

# A Direction Entropy-Based Forwarding Scheme in an Opportunistic Network

MinSeok Jeon, Sun-Kyum Kim, Ji-Hyeun Yoon, JunYeop Lee, Sung-Bong Yang\*

Department of Computer Science, Yonsei University, Seoul, Korea

zenoshock@yonsei.ac.kr, skyum@yonsei.ac.kr, ychick@yonsei.ac.kr, kotzyi@yonsei.ac.kr, sbyang@cs.yonsei.ac.kr

## Abstract

In an opportunistic network, one of the most challenging issues is the equilibrium of the network traffic and transmission delay for forwarding messages. To resolve this problem, we propose a new forwarding scheme, called the direction entropy-based forwarding scheme (DEFS), using the main direction and direction entropy based on the information collected about the directions of the nodes in the network. Since each node sends a message to another node with a different location and less direction entropy, DEFS utilizes those nodes that are more likely to travel to various locations to forward the messages to the destination nodes. Experiments were performed on the network simulator NS-2. The results show that DEFS provides better balance than the typical forwarding schemes, such as Epidemic, PROPHET, and WAIT.

**Category:** Ubiquitous computing

**Keywords:** Opportunistic networks; Information entropy; Direction; Forwarding

## I. INTRODUCTION

Opportunistic networks (OPPNETs, also known as pocket switched networks) have become one of the emerging wireless communication techniques due to the appearance of smart mobile devices such as cell phones and tablets. OPPNETs enable mobile users to communicate with their social contacts [1]. However, the frequent disconnection and sparse density in OPPNETs cause virtually no complete paths to be found between the source and destination. Such a network falls into the category of mobile ad hoc networks (MANETs) and delay-tolerant networks (DTNs) [2].

In OPPNETs, nodes communicate in multiple hops and relay nodes forward messages to other nodes. Furthermore, forwarding does not work on-the-fly, because the relay nodes store the messages when no forwarding

opportunity exists. A node should use any other node with connection opportunities to forward the messages [3]; this forwarding process is called a store, carry, and forwarding scheme.

Fig. 1 illustrates an example of message forwarding in OPPNETs. The arrows indicate the moving directions of the nodes and the circles are their communication ranges. The source node  $S$  wants to send a message to the destination node  $D$ . However, there may be no complete route between  $S$  and  $D$ . Thus,  $S$  sends its message to its neighbor node  $N_j$ , which carries the message until it encounters node  $N_i$ , to which it forwards the message and then  $N_i$  carries the message until it meets the destination  $D$ . As a result,  $S$ 's message is successfully delivered to the destination node  $D$ .

The problem of message forwarding is thus the selection of proper relay nodes to which the message (or dupli-

**Open Access** <http://dx.doi.org/10.5626/JCSE.2014.8.3.173>

<http://jcse.kiise.org>

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 23 May 2014; Revised 13 August 2014; Accepted 28 August 2014

\*Corresponding Author

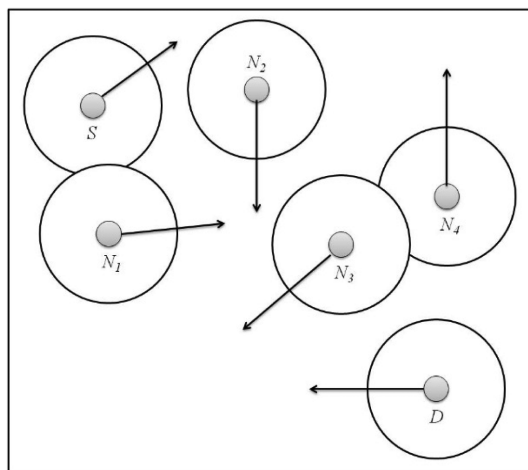


Fig. 1. Message forwarding in opportunistic networks (OPPNETs).

cate of the message for a multi-copy scheme) should be sent [4].

The representative social-unaware scheme, Epidemic [5], shows the highest network traffic, since it duplicates messages to any nodes without social information.

On the other hand, the other social-unaware scheme, WAIT, has the longest transmission delay and the lowest network traffic, because it uses a single copy of the message and waits until the source node meets the destination.

Even if PROPHET [6] is a social-aware scheme, it also has higher network traffic, because it still spreads the message to the nodes with a higher delivery predictability. It is very difficult for these schemes to optimize the equilibrium between the network traffic and transmission delay. Thus, it is important to select proper relay nodes as the next hops and to use an appropriate number of copies of the message considering both the network traffic and transmission delay.

To resolve this problem, we propose a new forwarding scheme, called the direction entropy-based forwarding scheme (DEFS), in order to achieve equilibrium between the network traffic and transmission delay. People usually move around with different speeds, locations, and directions. If we give a message to people going to different locations to which we may not go ourselves, we have a better chance of delivering the message to the destination more quickly. To take advantage of this concept, DEFS uses the main direction and the direction entropy of a node. By the main direction of a node, we mean the direction that the nodes move with the highest frequency among the four cardinal directions. In DEFS, the direction entropy captures the certainty of the direction of a node. The lower the direction entropy a node has, the greater the certainty of direction it has; that is, the likelihood that the node moves in a certain direction. Since each node sends a message to another node with a differ-

ent location and less direction entropy, DEFS utilizes those nodes that are more likely to travel to various locations to forward the messages to the destination node. For these reasons, DEFS selects the proper number of relay nodes during forwarding and, thus, it balances the network traffic and transmission delay quite well.

This paper is organized as follows. We first discuss related work in Section II. Section III introduces the proposed scheme in more detail. The simulation environment and experimental results are provided in Section IV. Finally, our conclusions are drawn and future work is discussed in Section V.

## II. RELATED WORK

Opportunistic routing schemes can be classified into two categories: social-unaware and social-aware ones. Social-unaware schemes do not use the social information at all, whereas social-aware ones take advantage of the information about the nodes' behaviors or social relations to make decisions about forwarding messages.

Routing schemes, such as Epidemic and Spray-and-Wait [7], can be considered as social-unaware routing schemes, as they do not need to any specific social information for routing. In Epidemic, a node duplicates the message to others whenever the node encounters other nodes. In Spray-and-Wait, a node sprays a fixed number  $k$  of duplicates of the message into some encountered nodes and then waits until these nodes meet the destination.

Various social-aware routing schemes have been proposed in OPPNETs [4, 6, 8-10]. SimBet [8] uses the locally determined social similarity and the betweenness centrality to the destination. When a node meets other nodes, it sends the message to the node that has a higher value computed with the similarity and betweenness centrality. Bubble rap [9] makes use of both the global and local centralities. Bubble rap sends a message to the destination or its community. However, when the destination belongs only to a community whose members have a low global centrality value, this scheme may fail to deliver the message. In this case, a relay node cannot be identified as the destination in the same local community. PROPHET [6] estimates a probabilistic metric called the delivery predictability. When the nodes are near enough to the destination, they exchange the vector of the information on the messages and the delivery predictability vector. The information in the vector is used to determine which message should be sent. PeopleRank [10] uses the PageRank algorithm of Google for forwarding decisions. When the adjacent nodes meet in the social network, they exchange their current values of PeopleRank and the numbers of neighbors in the social network. In [4], an analytical model based on Markov processes for social-aware routing in OPPNETs was proposed.

### III. PROPOSED SCHEME

#### A. System Model

The system used for the proposed scheme obeys the rules of typical data forwarding. Each node in a network has a unique identifier. The nodes in a network are denoted by  $N = \{N_1, N_2, N_3, \dots, N_m\}$ , where  $m$  is the number of nodes in the network. Every node periodically measures its own direction. To determine the direction, each mobile device is assumed to have a positioning system. A node determines the direction according to its position changes. Such processing incurs some additional computational overhead. However, this paper does not focus on the reduction of the computational cost. Note that this can be achieved with simple equations.

#### B. Estimate Information of Direction

A source node is not aware of the current position of the destination node. Nodes move around with various directions. DEFS utilizes the information about the directions of the nodes. Each node accumulates its moving direction changes within a fixed angle for the four cardinal directions: north, south, west, and east. A fixed angle is a right angle, as illustrated in Fig. 2.

Each node determines the highest frequency of its moving direction changes for its main direction and also computes the entropy of the movement directions for certainty.

The direction entropy is based on the information entropy of information theory, in which the entropy is a measure of the uncertainty in a random variable [11]. The direction entropy is a numerical value, such that the smaller the value is, the more reliable it is; that is, a large value indicates less certainty.

The main direction of a node  $N_i$  is denoted as  $\delta_i$  and  $E_i$  is the value of the direction entropy of  $N_i$ .  $\sigma_i$  is the total number of direction changes made by  $N_i$ . In this context, the direction entropy  $E_i$  of a node  $N_i$  is a numerical value

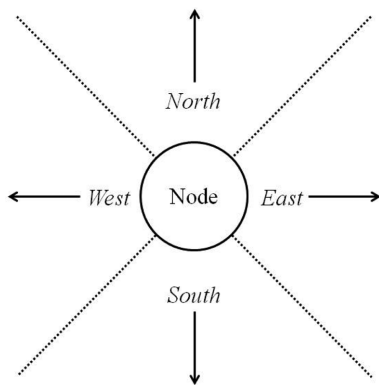


Fig. 2. The four moving directions of a node.

about the certainty of direction, which is defined as follows.

$$E_i = -\sum_{d=1}^4 p\left(\frac{x_d^i}{\sigma_i}\right) \log_4 p\left(\frac{x_d^i}{\sigma_i}\right)$$

In the equation above,  $p(x_d^i)$  is the probability mass function of  $N_i$ 's outcome (frequency)  $x_d^i$ , indicating one of the four cardinal directions, where  $d = 1$  (north), 2 (south), 3 (east), or 4 (west). The base of the logarithm is 4 in the equation, due to the entropy value normalization. Thus, the entropy value is between 0 and 1.0.

Fig. 3 shows an example of the direction information. The accumulated occurrences in each direction are recorded on the table. The direction of the maximum accumulated value is denoted as  $\delta$ ; thus  $\delta = east$  in the example. The values of  $\sigma$  and  $E$  are calculated based on the accumulated values. In the example of Fig. 3,  $\sigma = 97$ ; that is, the sum of accumulated values,  $21 + 19 + 33 + 24 = 97$ .

$E$  in the example is computed by applying the above equation as follows:

$$\begin{aligned} x_1 &= 21/97 = 0.2165, x_2 = 19/97 = 0.1959 \\ x_3 &= 33/97 = 0.3402, x_4 = 24/97 = 0.2474 \\ \log_4 x_1 &= -1.1038, \log_4 x_2 = -1.1760 \\ \log_4 x_3 &= -0.7778, \log_4 x_4 = -1.0074 \\ E &= -(x_1 \log_4 x_1 + x_2 \log_4 x_2 + x_3 \log_4 x_3 + x_4 \log_4 x_4) = 0.9831 \end{aligned}$$

Each node calculates its  $\sigma$  and  $E$  as explained in the example.

#### C. Forwarding Process

In DEFS, a node  $N_i$  waits and carries its message until it meets the destination node or sends the message to an encountered node  $N_j$  with  $\delta_j$  being different from  $\delta_i$  and  $E_j$  being less than a threshold  $\tau$ . The forwarding process is terminated when all the messages are delivered to their respective destinations. We determined the threshold  $\tau$  with extensive experiments, as described in the next section.

A message  $M$  consists of the following components:

$$M_i = \langle Data_i, S_i, D_i, t \rangle,$$

where  $Data_i$  is the message content,  $S_i$  is the ID of the source that generated the data,  $D_i$  is the ID of the destination node, and  $t$  is the time at which  $M_i$  is generated.

Fig. 4 shows an example of the forwarding decisions made in DEFS. We assume that the threshold  $\tau$  is 0.97 for

Direction information table			
North	South	East	West
21	19	33	24
$E$		$\delta$	
0.9831		east	

Fig. 3. Example of direction information.

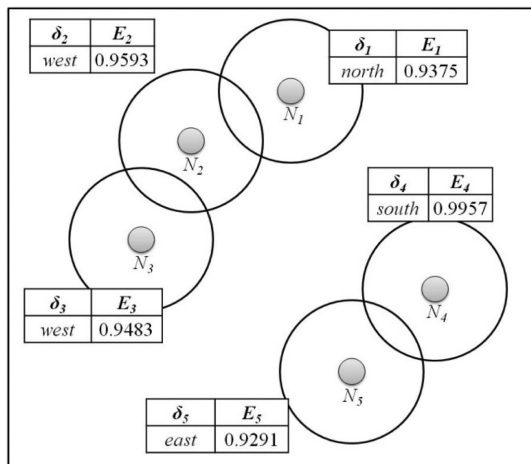


Fig. 4. Message forwarding for direction entropy-based forwarding scheme (DEFS).

the example. When  $N_2$  meets  $N_1$  and  $N_3$ , they exchange their direction information and direction entropies. Then  $N_2$  finds that  $E_1 < 0.97$  and  $\delta_1 = \text{north}$ , which different from  $\delta_2 = \text{west}$ . Hence  $N_2$  forwards the message to  $N_1$ . However,  $N_2$  does not send the message to  $N_3$  since  $\delta_2 = \delta_3$ . In the meantime,  $N_4$  does not transmit the message to  $N_5$ , since  $E_5$  is  $0.9957 > 0.97$ . However, since the direction entropy of  $N_4$  is the less than the threshold and it has a different main direction from  $\delta_5$ ,  $N_5$  transmits the message to  $N_4$ , if anywhere.

#### IV. EXPERIMENTAL RESULTS

##### A. Simulation Environment

In this paper, we use the network simulator NS-2 v.2.35 [12] for the simulations. The number of nodes is set to 40. The network area is set to  $450 \text{ m} \times 450 \text{ m}$  and the communication range is 10 m. The movement of a node follows the HCMM [13] which is a frequently used moving pattern in OPPNET simulations. The velocity of a node ranges from 1 to 9 m/s, which is appropriate for pedestrians and vehicles. In our simulator, a mobile node issues a single message. The total simulation time is 9,600 seconds. The warm-up time for estimating the direction entropies is 600 seconds. After transmitting a message to other nodes, the originating node does not delete it. In addition, for the sake of simplicity, we do not consider the buffer size, bandwidth, or power. We run each scheme 20 times to calculate the average of the results. Table 1 shows the parameters used in our simulation. In this experiment, we evaluate three metrics: the delivery success ratio, the network traffic, and the delay time. The delivery success ratio is the ratio of the number of messages reaching the destination to the total number

Table 1. The parameters of the simulation

Parameter	Value
Number of nodes	40
Communication range	10 m
Network area	$450 \text{ m} \times 450 \text{ m}$
Node velocity	1–9 m/s
Warm-up period	600 sec
Simulation time	9,600 sec

of messages generated. The network traffic is the number of packets sent and received between nodes. The delay time is the time period from the moment that a message is generated to the time at which it is delivered to the destination. Our simulation followed the hotspot [14].

##### B. Effect of Threshold $\tau$

We examine the performance of DEFS at different values of the threshold,  $\tau$ . Fig. 5 shows the results of the experiments in terms of the delivery ratio, the network traffic and the delay time with various values of  $\tau$ . The threshold  $\tau$  may vary up to 1.0 from 0.90.

Fig. 5(a) shows the variation of the delivery ratio as the simulation time increases. The delivery ratio increases with increasing threshold  $\tau$ .

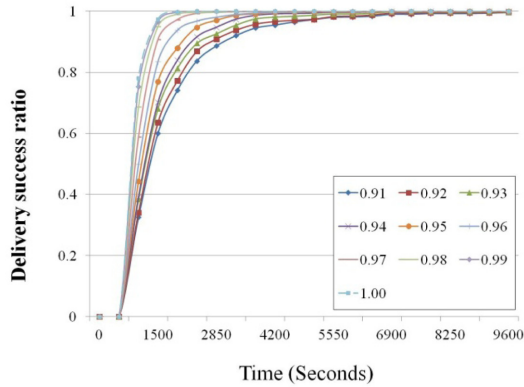
As shown in Fig. 5(b), the y-axis is on the log-scale and the x-axis represents the threshold  $\tau$  varying from 0.91 to 1.00 in increments of 0.01. When the threshold  $\tau$  is small, DEFS exhibits low traffic, while when it is large, DEFS show high traffic. Such a phenomenon arises since, in DEFS, a node forwards the message to other nodes with direction entropies less than  $\tau$  and different main directions.

On the other hand, in Fig. 5(c), as the threshold  $\tau$  increases, DEFS shows a lower delay, since in DEFS nodes have a better chance of forwarding the messages to more nodes; hence, the forwarding process is done more quickly. However, when the threshold  $\tau$  is large, DEFS has similar performance to the flooding-based scheme, since there are more and more nodes that participate in forwarding process.

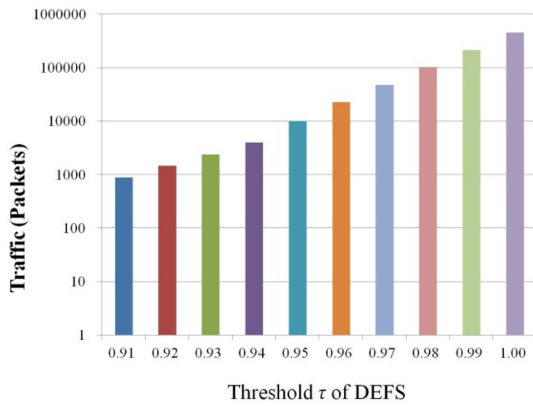
Note that when  $\tau = 1.0$ , all the nodes in the network have a chance of being involved in forwarding, as long as they have different main directions, because the maximum value of the direction entropy is 1.0.

However, since DEFS allows forwarding to the nodes with direction entropies of less than  $\tau$  and different main directions, its traffic amount is still smaller than those of the flooding-based schemes.

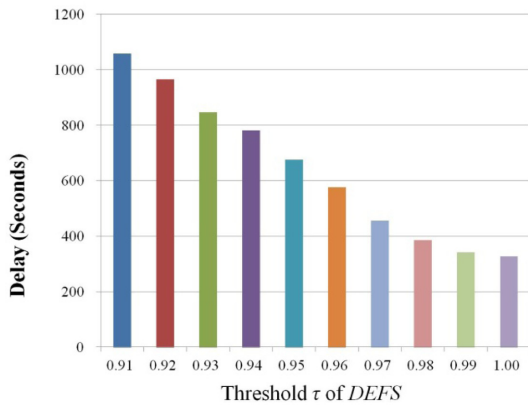
From the results in Fig. 5, we determined that the value of the threshold  $\tau = 0.97$  for our simulation, because when  $\tau = 0.97$  the delivery ratio approaches 1.0 after



(a)



(b)



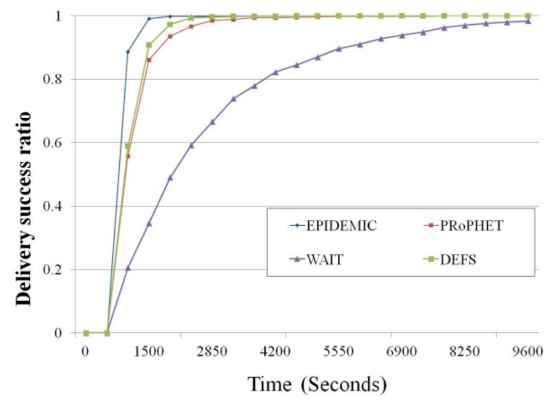
(c)

**Fig. 5.** Simulation results for threshold  $\tau$  of DEFS. (a) Delivery success ratio, (b) traffic, and (c) delay.

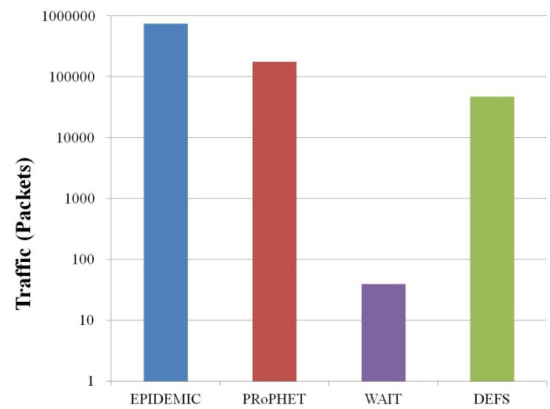
2,800 seconds and a good balance is achieved between the network traffic and the delay time.

### C. Comparison between DEFS and Other Schemes

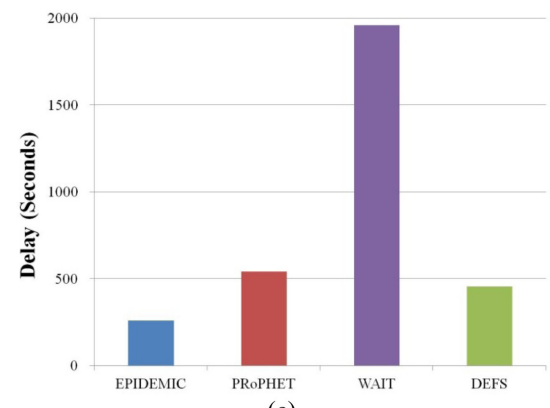
First, we evaluate the performance based on the simulation time for each of the three schemes: Epidemic,



(a)



(b)



(c)

**Fig. 6.** Simulation results for DEFS and other schemes. (a) Delivery success ratio, (b) traffic, and (c) delay.

PROPHET, and WAIT. We set the threshold  $\tau$  to 0.97 for DEFS. Fig. 6(a) shows the delivery ratios of the schemes as the simulation time increases. The results are shown after 600 seconds due to the warm-up period. The flooding-based schemes achieve the maximum delivery ratio faster, due to the use of multiple copies of the messages. WAIT does not distribute the message and waits for the destination instead.

Therefore, WAIT requires a longer time to reach a delivery ratio of 1.0. In DEFS, the nodes basically forward the packets by flooding. Hence, DEFS is faster than the other schemes except for Epidemic.

Fig. 6(b) shows the results of the network traffic for the various schemes. Note that the y-axis of the figure is on the log-scale. DEFS exhibits much lower traffic than Epidemic and PROPHET with 93.7% and 74.4% improvements, respectively. DEFS shows higher traffic than WAIT, because the nodes in WAIT forward the packets only when they meet the destination nodes.

The reason why DEFS shows such a reduction in traffic is that the relay nodes are suitably selected. Observe that those nodes with the same  $\delta$  tend to head in the same direction and those that move in the four directions evenly are highly likely to crisscross with other nodes. Thus, DEFS selects a relay node that has a different  $\delta$  from that of the neighboring node with the message, as well as a low entropy value.

Note that the traffic of WAIT is equivalent to the number of messages generated for the simulation. Fig. 6(c) compares the delay times of the schemes, showing almost a reverse trend compared to the network traffic results in Fig. 6(b). The average delay time of DEFS is about 198 seconds longer than that of Epidemic, while it is 84 seconds shorter than that of PROPHET. In DEFS, relaying nodes with lower entropies move all around the network, resulting in faster delivery of the message, although a source node doesn't know where the destination node is. The results of the experiments show that DEFS outperforms PROPHET in terms of both the network traffic and delay.

## V. CONCLUSIONS

In OPPNETs, the use of flooding-based schemes results in higher network traffic and lower transmission delays. On the other hand, the wait-based schemes suffer from unbearable transmission delays. In order to resolve this problem of imbalance between the traffic and delay, we proposed an effective forwarding scheme using the direction entropy and main direction, which capture the uncertainty in people's mobility. As a future work, we plan to study more enhanced predictable forwarding schemes with the coordinates of the nodes in OPPNETs.

## ACKNOWLEDGMENTS

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2013R1A1A2011114).

## REFERENCES

1. P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, "Pocket switched networks and human mobility in conference environments," in *Proceedings of ACM SIGCOMM Workshop on Delay-Tolerant Networking*, Philadelphia, PA, 2005, pp. 244-251.
2. Z. Zhang, "Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challengers," *IEEE Communication Surveys Tutorials*, vol. 8, no. 1, pp. 24-37, 2006.
3. M. Conti and M. Kumar, "Opportunities in opportunistic computing," *IEEE Computer*, vol. 43, no. 1, pp. 42-50, 2010.
4. C. Bordini, M. Conti, and A. Passarella, "Modeling social-aware forwarding in opportunistic networks," *Performance Evaluation of Computer and Communication Systems, Lecture Notes in Computer Science vol. 6821*, Heidelberg: Springer, pp. 141-152, 2011.
5. A. Vahdat and D. Becker, "Epidemic routing for partially-connected ad hoc networks," Duke University, *Technical Report CS-2000-06*, 2000.
6. T. Spyropoulos, K. Psounis, and C. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking*, Philadelphia, PA, 2005, pp. 252-259.
7. E. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant MANETs," in *Proceedings of the 8th ACM international symposium on Mobile Ad Hoc Networking and Computing*, Montreal, Canada, 2007, pp. 32-40.
8. P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: social-based forwarding in delay tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1576-1589, 2011.
9. A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," in *Service Assurance with Partial and Intermittent Resources, Lecture Notes in Computer Science vol. 3126*, Heidelberg: Springer, pp. 239-254, 2004.
10. A. Mtibaa, M. May, C. Diot, and M. Ammar, "PeopleRank: social opportunistic forwarding," in *Proceedings of IEEE INFOCOM*, San Diego, CA, 2010, pp. 1-5.
11. S. Ihara, *Information Theory for Continuous Systems*, Singapore: World Scientific Publishing, 1993.
12. The project of Network Simulator ns-2, <http://www.isi.edu/nsnam/ns/>.
13. C. Boldrini and A. Passarella, "HCMM: modelling spatial and temporal properties of human mobility driven by users' social relationships," *Computer Communications*, vol. 33, no. 9, pp. 1056-1074, 2010.
14. S. K. Kim, J. H. Choi, and S. B. Yang, "Hotspot: location-based forwarding scheme in an opportunistic network," *Ad Hoc & Sensor Wireless Networks*, in press.



---

**MinSeok Jeon**

---

MinSeok Jeon is currently an M.S. candidate in computer science at Yonsei University in Korea. His research interests are in opportunistic network, mobile social network, and distribute system.



---

**Sun-Kyum Kim**

---

Sun-Kyum Kim received his M.S. degree in computer science at Yonsei University in Korea in 2012. He has been a Ph.D. candidate at Yonsei University since 2012. His research interests are in opportunistic networks, mobile social networks, and mobile computing.



---

**Ji-Hyeun Yoon**

---

Ji-Hyeun Yoon is currently a Ph.D. candidate in computer science at Yonsei University in Korea. His research interests include opportunistic network, mobile social network, and mobility analysis.



---

**JunYeop Lee**

---

JunYeop Lee is currently a Ph.D. candidate in computer science at Yonsei University in Korea. His research interests include opportunistic network, social network, and machine learning.



---

**Sung-Bong Yang**

---

Sung-Bong Yang received his M.S. and Ph.D. degrees from the Department of Computer Science at the University of Oklahoma in 1986 and 1992, respectively. He has been a professor at Yonsei University since 1994. His research interests are in graph algorithms, mobile computing, and social network services.