because the satisfaction of a published content for a certain subscribed node is different at different times. For example, if node 2 obtains “traffic jam in Temple University” when it is published by node 1, then the satisfaction will be very high. However, if node 2 obtains it when it is going to expire, the satisfaction will be very low. This is because fresh contents are more useful for nodes.

Under the TFT scheme, any pair of nodes must offer other pairs an equal amount of contents while they are trading with each other. Therefore, in order to trade with others, nodes have to use their storage to store content for others [13], [18]. Since the storage space of nodes is limited, and the importance of the content is different for different nodes, nodes in the network have to choose some contents which are useful for them to store. An interesting optimization problem then arises: in order to get as many subscribed contents as possible, and guarantee the freshness of these contents, how should nodes act in the presence of TFT when the storage space is limited?

To answer this question, our proposed scheme, *ConDis*, introduces a novel content exchange protocol when nodes are in contact. Specifically, during each contact, the exchange order is decided by the content utility, which represents the usefulness of a content for a certain node. Intuitively, the utility of a certain content for a certain node should depend on two kinds of the current node’s 1-hop neighbors. The first kind are the nodes which are interested in this content, yet have not obtained this content; the second kind are the nodes which are not interested in this content, and have not obtained this content. From the perspective of real life trading, the first kind of nodes will absolutely choose to trade with the current node, because they are interested in this content and have not obtained it. The second kind of nodes may also choose to trade with the current node if many of their 1-hop neighbors are interested in this content. Here, we denote the value contributed by the first kind as the direct subscribed value, and the value contributed by the second kind as the indirect subscribed value. Therefore, in the definition of content utility, we only need to take these two kinds of nodes into consideration, and the objective of the nodes in the network is to maximize the utility of the content inventory stored in their buffer.

To summarize, the novelty and contributions of this paper are as follows:

1) We propose *ConDis*, an incentive-driven and freshness-aware pub/sub content dissemination scheme for selfish OppNets, in which TFT is employed as the incentive scheme to deal with selfish behaviors of nodes in selfish OppNets.
2) In *ConDis*, a novel content exchange protocol is proposed when nodes are in contact. Specifically, during each contact, the exchange order is decided by the content utility, which includes the direct subscribed value and the indirect subscribed value, and the objective of nodes in the network is to maximize the utility of the content inventory stored in their buffer.
3) Extensive realistic trace-driven simulations are conducted to evaluate the performance of our proposed scheme, *ConDis*. The simulation result shows that *ConDis* outperforms other existing schemes.

The remainder of this paper is organized as follows. Section 2 summarizes the related work. Section 3 introduces the node contact model, channel and content model, and assumptions. System architecture of *ConDis* (Section 4.1), details about how to compute the content utility in *ConDis* (Section 4.2), and the content exchange protocol (Section 4.3) are introduced in Section 4. Section 5 evaluates the performance of our proposed scheme, *ConDis*, and other existing schemes through extensive realistic trace-driven simulations. Section 7 concludes the paper.

2 Related Work

In this section, we first introduce the related work about content dissemination in OppNets, and then we introduce the related work about the incentive schemes in OppNets.

2.1 Content Dissemination in OppNets

Early research on content dissemination in OppNets mostly relies on existing infrastructure. TACO-DTN [19] is a related solution exploiting a hybrid architecture composed of a fixed backbone and mobile nodes. Specifically, TACO-DTN introduces a concept of temporal interest expressed using profiles with corresponding temporal utility for data forwarding and buffer management. Peoplenet [20] is a hybrid system that first publishes and matches information queries over infrastructure, and then uses opportunistic contacts among mobile nodes to forward them further.

Later studies on content dissemination among mobile nodes without the help of infrastructure are closely related to the pub/sub paradigm. Research on the pub/sub paradigm in OppNets is initiated by the PodNet project [6], [21], which proposes a Podcasting architecture for OppNets. In the first version of PodNet [6], nodes only retrieve contents of the channels they subscribe to. To improve the overall performance, another version was proposed in [21], in which some strategies for nodes are designed to also store contents associated with other channels. ContentPlace [18] aims to improve the performance of Podcasting using explicit knowledge of social relationships among nodes. SocialCast [22] employs social links of participants as well, which investigates the "homophily" phenomenon [23], and assumes that users with common interests have more frequent contacts with each other. Furthermore, since the cached data may be refreshed periodically and is subject to expiration, a novel scheme was proposed in [24] to efficiently maintain freshness of the cached data.

Different from existing studies on content dissemination in OppNets, in our proposed scheme, *ConDis*, we consider nodes in the network to be inherently selfish rather than cooperative.

2.2 Incentive Schemes in OppNets

Recently, there are several incentive schemes proposed for OppNets. A novel incentive scheme called IRONMAN [25] uses pre-existing social-network information (e.g., interview, or from an online social network) to detect and punish selfish nodes, hence stimulating them to participate in the network. In [12], an incentive-aware routing protocol for OppNets was proposed to adaptively optimize individual performance while conforming to TFT constraints. In [26], a credit-based incentive system named MobiCent was proposed for data forwarding in selfish OppNets, which integrates credit and cryptographic techniques to solve the edge insertion and edge hiding attacks. In [27], a credit-based incentive scheme was proposed for content dissemination in selfish OppNets, which incorporates the incentive scheme into content dissemination in selfish OppNets with multiple interest types. However, the above two credit-based incentive schemes need a trusted third-party to manage the verification and payment services, which is hard to deploy in OppNets. An incentive content cooperation scheme was proposed in [28], to encourage node participation in OppNets; however, this work does not take the resource constraints like buffers into consideration. In [13], a utility-driven trading system, called MobiTrade, was proposed to optimize the content sharing strategy in OppNets, and derived an optimal policy to split the buffer of a node into zones allocated to each channel. This work chooses TFT as the incentive scheme to deal with selfish behaviors of nodes in OppNets, however, it does not take into account the TTL of contents. Indeed, TTL has a significant impact on the definition of content utility. If the TTL of a certain content is going to expire, the contact probability of meeting those nodes interested in this content will be low; hence the utility of this content will also be low. Therefore, an incentive-based pub/sub content dissemination scheme, called ConSub, was proposed in [8], in which the content utility is determined by the contact probability and cooperation level between the current node and its 1-hop neighbors subscribing to the associated channel.

Different from the above studies, our proposed scheme, *ConDis*, takes both the direct subscribed value and the indirect subscribed value into consideration from the perspective of real life trading. Furthermore, *ConDis* also takes into account the freshness of contents.

3 Models and Assumptions

This section gives a brief introduction to the node contact model, the channel and content model, and several assumptions in our proposed scheme.

3.1 Node Contact Model

In OppNets, nodes' contact in the network can be described as a graph $G(V, E)$, where the random contact process between a node pair i, $j \in V$ can be modeled as an edge $e_{ij} \in E$. The characteristics of an edge $e_{ij} \in E$ are mainly determined by the properties of inter-contact time among mobile nodes. Recently, some studies [29], [30] in OppNets found that the pair-wise inter-contact time in realistic traces follows an exponential distribution. Specifically, authors in [29] conduct the χ^2 hypothesis test on each contacted node pair in the *Infocom 06* [31] and *MIT Reality* [32] traces, to test whether "the pair-wise inter-contact time of nodes follows an exponential distribution." Their results show that when a large enough number of test intervals (≥ 10) is used, over 85 percent Hence, the contact between nodes i and j becomes a homogeneous Poisson process with a contact frequency of λ_{ij}.

3.2 Channel and Content Model

Nodes in the network need to express their interests towards different kinds of contents in a certain way, and accordingly subscribe to those channels. They may subscribe to one or several channels. Here, we identify contents of those channels based on the matching between the keywords of subscription and the description of the channel. Since this paper focuses on introducing the incentive-driven and freshness-aware pub/sub content dissemination scheme into a selfish network environment, we do not take into account the detailed matching process of channel and content. The matching model for these keywords is suitable to the pub/sub scheme discussed in this paper, because nodes in the network may have no idea what they are looking for, but just have interest in the contents which match some keywords. For simplicity, we assume that the total number of channels in the network is C, and each content subscribed by nodes in the network can only be described by one channel. Furthermore, we assume that each node decides its subscribed channel (s) at the beginning of the network, and will not change it (them) in a long period.

In the pub/sub scheme, each node can be the subscriber or the publisher, or both the subscriber and the publisher. Each published content includes (d, c, T_d, T), while d is the sequence number of the content, c is the sequence number of the associated channel, T_d is the generation time of the content, and T is the Time-To-Live (TTL) of the content, which includes several units of time. Each published content also includes some keywords to describe it. After publishing a content into the network, each node will store the published content to its buffer, so as to trade with others.

3.3 Assumptions

Without loss of generality, we assume that buffer spaces of nodes in the network are all B. Furthermore, contents generated in the network have the same volume capacity; thus when nodes exchange contents with each other, we can count the total content volume by counting the number of contents. Finally, during each contact, we assume that the contact duration between pair-wise nodes is long enough to complete the content exchange.

4 Proposed Scheme *ConDis*

In this section, we first introduce the architecture of our proposed scheme, *ConDis*, and then provide the detailed computing method of the content utility, which includes the contact probability and the expected delay. Finally, we introduce the content exchange protocol when nodes are in contact.

4.1 The Architecture of ConDis

In this part, we give the architecture of our proposed scheme, *ConDis*. It includes the following four parts:

4.1.1 *Subscribed Channel Manager*

Each node's subscribed channel manager keeps the channel information subscribed by nodes in the network, and merges it with its own subscribed channels to form a list of subscribed channels. Furthermore, each node also needs to keep the pair-wise inter-contact time with other nodes in the network.

4.1.2 *Buffer State Collector*

Each node's buffer state collector collects the content information stored at nodes in the network, and merges it with its own stored content information to form a list of its buffer state. Specifically, we assume that each content published in the network is associated with a descriptive metadata, which includes its associated node, sequence number (d), associated channel (i.e., c), generation time (T_d), and Time-To-Live. In order to reduce the communication and storage overhead, we assume that each node's buffer state collector only needs to collect these metadatas.

4.1.3 *Content Utility Estimator*

Each content has an initial freshness value V when it is published by a certain node. Without loss of generality, we assume that the freshness value of a certain content for a certain subscribed node will decrease as time passes, while the corresponding freshness value for a certain unsubscribed node will be 0. Furthermore, when the TTL of a certain content is expired, the corresponding freshness value for a certain subscribed node will be 0. Therefore, the freshness value of a certain content d for a subscribed node i can be expressed as:

$$v_i(d) = \frac{R_d V}{T}, \tag{1}$$

where R_d is the remaining valid time of content d, which can be calculated by using the current time, the generation time T_d and TTL. Here, T is the TTL of content d.

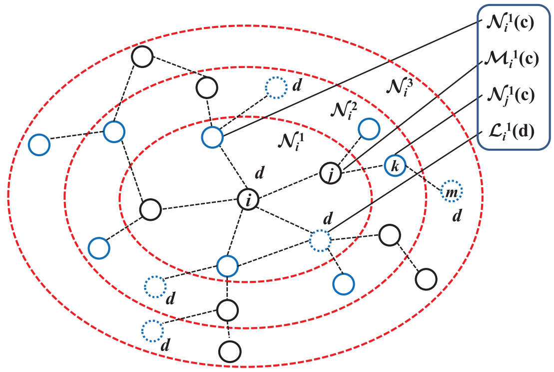


Fig. 2. Illustrating the utility definition of a certain content d in channel c for node i. The dotted line between two nodes indicates that there is a link between the two nodes.

Under the TFT scheme, in order to trade with others, nodes in the network need to store some contents in their buffer. However, since the importance of a certain content is different for different nodes, and the size of their buffer is limited, nodes have to choose some contents which are useful for them to store. Here, the usefulness of a certain content for a certain node depends on its neighbors. As shown in Fig. 2, we denote $\mathcal{N}_i^1$, $\mathcal{N}_i^2$ and $\mathcal{N}_i^3$ as node i's 1-hop, 2-hop and 3-hop neighbors, respectively, and divide the set of node i's 1-hop neighbors into three kinds: the first kind are the nodes which subscribe to channel c and have not obtained content d, denoted as $\mathcal{N}_i^1(c)$; the second kind are the nodes which do not subscribe to channel c and have not obtained content d, denoted as $\mathcal{M}_i^1(c)$; and the third kind are the nodes which have already obtained content d, denoted as $\mathcal{L}_i^1(d)$. From the perspective of real-life trading, the first kind of nodes will absolutely choose to trade with node i, because they subscribe to channel c and have not obtained content d. Therefore, in the definition of content utility, we denote the value contributed by this part as the direct subscribed value. The second kind of nodes may also choose to trade with node i if many of their 1-hop neighbors subscribe to channel c. This is because these nodes can use content d to trade with their 1-hop neighbors. Similarly, we denote the value contributed by this part as the indirect subscribed value. The third kind of nodes will absolutely not choose to trade with node i, because they have already obtained content d. As a result, in the utility definition of content d for node i, we only need to take into account the first and the second kind of node i's 1-hop neighbors. Hence, the utility of a content d in channel c for node i is defined as:

$$U_i(d) = wU_{di}(d) + (1-w)U_{indi}(d), \tag{2}$$

where $U_{di}(d)$ is the direct subscribed value, and $U_{indi}(d)$ is the indirect subscribed value; w in the range of $[0, 1]$; here, w and $1-w$ represent the weight of the direct subscribed value, and the indirect subscribed value, respectively.

In order to obtain the expression of $U_i(d)$, node i needs to obtain the direct subscribed value $U_{di}(d)$, and the indirect subscribed value $U_{indi}(d)$ first. As shown in Fig. 2, for node $j \in \mathcal{N}_i^1(c)$, node j will request content d from node i only when the following three conditions are met.

1) Node j has not yet received content d until node j comes into contact with i.
2) Node i can deliver content d to node j before this content is out of time.
3) When content d is delivered to node j, node i should guarantee that this content is fresh.

Taking the above three conditions into consideration, the direct subscribed value $U_{di}(d)$ can be expressed as:

$$U_{di}(d) = \sum_{j\in\mathcal{N}_i^1(c)} \int_0^{R_d} \frac{(R_d - t)V}{T} f_{X_{ij}}(t) \prod_{k\in\mathcal{L}_j^1(d)} [1 - Pr_{jk}(t)]dt, \tag{3}$$

where $\mathcal{N}_i^1(c)$ is the set of node i's 1-hop neighbors which subscribe to channel c, and have not obtained content d; $\mathcal{L}_j^1(d)$ is the set of node j's 1-hop neighbors which have already obtained content d; X_{ij} is the time needed for transmitting content d from node i to node j, and $f_{X_{ij}}(t)$ is the probability distribution function (PDF) of X_{ij}; $Pr_{jk}(t)$ is the contact probability between nodes j and k in a certain time t.

Similarly, as shown in Fig. 2, for node $j \in \mathcal{M}_i^1(c)$ and $k \in \mathcal{N}_j^1(c)$, node j will request content d from node i only when the following three conditions are met.

1) Node k has not yet received content d until node j comes into contact with k.
2) Node j can deliver content d to node k before this content is out of time.
3) When content d is delivered to node k, node j should guarantee that this content is fresh.

Taking the above three conditions into consideration, the indirect subscribed value $U_{indi}(d)$ can be expressed as:

$$U_{indi}(d) = \sum_{j\in\mathcal{M}_i^1(c)} \sum_{k\in\mathcal{N}_j^1(c)} \int_0^{R_d} \frac{(R_d - t)V}{T} f_{X_{ik}^j}(t) \prod_{m\in\mathcal{L}_k^1(d)} [1 - Pr_{km}(t)]dt, \tag{4}$$

where $\mathcal{M}_i^1(c)$ is the set of node i's 1-hop neighbors which do not subscribe to channel c, and have not obtained content d; $\mathcal{N}_j^1(c)$ is the set of node j's 1-hop neighbors which subscribe to channel c, and have not obtained content d; X_{ik}^j is the time needed for transmitting content d from i to k through j; $f_{X_{ik}^j}(t)$ is the PDF of X_{ik}^j; $Pr_{km}(t)$ is the contact probability between nodes k and m in a certain time t.

Substituting Eq. (3) together with Eq. (4) into Eq. (2), the content utility $U_i(d)$ of content d in channel c for node i can be expressed as:

$$U_i(d) = w \sum_{j\in\mathcal{N}_i^1(c)} \int_0^{R_d} \frac{(R_d - t)V}{T} f_{X_{ij}}(t) \prod_{k\in\mathcal{L}_j^1(d)} [1 - Pr_{jk}(t)]dt + (1-w) \sum_{j\in\mathcal{M}_i^1(c)} \sum_{k\in\mathcal{N}_j^1(c)} \int_0^{R_d} \frac{(R_d - t)V}{T} f_{X_{ik}^j}(t) \prod_{m\in\mathcal{L}_k^1(d)} [1 - Pr_{km}(t)]dt. \tag{5}$$

In order to obtain the content utility $U_i(d)$ of content d in channel c for node i, we have to compute $f_{X_{ij}}(t)$, $Pr_{jk}(t)$, $f_{X_{ik}^j}(t)$, and $Pr_{km}(t)$ first. Therefore, in the next part, we will introduce the detailed computing method of $f_{X_{ij}}(t)$, $Pr_{jk}(t)$, $f_{X_{ik}^j}(t)$, and $Pr_{km}(t)$.

4.1.4 Buffer Manager

Since the usefulness of a certain content is different for different nodes, and the size of the buffer is limited, nodes in the network have to choose some useful contents to store. Therefore, in order to get as many subscribed contents as possible, and to guarantee the freshness of these subscribed contents, the objective of the nodes in the network is to maximize the expected content utility in their buffer, which can be expressed as:

$$Max\, U_i = \sum_{c=1}^{C} \left(\sum_{d\in\theta(c)} U_i(d) - \sum_{d\in\phi(c)} U_i(d) \right), \tag{6}$$

where U_i is the utility function of node i; C is the total number of channels; and $\theta(c)$ and $\phi(c)$ are the set of contents associated to channel c in their buffer after and before exchange, respectively.

The cache management of a node is mainly based on the content utility. When a new content is sent to the current node, it places the content into the buffer corresponding to its content utility. If the buffer size of the current node is full, the content with higher utility can seize the cache position that is occupied by the content with lower utility. Moreover, the expired contents will be deleted directly from the buffer, even if there is unoccupied space.

4.2 Computing Content Utility

In this part, we introduce how to compute $f_{X_{ij}}(t)$, $Pr_{jk}(t)$, $f_{X_{ik}^j}(t)$, and $Pr_{km}(t)$ in the definition of content utility.

As introduced in Section 3, in this paper we also assume that the pair-wise inter-contact time in realistic traces follows an exponential distribution. The contact frequency λ_{ij} between nodes i and j is indicated by the contact rate, and can be computed by the following time average method:

$$\lambda_{ij} = \frac{n}{\Sigma_{l=1}^{n} T_{ij}^{l}}, \tag{7}$$

where $T_{ij}^1, T_{ij}^2, \ldots, T_{ij}^n$ are pair-wise inter-contact time samples between nodes i and j.

Thus, the probability distribution function of the pair-wise inter-contact time X_{ij} between nodes i and j can be expressed as:

$$f_{X_{ij}}(x) = \lambda_{ij} e^{-\lambda_{ij} x}. \tag{8}$$

Then, the contact probability between nodes i and j in a certain time t can be expressed as:

$$Pr_{ij}(t) = Pr(X_{ij} \le t) = \int_0^t f_{X_{ij}}(x)dx = 1 - e^{-\lambda_{ij} t}. \tag{9}$$

Similarly, the contact probability between nodes k and m in a certain time t can be expressed as:

$$
\begin{aligned}
Pr_{km}(t) = Pr(X_{km} \le t) &= \int_0^t f_{X_{km}}(x)dx \\
&= 1 - e^{-\lambda_{km} t}.
\end{aligned} \tag{10}
$$

After being given the expression of $Pr_{jk}(t)$, $f_{X_{ik}^j}(t)$, and $Pr_{km}(t)$, we give the expression of $f_{X_{ij}}(t)$. Note that the total time to transfer a certain content from i to k through j is $X_{ik}^j = X_{ij} + X_{jk}$, while X_{ij} is the time needed for transmitting the content from i to j, and X_{jk} is the time needed for transmitting the content from j to k. Since the pair-wise inter-contact time follows an exponential distribution, we obtain that X_{ij} follows the exponential distribution with the parameter λ_{ij}. According to the memoryless property of the exponential distribution, X_{jk} also follows the exponential distribution, but with the parameter λ_{jk}. Therefore, based on Eq. (8), the PDF $f_{X_{ik}^j}(t)$ can be calculated as:

$$
\begin{aligned}
f_{X_{ik}^j}(t) &= f_{X_{ij}}(t) \otimes f_{X_{jk}}(t) \\
&= \lambda_{ij}\lambda_{jk} \int_0^t e^{-(\lambda_{ij}-\lambda_{jk})x} e^{-\lambda_{jk}x} dx \\
&= \frac{\lambda_{ij}\lambda_{jk}(e^{-\lambda_{ij}t} - e^{-\lambda_{jk}t})}{\lambda_{jk} - \lambda_{ij}},
\end{aligned} \tag{11}
$$

where $\otimes$ is the convolution operator; $f_{X_{ij}}(t)$ is the PDF of the inter-contact time X_{ij} between nodes i and j, and $f_{X_{jk}}(t)$ is the PDF of the inter-contact time X_{jk} between nodes j and k.

Substituting Eq. (8), Eq. (9), Eq. (10), together with Eq. (11), into Eq. (5), we can express the content utility $U_i(d)$ of content d in channel c for node i as follows:

$$
\begin{aligned}
&U_i(d) = w \sum_{j \in \mathcal{N}_i^1(c)} \int_0^{R_d} \frac{(R_d - t)V}{T} \lambda_{ij} e^{-\lambda_{ij}x} \\
&\prod_{k \in \mathcal{L}_j^1(d)} e^{-\lambda_{jk}t} dt + (1-w) \sum_{j \in \mathcal{M}_i^1(c)} \sum_{k \in \mathcal{N}_j^1(c)} \int_0^{R_d} \\
&\frac{(R_d - t)V}{T} \frac{\lambda_{ij}\lambda_{jk}(e^{-\lambda_{ij}t} - e^{-\lambda_{jk}t})}{\lambda_{jk} - \lambda_{ij}} \prod_{m \in \mathcal{L}_k^1(d)} e^{-\lambda_{km}t} dt.
\end{aligned} \tag{12}
$$

4.3 The Content Exchange Protocol

Based on the above models, assumptions, and definitions, our proposed scheme, *ConDis*, is outlined below. We take two nodes i and j as an example. When i meets j, node i needs to decide whether or not to exchange contents in its buffer with the latter. If j has some useful contents in its buffer for i, for example, some contents which are subscribed by i, or some contents which can increase the content utility in i's buffer, then i will choose to exchange contents with j. From the perspective of real life trading, node i will give priority to obtaining its subscribed contents with higher freshness value, and then aims to obtain contents which can increase its own expected content utility in its buffer. Then, our proposed scheme *ConDis* works as follows, in five steps:

1) When node i meets node j, node i first sends a control message to node j, which includes its collected network state information (including its collected subscribed channel information, the contact rate information, and buffer state information). Node j also sends a similar control message to node i.
2) When node i receives the control message from node j, it first updates the past stored control messages with the new received control message from node j. Then, it creates a set S_i to denote the set of metadata of contents that are stored at j, and a set L_i to denote the set of contents that are available at j but not at i, i.e., $L_i = S_i - (S_i \cap S_j)$. Based on the stored control messages, node i calculates the freshness value of contents in L_i which meet its interest, and the content utility of all contents in L_i according to Eq. (12). Similarly, node j does so in a similar way.
3) Node i checks if there are any contents in L_i matching its interest. Let $L_i^{'}$ denote the set of such contents. Then, node i adds them into the candidate request list R_i in the decreasing order of the freshness value with a high priority. After determining the subscribed contents, node i then determines the contents not matching its interest. Let $L_i^{''}$ denote the set of contents in L_i that does not include contents matching its interest. Node i then adds them into the candidate request list R_i in the decreasing order of the content utility with a low priority. Accordingly, node j does so in a similar way, and obtains the candidate request list R_j.
4) After determining the candidate request lists of each other, nodes i and j will trade with each other to obtain subscribed contents, and these contents can increase the total content utility in their buffer from each other. In the decreasing order of priority and content utility, nodes i and j will trade contents in the candidate request list of each other, one by one. After obtaining a new content, they will store it in their buffer according to its content utility. They will finish the trading until one side does not have contents, which can increase the total content utility in the buffer for the other side.
5) After finishing the trading with each other, nodes i will update L_i, and send a new control message to j. Node j does so in a similar way.

Fig. 3 gives an example about the content exchange process between nodes i and j in *ConDis*. As introduced above, R_i and R_j represent the candidate request list of node i and node j, respectively. For simplicity, we use small numbers: $1, 2, \ldots, 13$, to represent the sequence number of contents stored in nodes i and js' buffers. Here, $v_j(1)$ denotes the freshness value of content 1 for subscribed node j, and $v_j(1) \ge v_j(2) \ge v_j(3)$. Similarly, $v_i(8)$ denotes the freshness value of content 8 for subscribed node i, and $v_i(8) \ge v_i(9)$. $U_j(4)$ denotes the utility of content 4 for node j, and $U_j(4) \ge U_j(5) \ge U_j(6) \ge U_j(7)$. Similarly, $U_i(10)$ denotes the utility of content 10 for node i, and $U_i(10) \ge U_i(11) \ge U_i(12) \ge U_i(13)$. According to the content exchange protocol introduced above, nodes i and j start the trading from the top of each other's candidate request list, which means they

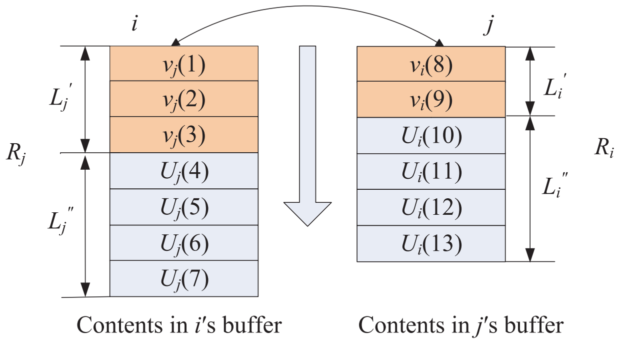


Fig. 3. The content exchange process between nodes i and j in *ConDis*.

will give priority to obtaining subscribed contents which have higher freshness values. It is worth noticing that if $|L_i^{'}| = |L_j^{'}|$, nodes i and j will finish the trading of their subscribed contents. However, if $|L_i^{'}| \neq |L_j^{'}|$, in order to obtain other subscribed contents, one side which does not have further subscribed contents for the other side needs to provide contents, which can increase the total content utility in the buffer for the other side.

4.4 The Overhead of Maintaining Network State Information

According to the definition of content utility in Section 4.1, in order to obtain the direct subscribed value, each node needs to collect the network state information within its two-hop range. Similarly, in order to obtain the indirect subscribed value, each node needs to collect the network state information within its three-hop range. Therefore, in order to calculate the utility of each content, each node needs to collect the network state information within its three-hop range. The related network state information includes:

1) The pair-wise inter-contact time of each edge within the three-hop range;
2) The channel set subscribed by each node within the three-hop range;
3) The buffer state of each node within the three-hop range.

The related network state information mentioned above is forwarded to each node in the network hop by hop. As shown in Fig. 2, in order to transmit node k's network state information to node i, node k firstly forwards its network state information to node j, and later node j delivers the information to node i. Therefore, in order to maintain network state information within the r-hop range of each node, we have the following theorem:

Theorem 1. *For each node, the communication overhead of maintaining network state information within the r-hop range is $\Omega(rO^r)$, and the corresponding storage overhead is $\Omega(O^r)$, where O is a graph-dependent invariant.*

Proof. In OppNets, nodes' contact in the network can be described as a graph $G(V,E)$. Let D be the maximum node degree in graph $G(V,E)$. Without loss of generality, we assume that $G(V,E)$ is connected, and $D \geq 2$. As shown in Fig. 2, we take node i as an example. The r-hop range network of node i is a subgraph of $G(V,E)$, we have $|N_i^1| \leq D$, and $|N_i^r| \leq D(D-1)^{r-1}$.

For $\forall j \in N_i^r$, the range between nodes i and j is at least r hops. Therefore, for node i, the maximum communication overhead of maintaining network state information within the r-hop range is $\sum_{h=1}^{r} hD(D-1)^{h-1} = \frac{Dr(D-1)^r}{D-2} + \frac{D[1-(D-1)^r]}{(D-2)^2}$, that is $\Omega(rO^r)$, where O is an invariant related to D.

Let n_r^i be the total number of nodes within r-hop range of node i, then $n_r^i \leq \sum_{h=1}^{r} |N_i^h| = \sum_{h=1}^{r} D(D-1)^{h-1} = \frac{D[(D-1)^r-1]}{D-2}$, which means there are at most $\frac{D[(D-1)^r-1]}{D-2}$ nodes within the r-hop range of node i. Therefore, for node i, the maximum overhead of storing network state information within the r-hop range is $a \times \frac{D[(D-1)^r-1]}{D-2}$, that is $\Omega(O^r)$, where a is the overhead of storing network state information of one node, and O is an invariant related to D and a. □

From Theorem 1, it can be found that when the range r increases, the overhead of maintaining network state information (both the overhead of communication and storage) increases at an exponential rate. The resource of nodes in the network is limited, so the range of network state information each node maintains cannot be too large. Extensive studies illustrate that OppNets exhibit the "small world" phenomenon, which means that the diameters of OppNets are usually small [33], [34], [35], [36], [37]. Therefore, it is reasonable that each node in the network maintains network state information within the range of a few hops.

5 Performance Evaluation

In this section, we evaluate the performance of our proposed scheme, *ConDis*, in selfish OppNets. Specifically, we compare it with the following three existing schemes:

1) *Podcasting* [21]. Nodes receive all contents which their neighbors subscribe, and randomly discard contents when their buffer is full.
2) *MobiTrade* [13]. Each node defines a buffer quota for each channel based on the past reward of the channel, and adaptively manages its buffer according to the buffer quota.
3) *ConSub* [8]. Each node exchanges contents according to the content utility, which is determined by the contact probability and the cooperation level between the current node and its one-hop neighbors subscribing to the associated channel. During each contact, the objective of nodes in the network is to maximize the utility of the content inventory stored in their buffer.

We use two experimental traces, *Infocom 06* [31] and *MIT Reality* [32], collected from realistic environments to evaluate the performance of the above schemes. Nodes in these two traces carry bluetooth-enabled portable devices, which record contacts by periodically detecting their nearby peers. The traces cover various types of corporate environments, and have various experimental durations. The details of these two traces are summarized in Table 1.

5.1 Simulation Setup

During the whole experiment, we consider that each node generates contents that match one of the channels, and the

TABLE 1
Basic Statistics of the Traces

Trace	*Infocom 06*	*MIT Reality*
Device	iMote	Smart Phone
Network type	Bluetooth	Bluetooth
No. of internal contacts	182,951	114,046
Duration (days)	3	246
Granularity (seconds)	120	300
No. of devices	78	97
Contact frequency/pair/day	6.7	0.024

content generation rate follows a uniform distribution, while 1 content/hour means that nodes generate 1 content per hour. In particular, unless otherwise stated, we set the content generation rate as 1 content/hour, and TTL as 2 hours in the *Infocom 06* trace; while in the *MIT Reality* trace, we set the content generation rate as 1 content/day, and TTL as 24 hours. Furthermore, there are five channels in the network, and each node only expresses interest, randomly, in one channel. The size of all contents equals to $40K$, each experiment has the same TTL, and the buffer size of nodes in the network $B = 1{,}000K$. Finally, the initial freshness value of all contents in the network $V = 1$, and when TTL of a certain content is expired, the corresponding freshness value will be 0. In our simulation studies, we focus on the following three performance metrics for performance evaluation:

1) *Total freshness value*. The total freshness value of contents successfully delivered for channels subscribed by nodes in the network, which reflects the effectiveness of the scheme.
2) *Total delivered contents*. The total number of contents successfully delivered for channels subscribed by nodes in the network, which reflects the effectiveness of the scheme.
3) *Total transmission cost*. The total number of contents exchanged by nodes in the network, which reflects the energy consumption of the scheme.

5.2 Performance Comparison

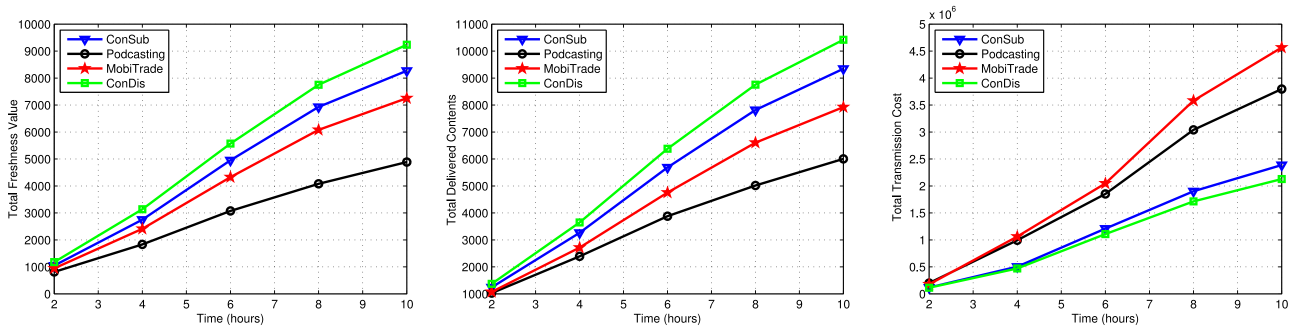
In this part, we use the *Infocom 06* trace to carry out experiments with different content generation rates (1 content/hour and 2 contents/hour), and use the *MIT Reality* trace to carry out experiments with different TTL values (24 and 48 hours), aiming to compare the performance of our proposed scheme, *ConDis*, with three other existing schemes in different traces.

5.2.1 Different Content Generation Rates

When other settings are fixed, increased content generation rate leads to more contents being published in the network. Under this situation, it is more challenging for the scheme to store right contents and discard others when the traffic in the network becomes heavy. Therefore, in this part, we change the content generation rate from 1 content/hour to 2 contents/hour in the *Infocom 06* trace, so as to compare the performance of our proposed scheme, *ConDis*, with three other existing schemes under different situations.

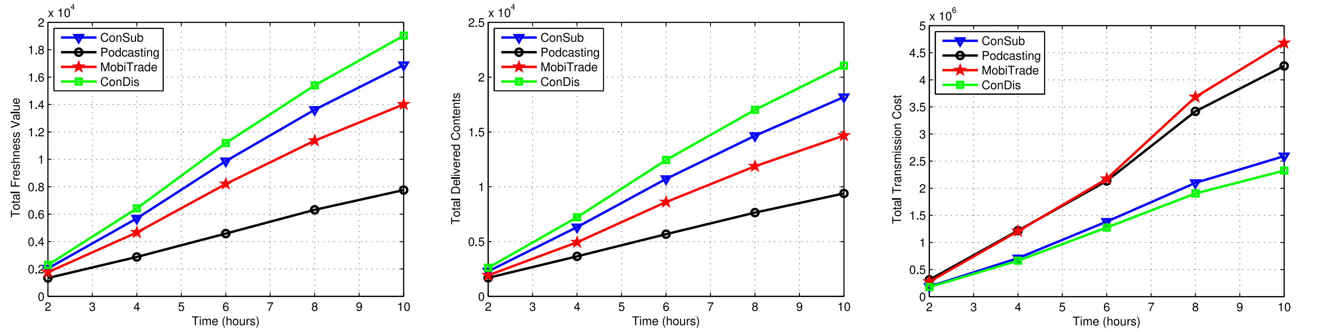
Fig. 4 shows the performance comparison of our proposed scheme, *ConDis*, with other three existing schemes in the *Infocom 06* trace when the content generation rate is 1 content/hour. It can be found that as the simulation time increases, *ConDis* outperforms three other existing schemes in terms of total freshness value, total delivered contents, and total transmission cost in the *Infocom 06* trace. Moreover, the longer the simulation time is, the better *ConDis* performs. This is because *ConDis* takes both the direct subscribed value and the indirect subscribed value into consideration from the perspective of real life trading. Furthermore, *ConDis* also takes into account the freshness of contents, and designs a novel content exchange protocol in the content exchange process when nodes are in contact. It can be also found that Podcasting performs worst in the *Infocom 06* trace. The main reason is that nodes in Podcasting receive all contents to which their 1-hop neighbors subscribe, and do not manage their buffer. Although Mobitrade and Consub take the buffer management into consideration, they both rely on the past transaction between nodes to define the utility of contents, which may be inaccurate for predicting the future trading value of contents. Moreover, they do not take the freshness of contents into consideration. Therefore, *ConDis* can achieve a better performance than Podcasting, MobiTrade, and Consub in terms of total freshness value, total delivered contents, and total transmission cost in the *Infocom 06* trace.

Fig. 5 shows the performance comparison of our proposed scheme, *ConDis*, with three other existing schemes in the *Infocom 06* trace when the content generation rate is 2 contents/hour. It can be found that when the content generation rate changes from 1 to 2 contents/hour in the *Infocom 06*



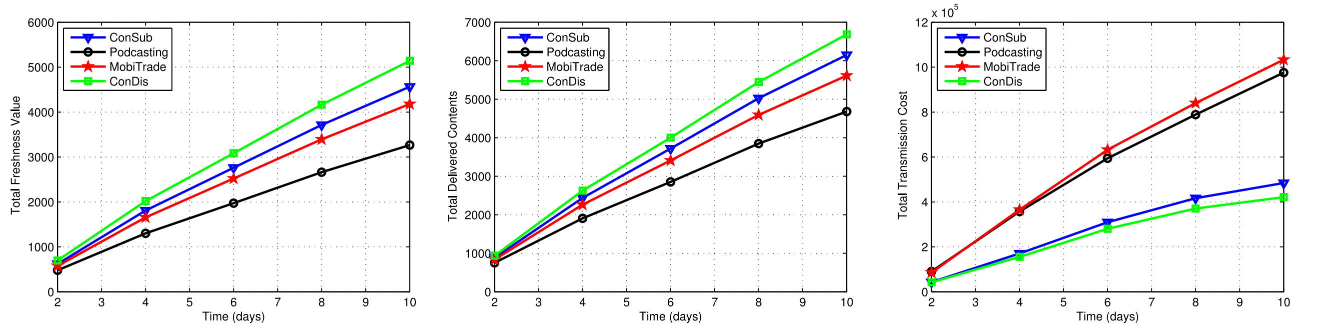
(a) Total Freshness Value (b) Total Delivered Contents (c) Total Transmission Cost

Fig. 4. Performance comparison of *ConDis* with other existing schemes in the *Infocom 06* trace when the content generation rate is 1 content/hour.



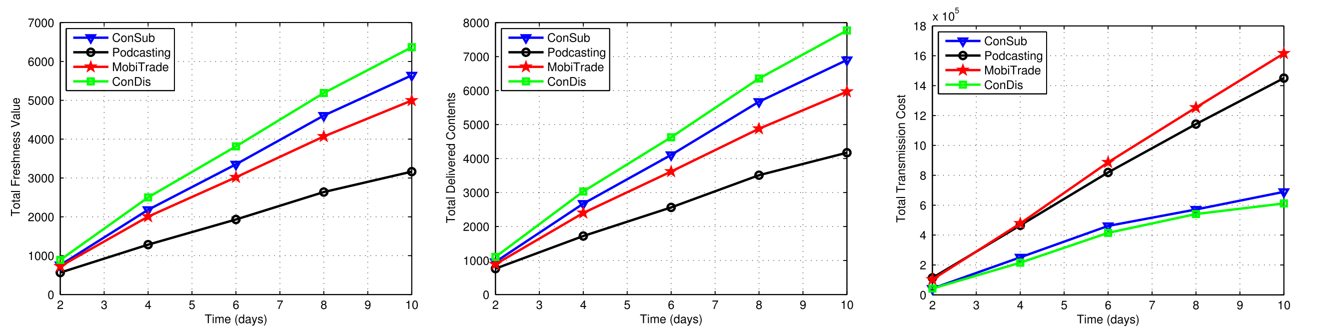
(a) Total Freshness Value (b) Total Delivered Contents (c) Total Transmission Cost

Fig. 5. Performance comparison of *ConDis* with other existing schemes in the *Infocom 06* trace when the content generation rate is 2 contents/hour.



(a) Total Freshness Value (b) Total Delivered Contents (c) Total Transmission Cost

Fig. 6. Performance comparison of *ConDis* with other existing schemes in the *MIT Reality* trace when TTL is 24 hours.



(a) Total Freshness Value (b) Total Delivered Contents (c) Total Transmission Cost

Fig. 7. Performance comparison of *ConDis* with other existing schemes in the *MIT Reality* trace when TTL is 48 hours.

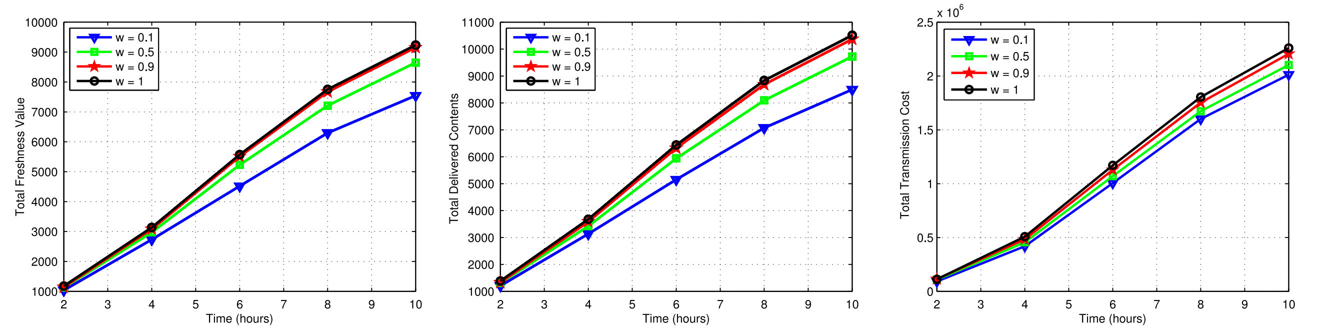
trace, total freshness value, total delivered contents and total transmission cost in different schemes all increase. It can be also found that similar to the results in Fig. 4, *ConDis* also outperforms other three existing schemes in terms of total freshness value, total delivered contents, and total transmission cost. Therefore, *ConDis* performs better in the *Infocom 06* trace when the contact generation rate increases.

To summarize, *ConDis* outperforms other schemes in terms of total freshness value, total delivered contents and total transmission cost in the *Infocom 06* trace when the content generation rate is different, which demonstrates the effectiveness of our proposed scheme. Moreover, the longer the simulation time is, the better *ConDis* performs.

5.2.2 Different TTL

When other settings are fixed and TTL increases, contents in the network have more opportunities to be delivered to the subscribed nodes, but they also add traffic into the network. Under this situation, it is more challenging for the scheme to store the right contents and discard others when the traffic in the network becomes heavy. Therefore, in this part, we change TTL from 24 to 48 hours in the *MIT Reality* trace, so as to investigate how it influences *ConDis* and other existing schemes.

From Figs. 6 and 7, it can be found that when TTL increases from 24 to 48 hours in the *MIT Reality* trace, total freshness value, total delivered contents, and total transmission cost in different schemes all increase. It can be also found that as the simulation time increases, *ConDis* again outperforms three other existing schemes in terms of total freshness value, total delivered contents, and total transmission cost in the *MIT Reality* trace. Moreover, the longer the simulation time is, the better *ConDis* performs. Podcasting also performs worst in the *MIT Reality* trace, and Consub also performs better than



(a) Total Freshness Value (b) Total Delivered Contents (c) Total Transmission Cost

Fig. 8. Performance of *ConDis* with different w in the *Infocom 06* trace.

Mobitrade in the *MIT Reality* trace. It is worth noticing that the simulation time and TTL in the *MIT Reality* trace are much larger than that in the the *Infocom 06* trace. This is because the experimental duration in the *MIT Reality* trace is much longer than that in the *Infocom 06* trace, and the contacts in the *MIT Reality* trace are much sparser than those in the *Infocom 06* trace.

To summarize, *ConDis* outperforms other schemes in terms of total freshness value, total delivered contents and total transmission cost in the *MIT Reality* trace when TTL is different, which demonstrates the effectiveness of our proposed scheme.

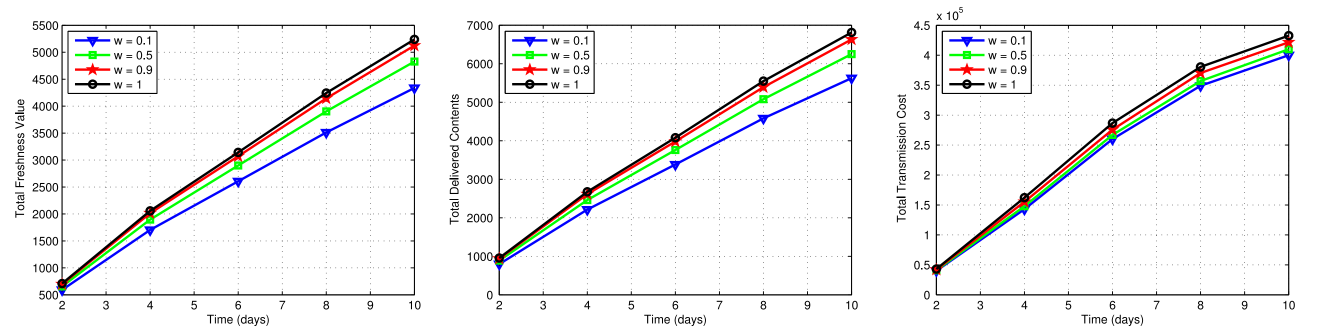
5.2.3 Impact of w

In this part, we evaluate the performance of our proposed scheme, *ConDis*, with different values of w (w = 0.1, 0.5, 0.9 and 1) using the *Infocom 06* trace and the *MIT Reality* trace, aiming to check the impact of the changing metrics w on the performance of *ConDis* in different traces.

Figs. 8 and 9 show the performance of our proposed scheme, *ConDis*, with different values of w (w = 0.1, 0.5, 0.9 and 1) in the *Infocom 06* trace and the *MIT Reality* trace, respectively. It can be found that, as w increases from 0.1 to 1, total freshness value, total delivered contents, and total transmission cost of *ConDis* all increase, not only in the *Infocom 06* trace, but also in the *MIT Reality* trace. The main reason is that w changes the balance between the direct subscribed value and the indirect subscribed value, which are determined by nodes' one-hop neighbors and two-hop neighbors, respectively. As w increases, nodes in the network will give priority to storing contents subscribed directly by their one-hop neighbors, which means nodes can obtain more contents matching their interest from the trading with their one-hop neighbors. Moreover, in order to calculate the indirect subscribed value of a certain one-hop neighbor, the current node needs to predict its two-hop neighbors' information based on its past stored control messages, which may be inaccurate. Therefore, total freshness value, total delivered contents, and total transmission cost of *ConDis* all increase when w increases.

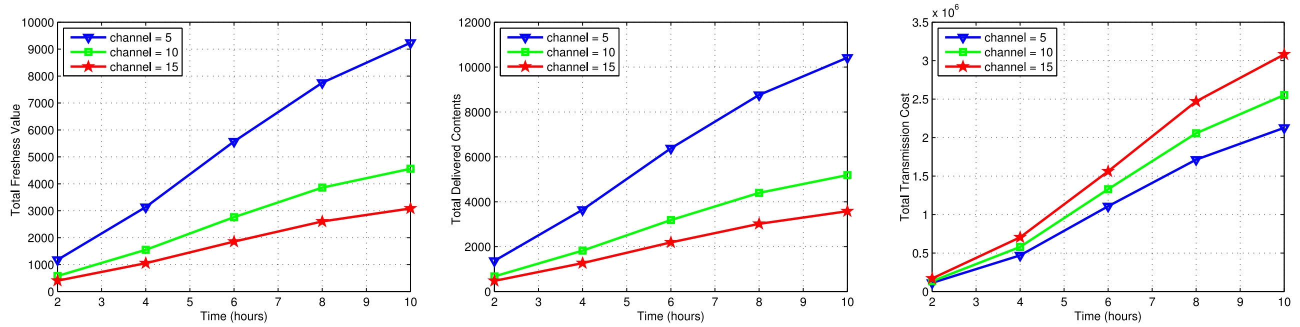
When w increases from 0.9 to 1, total freshness value, total delivered contents, and total transmission cost of *ConDis* in the *Infocom 06* trace and the *MIT Reality* trace all increase a little. Furthermore, when w increases to 1, each node in the network only needs to calculate the direct subscribed value in the content utility definition. Therefore, in order to calculate the utility of each content, each node only needs to collect the network state information within its two-hop range. According to Theorem 1, the overhead of maintaining network state information will decrease a lot when the maintaining range decreases from 3 to 2 hops. Therefore, in order to increase the performance of *ConDis*, nodes in the network should give priority to storing contents subscribed directly by their one-hop neighbors.

In summary, w has a significant impact on the performance of *ConDis*. Since it may be inaccurate for a certain node to predict its two-hop neighbors' network state information based on its past stored control messages, it is better for nodes in the network to store contents subscribed directly by their one-hop neighbors. Therefore, in order to increase the performance of *ConDis*, nodes in the network should give priority to storing contents subscribed directly by their one-hop neighbors.



(a) Total Freshness Value (b) Total Delivered Contents (c) Total Transmission Cost

Fig. 9. Performance of *ConDis* with different w in the *MIT Reality* trace.



(a) Total Freshness Value (b) Total Delivered Contents (c) Total Transmission Cost

Fig. 10. Performance of *ConDis* with different number of channels in the *Infocom 06* trace.

5.2.4 Impact of the Number of Channels

In this part, we evaluate the performance of our proposed scheme, *ConDis*, with different numbers of channels (5, 10 and 15) using the *Infocom 06* trace and the *MIT Reality* trace, aiming to check the impact of the number of channels on the performance of *ConDis* in different traces.

Figs. 10 and 11 show the performance of our proposed scheme, *ConDis*, with different numbers of channels (5, 10 and 15) in the *Infocom 06* trace and the *MIT Reality* trace, respectively. It can be found that, as the number of channels increases from 5 to 15, total freshness value, total delivered contents, and total transmission cost of *ConDis* all decrease, not only in the *Infocom 06* trace, but also in the *MIT Reality* trace. A reasonable explanation is that it is hard for nodes in the network to choose contents with higher utility for exchange when the number of channels increases. Therefore, nodes will exchange less amount of contents with others, which causes a decrease of total freshness value, total delivered contents, and total transmission cost.

To summarize, the number of channels has a significant impact on the performance of *ConDis*. If the number of channels is too large, it will obviously decrease the performance of *ConDis*; thus we should choose an appropriate number of channels according to different application scenarios.

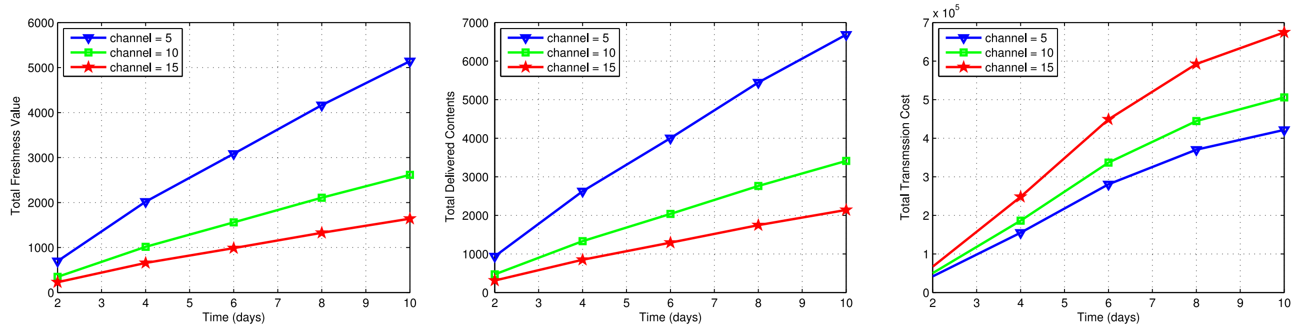
6 Discussion

Though the current design of ConDis only considers two nodes during each contact, the content exchange process in which there are more than two nodes during each contact is similar to the process when there are only two nodes during each contact. For example, if there are three nodes, i, j and k within each other's contact range. According to the content exchange protocol in Section 4.3, node i will first send control messages to j and k; j and k will do so in a similar way. After receiving the control messages from j and k, node i will calculate the freshness value of contents in j and ks' buffer which meet its interest, and the content utility of all contents in j and ks' buffer according to Eq. (12). Then, node i will determine the candidate request lists based on the freshness value of contents and the content utility. Here, the only difference is that a parameter needs to be added in the candidate request lists to represent which node contents in the candidate request lists belong to. If content in the candidate request list of i belongs to j, then node i will trade with j to obtain this content. Nodes j and k will do so in a similar way. After determining the candidate request lists, node i will trade with j and k to obtain subscribed contents, and those contents which can increase the total content utility in its buffer. In the decreasing order of priority and content utility, i will trade contents in the candidate request list with j and k, one by one. Nodes j and k will do so in a similar way. They will finish the trading until one side does not have contents, which can increase the total content utility in the buffer for the other side.

Therefore, if there are more than two nodes during each contact, the only difference is that we need to add a parameter to represent which node contents in the candidate request lists belong to, then they will trade with each other in a similar way.

7 Conclusion

In this paper, we have investigated the pub/sub content dissemination in OppNets. Considering the selfish behavior



(a) Total Freshness Value (b) Total Delivered Contents (c) Total Transmission Cost

Fig. 11. Performance of *ConDis* with different number of channels in the *MIT Reality* trace.

of nodes and the freshness of contents, we propose an incentive-driven and freshness-aware pub/sub content dissemination scheme, called *ConDis*, for selfish OppNets. In *ConDis*, we choose the TFT scheme as the incentive scheme to deal with selfish behaviors of nodes in the network. Moreover, we also propose a novel content exchange protocol when nodes are in contact. Specifically, during each contact, the exchange order is decided by the content utility, and the objective of nodes in the network is to maximize the utility of the content inventory stored in their buffer. Extensive realistic trace-driven simulations are conducted to evaluate the performance of our proposed scheme, *ConDis*, and the results show that *ConDis* is superior to other existing schemes in terms of total freshness value, total delivered contents, and total transmission cost.

Acknowledgments

This research was supported in part by NSFC under grants 61222305, 61228302, SRFDP under grant 20120101110139, 111 Program under grant B07031, NCET-11-0445, Natural Science Foundation of Hubei Province of China under grant 2014CFB145, and the Fundamental Research Funds for the Central Universities under Grants 2013QNA5013, 2013FZA5007. Part of this work was presented at IEEE MASS 2013 Jiming Chen is the corresponding author.

References

[1] J. Fan, J. Chen, Y. Du, W. Gao, J. Wu, and Y. Sun, "Geo-community-based broadcasting for data dissemination in mobile social networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 4, pp. 734–743, Apr. 2013.

[2] J. Fan, J. Chen, Y. Du, P. Wang, and Y. Sun, "DelQue: A socially-aware delegation query scheme in delay tolerant networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 5, pp. 2181–2193, Jun. 2011.

[3] J. Wu and Y. Wang, "Social feature-based multi-path routing in delay tolerant networks," in *Proc. IEEE INFOCOM*, 2012, pp. 1368–1376.

[4] H. Zhou, J. Chen, H. Zhao, W. Gao, and P. Cheng, "On exploiting contact patterns for data forwarding in duty-cycle opportunistic mobile networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 9, pp. 4629–4642, Nov. 2013.

[5] H. Zhou, H. Zhao, and J. Chen, "Energy saving and network connectivity tradeoff in opportunistic mobile networks," in *Proc. IEEE GLOBECOM*, 2012, pp. 524–529.

[6] V. Lenders, G. Karlsson, and M. May, "Wireless ad hoc podcasting," in *Proc. 4th Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw.*, 2007, pp. 273–283.

[7] F. Li and J. Wu, "MOPS: Providing content-based service in disruption-tolerant networks," in *Proc. IEEE 29th Int. Conf. Distrib. Comput. Syst.*, 2009, pp. 526–533.

[8] H. Zhou, J. Chen, J. Fan, Y. Du, and S. K. Das, "ConSub: Incentive-based content subscribing in selfish opportunistic mobile networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 669–679, Sep. 2013.

[9] 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 *Proc. IEEE 10th Int. Conf. Mobile Ad-Hoc Sensor Syst.*, 2013, pp. 333–341.

[10] A. Balasubramanian, B. Levine, and A. Venkataramani, "DTN routing as a resource allocation problem," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 373–384, 2007.

[11] L. Buttyán, L. Dóra, M. Félegyházi, and I. Vajda, "Barter-based cooperation in delay-tolerant personal wireless networks," in *Proc. IEEE Int. Symp. World Wireless, Mobile Multimedia Netw.*, 2007, pp. 1–6.

[12] U. Shevade, H. H. Song, L. Qiu, and Y. Zhang, "Incentive-aware routing in DTNs," in *Proc. IEEE Int. Conf. Netw. Protocols*, 2008, pp. 238–247.

[13] A. Krifa, C. Barakat, and T. Spyropoulos, "MobiTrade: Trading content in disruption tolerant networks," in *Proc. 6th ACM Workshop Challenged Netw.*, 2011, pp. 31–36.

[14] R. Mahajan, M. Rodrig, and D. Wetherall, and J. Zahorjan, "Sustaining cooperation in multi-hop wireless networks," in *Proc. 2nd Conf. Symp. Netw. Syst. Des. Implementation*, 2005, pp. 231–244.

[15] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *Proc. IEEE INFOCOM*, 2003, vol. 3, pp. 1987–1997.

[16] J. J. Jaramillo and R. Srikant, "Darwin: Distributed and adaptive reputation mechanism for wireless ad-hoc networks," in *Proc. ACM MobiCom*, 2007, pp. 87–98.

[17] V. Srinivasan, P. Nuggehalli, C. F. Chiasserini, and R. R. Rao, "Cooperation in wireless ad hoc networks," in *Proc. IEEE INFOCOM*, 2003, vol. 2, pp. 808–817.

[18] C. Boldrini, M. Conti, and A. Passarella, "ContentPlace: Social-aware data dissemination in opportunistic networks," in *Proc. 11th Int. Symp. Model., Anal. Simulation Wireless Mobile Syst.*, 2008, pp. 203–210.

[19] G. Sollazzo, M. Musolesi, and C. Mascolo, "TACO-DTN: A time-aware content-based dissemination system for delay tolerant networks," in *Proc. ACM 1st Int. MobiSys Workshop Mobile Opportunistic Netw.*, 2007, pp. 83–90.

[20] M. Motani, V. Srinivasan, and P. S. Nuggehalli, "Peoplenet: engineering a wireless virtual social network," in *Proc. ACM MobiCom*, 2005, pp. 243–257.

[21] M. May, V. Lenders, G. Karlsson, and C. Wacha, "Wireless opportunistic podcasting: Implementation and design tradeoffs," in *Proc. ACM 2nd ACM Workshop Challenged Netw.*, 2007, pp. 75–82.

[22] P. Costa, C. Mascolo, M. Musolesi, and G. P. Picco, "Socially-aware routing for publish-subscribe in delay-tolerant mobile ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 5, pp. 748–760, Jun. 2008.

[23] M. McPherson, L. Smith-Lovin, and J. Cook, "Birds of a feather: Homophily in social networks," *Annu. Rev. Sociol.*, vol. 27, pp. 415–444, 2001.

[24] W. Gao, G. Cao, M. Srivatsa, and A. Iyengar, "Distributed maintenance of cache freshness in opportunistic mobile networks," in *Proc. IEEE 32nd Int. Conf. Distrib. Comput. Syst.*, 2012, pp. 132–141.

[25] G. Bigwood and T. Henderson, "IRONMAN: Using social networks to add incentives and reputation to opportunistic networks," in *Proc. IEEE 3rd Int. Conf. Privacy, Security, Risk Trust, IEEE 3rd Int. Conf. Social Comput.*, 2011, pp. 65–72.

[26] B. Chen and M. Chan, "Mobicent: A credit-based incentive system for disruption tolerant network," in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.

[27] T. Ning, Z. Yang, X. Xie, and H. Wu, "Incentive-aware data dissemination in delay-tolerant mobile networks," in *Proc. IEEE 8th Annu. Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw.*, 2011, pp. 539–547.

[28] K. Srinivasan, S. Rajkumar, and P. Ramanathan, "Incentive schemes for data collaboration in disruption tolerant networks," in *Proc. IEEE GLOBECOM*, 2010, pp. 1–5.

[29] W. Gao, Q. Li, B. Zhao, and G. Cao, "Multicasting in delay tolerant networks: A social network perspective," in *Proc. 10th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2009, pp. 299–308.

[30] V. Conan, J. Leguay, and T. Friedman, "Characterizing pairwise inter-contact patterns in delay tolerant networks," in *Proc. ACM SenSys*, 2007, pp. 321–334.

[31] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau. (2009) Crawdad data set cambridge/haggle (v. 2009-05-29) [Online]. Available: http://crawdad.cs.dartmouth.edu/cambridge/haggle

[32] N. Eagle, A. Pentland, and D. Lazer, "Inferring social network structure using mobile phone data," *Proc. Nat. Academy Sci. USA*, vol. 106, pp. 15274–15278, 2009.

[33] A. Chaintreau, A. Mtibaa, L. Massoulie, and C. Diot, "The diameter of opportunistic mobile networks," in *Proc. ACM CoNEXT Conf.*, 2007, p. 12.

[34] E. M. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets," in *Proc. 8th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2007, pp. 32–40.

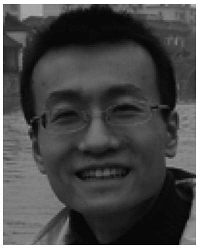
[35] Z. Dong, G. Song, K. Xie, and J. Wang, "An experimental study of large-scale mobile social network," in *Proc. 18th Int. Conf. World Wide Web*, 2009, pp. 1175–1176.

[36] S. Milgram, "The small world problem," *Psychol. Today*, vol. 2, no. 1, pp. 60–67, 1967.

[37] J. Kleinberg, "The small-world phenomenon: An algorithm perspective," in *Proc. ACM Symp. Theory Comput.*, 2000, pp. 163–170.

Huan Zhou is currently working toward the PhD degree in the Department of Control Science and Engineering, and a member of the Group of Networked Sensing and Control (IIPC-nesC) in the State Key Laboratory of Industrial Control Technology at Zhejiang University. He was a visiting scholar at the Temple University from November, 2012 to May, 2013. His research interests include data transmission and energy saving in mobile social networks and opportunistic mobile networks.

Jie Wu is the chair and a Laura H. Carnell professor in the Department of Computer and Information Sciences at Temple University. Prior to joining Temple University, he was a program director at the National Science Foundation and a distinguished professor at Florida Atlantic University. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications. He regularly publishes in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including the *IEEE Transactions on Computers*, *IEEE Transactions on Service Computing*, and *Journal of Parallel and Distributed Computing*. He was a general co-chair/chair for IEEE MASS 2006 and IEEE IPDPS 2008 and a program co-chair for IEEE INFOCOM 2011. He is currently serving as a general chair for IEEE ICDCS 2013 and ACM MobiHoc 2014, and a program chair for CCF CNCC 2013. He was a IEEE Computer Society distinguished visitor, a ACM distinguished speaker, and chair for the IEEE Technical Committee on Distributed Processing (TCDP). He is a CCF distinguished speaker. He received the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award. He is a fellow of the IEEE.



Hongyang Zhao received the BE degree in automation from Shanghai Jiaotong University in 2011. He is currently a master's student in the Department of Control Science and Engineering, and a member of the Group of Networked Sensing and Control (IIPC-nesC) in the State Key Laboratory of Industrial Control Technology at Zhejiang University. His research interests include delay tolerant networks and mobile social networks.

Shaojie Tang received the BS degree in radio engineering from Southeast University, P.R. China in 2006, and the PhD degree from the Department of Computer Science at Illinois Institute of Technology in 2012. He is a research assistant professor in the Department of Computer and Information Sciences at Temple University. His main research interests focus on wireless networks (including sensor networks and cognitive radio networks), social networks, pervasive computing, mobile cloud computing and algorithm analysis and design. He has served on the editorial board of *Journal of Distributed Sensor Networks*. He also served as a TPC member of a number of conferences such as ACM MobiHoc, IEEE SECON, IEEE ICNP, and IEEE ICPP. He is a member of the IEEE.

Canfeng Chen (S'01-M'05) received the BEng degree in information engineering and the PhD degree of engineering in signal and information processing from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2000 and 2005, respectively. Since July 2005, he has been with Nokia Research Center. He has been leading a Radio Prototyping team since January 2011 and contributing to several demo systems including mobile centric high accuracy indoor positioning, and lightweight IPv6 stack over Bluetooth low energy. He is the author of 100 technical papers in international conferences and journals, two books, and 30 patent applications. His main research interests cover Bluetooth Low Energy, Internet-of-Things systems and protocols, embedded computing, and wireless communication. He is a member of the IEEE.

Jiming Chen (M'08-SM'11) received the BSc and PhD degrees in control science and engineering from Zhejiang University in 2000 and 2005, respectively. He was a visiting researcher at INRIA in 2006, National University of Singapore in 2007, and University of Waterloo from 2008 to 2010. He is currently a full professor with the Department of Control Science and Engineering, and the coordinator of a group of Networked Sensing and Control in the State Key laboratory of Industrial Control Technology at Zhejiang University, China. His research interests include estimation and control over sensor network, sensor and actuator network, coverage and optimization in sensor network. He currently serves associate editors for several international Journals including the *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Transactions on Industrial Electronics*, and *IEEE Networks*. He is a guest editor of the *IEEE Transactions on Automatic Control*, *Computer Communication (Elsevier)*, *Wireless Communication and Mobile Computer (Wiley)*, and *Journal of Network and Computer Applications (Elsevier)*. He also serves as a co-chair for Ad hoc and Sensor Network Symposium, IEEE Globecom 2011, general symposia co-chair of ACM IWCMC 2009 and ACM IWCMC 2010, WiCON 2010 MAC track co-chair, IEEE MASS 2011 Publicity co-chair, IEEE DCOSS 2011 Publicity co-chair, IEEE ICDCS 2012 Publicity co-chair and TPC member for IEEE ICDCS 2010, IEEE MASS 2010, IEEE SECON 2011, IEEE INFOCOM 2011, IEEE INFOCOM 2012, IEEE ICDCS 2012, etc. He is a senior member of the IEEE.