Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2015, Article ID 971704, 10 pages
http://dx.doi.org/10.1155/2015/971704

Research Article

SAFM: An Adaptive Socially Aware Feedback Mechanism in Delay Tolerant Sensor Networks

Yan Liu,[1] Kun Wang,[2] Guo Huang,[2] Jin Qi,[2] and Yanfei Sun[2]

[1] *College of Electrical Engineering and Control Science, Nanjing Tech University, Nanjing 210009, China*
[2] *Key Lab of Broadband Wireless Communication and Sensor Network Technology,*
Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Correspondence should be addressed to Kun Wang; kwang@njupt.edu.cn

Received 18 June 2015; Accepted 16 August 2015

Academic Editor: Jianping He

In Delay Tolerant Sensor Networks (DTSN), it is a challenge to ensure the delivery reliability, due to the lack of end-to-end path, intermittent connection, and high latency. In addition, the resource utilization is decreased because of the limited resources and a large number of redundant copies. To this end, this paper proposes an adaptive Socially Aware Feedback Mechanism (SAFM). In this mechanism, the historical information of the encountered nodes is utilized to construct social links which indicates the level of social relationship between nodes. In the feedback process, acknowledgements are forwarded to the nodes whose social link (SL) is higher than a given threshold α. After getting the acknowledgements, nodes will delete the copies of messages which have been received by the destination nodes, so as to reduce the redundancy. This mechanism is to reach a tradeoff between overhead and delivery efficiency. In simulation, the threshold α is obtained to reach the best performance of SAFM. Compared with active and passive receipt approaches in an acceptable range of delay, SAFM improves the delivery probability, decreases the buffer occupancy, and reduces the overhead.

1. Introduction

Delay Tolerant Sensor Network (DTSN) [1] was initially proposed by Kevin Fall who expanded the research of the Interplanetary Network. DTSN is a kind of challenged network supporting interoperation among various local area networks, and long delays can be tolerated. In DTSN, reliable delivery is guaranteed by the custody transfer service [2]. Besides, message forwarding usually follows Store-Carry-Forward scheme to deal with some issues in traditional networks such as intermittent connection, asymmetric data rate, and high error rate.

Social awareness is defined as a sociological concept which describes various social phenomena and human sociability. But in computer science, the main ideas of social awareness are the awareness and response of the computer system to the social context [3]. Human interaction will generate large amount of history information, and the social awareness can be got through analysis of these history data to reflect the social features of human interaction. DTSN with social awareness refers to the social applications of DTSN. In this kind of applications, the mobile devices carried by people are indicated as nodes. So the nodes' movement possesses social features. These social features can reflect the potential social relationships, the social statuses, and the forwarding ability of nodes. With the proposal of socially aware computing [3], it is an important technology where the social characters are utilized to solve the issues in DTSN.

Since there is no end-to-end path between the source and destination, and message transmission follows opportunistic forwarding manner, it is a challenge to provide reliable delivery in DTSN [4, 5], as is illustrated in Figure 1. One solution to this reliability problem is that every node keeps the copy of message. However, when the message is received by the destination node, it is probable that a large number of message copies still exist. These redundant copies occupy the limited buffer space so that the resource utilization reduces and network performance degrades. In order to provide reliable transmission and to reduce redundancy, an efficient and secure feedback mechanism is required. Acknowledgments

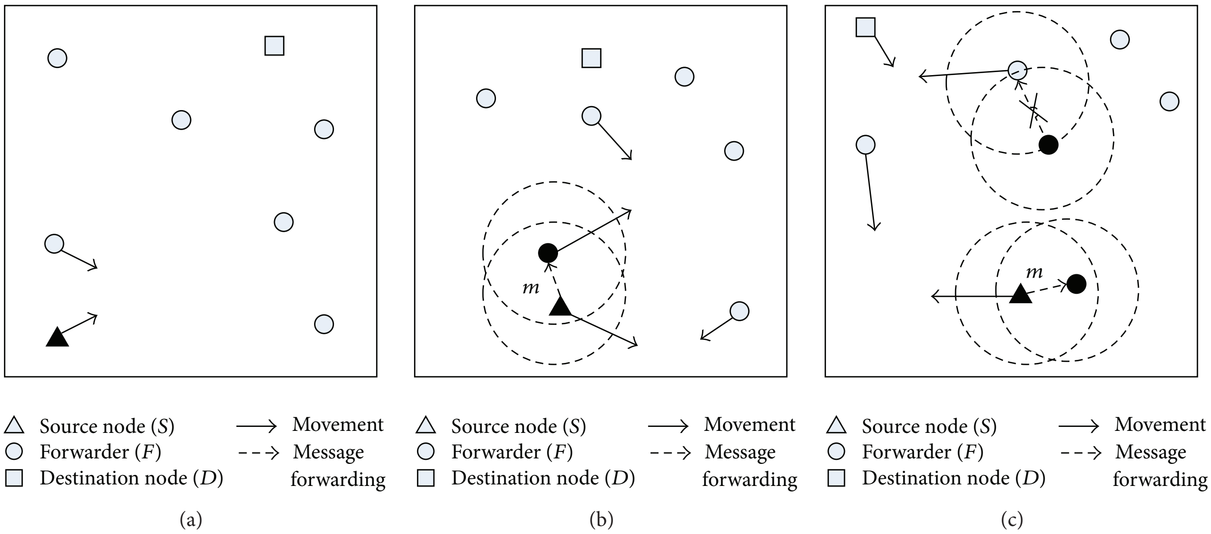


Figure 1: Problem statement (message forwarding follows an opportunistic manner shown in (a) and (b). Black nodes carry messages and white nodes are those without messages. However, if an error happens in opportunistic forwarding, shown in (c), the transmission from source to destination fails. As for DTSN, it is a challenging task to provide reliable delivery).

are sent to the network to delete redundant copies. However, traditional feedback mechanisms based on TCP/IP protocol are inapplicable in DTSN due to the lack of an end-to-end path between nodes. Therefore, it is a challenging task to design an efficient feedback mechanism to meet the demand of the reliable delivery and the improvement of resource utilization for DTSN.

To this end, the major contributions of our work are summarized as follows:

(i) This paper defines a new concept of social link to indicate the social relationship between nodes pair according to the encountering history, reflecting the social features of nodes' interaction in DTSN.

(ii) This paper designs an adaptive Socially Aware Feedback Mechanism (SAFM) to reduce redundancy with the utilization of social links of nodes, reaching a tradeoff between overhead and delivery efficiency.

The remainder of this paper is organized as follows. Section 2 shows the related work. Section 3 describes the proposed adaptive Socially Aware Feedback Mechanism (SAFM) in detail, and the complexity analysis of this mechanism is also presented. Performance evaluation of SAFM is given in Section 4. Finally, Section 5 concludes the paper.

2. Related Work

To deal with the problem of delivery reliability in DTSN, the available researches can be classified into two categories: hop-by-hop transmission reliability and end-to-end reliability [4].

(1) Hop-by-Hop Transmission Reliability. Hop-by-hop method is the classic transmission method in DTSN [4]. The main idea is that message is forwarded to the destination node along the path in an area and every area is a hop. The gateway at the edge of area, denoted as custodian, provides the reliable custody transfer between different areas. Therefore, there is no end-to-end feedback mechanism in this method. The source node only wonders whether the next gateway has received the message and assumes that the gateway provides custody transfer to the message. The researches on the reliability for hop-by-hop transmission focus on the next node selection and routing strategies, which can be divided into two kinds:

(A) Multicopy based routing, such as Epidemic [6], Spray and Wait [7], Spray and Focus [8], and SLR [9]. The copies of the messages are exchanged between the meeting nodes so that the messages can be sprayed into the network quickly. Therefore, the multicopy based routing provides the shortest transmission delay but the most redundancy and overhead.

(B) Single-copy based routing, such as PROPHET [10], MaxProp [11], and EMD [12]. The forwarding mechanisms based on probability prediction are adopted in these routing algorithms. A network is constructed with a random model and the probabilities of meeting the destination node are predicted to select the proper forwarding nodes.

(2) End-to-End Reliability. In order to provide end-to-end reliability, some feedback mechanisms are adopted to send acknowledgements in network, informing other nodes to delete the redundant copies. In this way, the overhead can be controlled and the resource utilization can be improved. Takahashi et al. [13] proposed a convergent message reliable transmission mechanism with controlled copies. The maximum number of copies depends on the locations of source node and destination node. Bulut et al. [14] adopted the erased coding technology in DTSN to reduce network

overhead and improve the resource utilization. Dias et al. [15] investigated the efficiency and tradeoffs of several scheduling and dropping policies enforced in a Spray and Wait routing scheme. Soares et al. [16] studied the impact of different cooperation levels on the performance of Vehicular Delay Tolerant Networks. Xiao et al. [17] proposed a utility-aware data transmission scheme which considers both internal property and external contact of nodes.

In terms of feedback mechanism in DTSN, there are only a few researches currently. Harras and Almeroth [4] introduced two available feedback mechanisms: active and passive receipt:

(A) *Active Receipt.* After receiving the message, destination node generates a corresponding acknowledgment of the message. Then, the acknowledgment is actively forwarded to the encountered nodes in the flooding way. Every encountered node receives the acknowledgment and then deletes the copies of the received message. However, active receipt may bring about great overhead. Abundant acknowledgments are generated in active receipt, which occupies the limited resources and increases the overhead dramatically.

(B) *Passive Receipt.* In passive receipt, the implicit acknowledgment of the received message is generated. Different from active receipt, the acknowledgment will not be sent to the encountered node until the encountered node tries to send the copy of received message. In other words, the passive receipt mechanism only prevents the retransmission of the received message, rather than deleting redundant copies actively. Therefore, the overhead caused by acknowledgment is low. But it is also inefficient in decreasing redundancy.

Besides, Ali et al. [18] designed a global selective strategy based feedback mechanism (G-SACK), where destination node receives the message and records the global information at the same time to ensure reliable transmission. Seo and Lee [19] proposed that destination node can select an optimal path according to the location information in message. Then, the acknowledgment is sent along the optimal path to source node. An et al. [20] proposed an adaptive feedback mechanism based on the congestion level of the network, so as to reduce the overhead and latency. However, the social features of the nodes movement are not concerned in all the works above.

To deal with the problems of great overhead in active receipt and the inefficiency in passive receipt, this paper proposed an adaptive Socially Aware Feedback Mechanism (SAFM) for delivery reliability in DTSN.

3. Social Awareness Based Feedback Mechanism

3.1. Network Model. Figure 2 illustrates the network model. The social DTSN is represented by the graph $G = (V, E)$, where $V = \{v_1, v_2, \ldots, v_n\}$ is the set of vertexes in the network and $E = \{(v_i, v_j) \mid d(v_i, v_j) \le d_{\mathrm{MAX}}, v_i, v_j \in V\}$ is the set of edges. The following assumptions are used in this paper:

(1) The node possesses four properties: P_{MAX} is the maximum transmission power, $P(i)$ is the transmission power of node v_i, $N(i)$ stands for the neighbor node of v_i, and $\mathrm{SLT}(i)$ denotes the social link table of v_i. i is denoted by v_i for convenience.

(2) The edge possesses three properties: d_{MAX} is the maximum distance for communication, $d(v_i, v_j)$ is the distance of nodes i and j, and $\mathrm{SL}(i, j)$ is the social link between nodes i and j.

(3) Messages are forwarded by the nodes in an opportunistic manner. In other words, the message transmission takes place only when two nodes are encountering each other and are within the range of communication at the same time, or $P_{(i)} \le P_{\mathrm{MAX}}$.

(4) There is a social link between two nodes if they contact with high frequency, for long time and regularity, or there is an edge connecting the two nodes. Therefore, the social links $\mathrm{SL}(i, j)$ between nodes can be quantified based on their encountering history.

(5) The stronger the social link is, the more the probability of successful delivery will be and the more possible the taking of the same copies of message will be.

(6) The location information of the nodes is not considered.

3.2. Social Link. In this paper, the social characteristics of the nodes' movement are considered to define the social pressure metric (SPM) [21], which indicates the relationship of the nodes. Furthermore, the social link (SL) of the node pair is also defined.

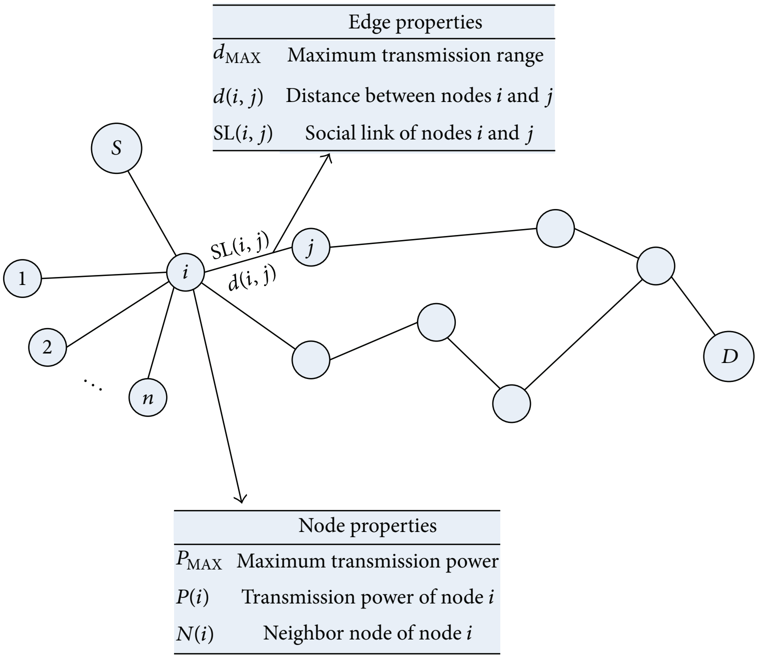
Definition 1. Node S is the source node sending messages, node D is the destination node, and nodes i, j, and k are the meeting nodes.

Definition 2. T is a period of time, t is the current time, and t_{inter} is the intermeeting time. If nodes are contacting with each other at time t, then $t_{\mathrm{inter}} = 0$.

Definition 3. With time unit tending to 0, the average forwarding delay from node i to node j in the time period of T is the social pressure metric $\mathrm{SPM}_{i,j}$, which is defined as follows:

$$\mathrm{SPM}_{i,j} = \frac{\int_{t=0}^{T} t_{\mathrm{inter}} dt}{T} = \frac{\sum_{x=1}^{n} t_{\mathrm{inter},x}^{2}}{2T}, \quad (1)$$

where t_{inter} is the intermeeting time and n is the meeting times of node i and node j in the period of time T. SPM indicates the transmitting time of packets between nodes. In Performance Evaluation, the starting and ending time of each encounter are recorded. So $t_{\mathrm{inter},x}$ is the difference between the starting time of the xth meeting and the ending time of the last meeting.



Edge properties	
d_{MAX}	Maximum transmission range
$d(i, j)$	Distance between nodes i and j
$\mathrm{SL}(i, j)$	Social link of nodes i and j

Node properties	
P_{MAX}	Maximum transmission power
$P(i)$	Transmission power of node i
$N(i)$	Neighbor node of node i

Figure 2: Network model.

Table 1: Social link table.

Neighbor node ID	Social link $\mathrm{SL}_{(i,j)}$	Intermeeting time t_{inter}
8 bits	16 bits	16 bits

Definition 4. The social link $\mathrm{SL}(i, j)$ of nodes i and j is the inverse of the value of $\mathrm{SPM}_{i,j}$; that is,

$$\mathrm{SL}\left(i, j\right) = \frac{1}{\mathrm{SPM}_{i,j}} = \frac{2T}{\sum_{x=1}^{n} t_{\mathrm{inter},x}^{2}}. \tag{2}$$

The social link provides the friendship relationship of each node pair.

3.3. Algorithm Description. In order to reduce redundancy and decrease buffer occupancy, we proposed an adaptive Socially Aware Feedback Mechanism (SAFM) for DTSN. Epidemic routing is adopted in the forwarding process. Messages are replicated to nodes when encounter occurs. There are a lot of message copies in Epidemic routing due to its flooding strategy. Therefore, we can use feedback mechanism to reduce redundant copies and improve resource utility rate and reliability. Each node locally maintains a social link table (SLT) to record the social links with all the neighbor nodes encountered before. The structure of social link table SLT(i) of node i is shown in Table 1.

In order to eliminate redundancy and reduce buffer occupation, we propose a feedback mechanism. After receiving the message, the destination node generates an acknowledgement (ACK) of the received message and then forwards this ACK to other nodes, informing them to delete the copies of the message. When the destination node is encountering a node, it calculates the encountered node social link value. If this value is larger than the threshold (here it is set as α), it indicates that they have strong social relation and more meeting opportunities. It is highly possible that the encountered node is carrying a copy of received message. So destination node will send ACK to inform the node to delete the message copy. The selection of threshold value will be presented in Performance Evaluation. Our social link aware feedback mechanism can reduce the high overhead caused by sending large amount of ACK and offer high efficiency.

```
Inputs: Received message $m$, Acknowledgement ACK($m$),
        Node $i$ carrying acknowledgement, Encounter node $j$
Outputs: Message queue in node $j$ $M_1, M_2, \ldots, M_n$
Check social link table SLT($i$) and get SL($i$, $j$)
if SL($i$, $j$) > $\alpha$
    $i$ send ACK($m$) to $j$
else if $j$.sendRequest == $m$
    $i$ reject $m$
    $i$ send ACK($m$) to $j$
for each message $M_j$ do
    if $M_j$ == $m$
    $j$ delete $M_j$
    end if
end for
```

Pseudocode 1: Pseudocode of SAFM.

The pseudocode of SAFM is shown in Pseudocode 1.

After receiving the message m, destination node D generates an acknowledgment ACK(m). When the node i carrying ACK(m) encounters node j, the following steps are executed.

Step 1. Node i searches its social link table $\mathrm{SLT}(i)$ to get the social link value $\mathrm{SL}(i, j)$ of node j.

Step 2. Compare $\mathrm{SL}(i, j)$ with α; if $\mathrm{SL}(i, j) > \alpha$, send $\mathrm{ACK}(m)$ to node j, and then turn to Step 5; otherwise, turn to Step 3.

Step 3. If node i receives the sending request of message m from node j, it will reject the sending request.

Step 4. Send $\mathrm{ACK}(m)$ to j.

Step 5. After receiving the acknowledgement $\mathrm{ACK}(m)$, node j traverses all the messages in buffer space and deletes the copies of message m.

3.4. Algorithm Analysis

3.4.1. Time Complexity

(1) Sending Node. After receiving the message, destination node generates the acknowledgment $\mathrm{ACK}(m)$. In feedback process, when node i carrying $\mathrm{ACK}(m)$ encounters node j, the sending node i needs to calculate node j social link $\mathrm{SL}(i, j)$, and this time complexity is $O(N)$, where N is the number of neighbor nodes in the social link table of node i. This is because node i needs to search for its social link with node j in the social link table $\mathrm{SLT}(i)$. The number of neighbor nodes in the table is N, so the time complexity is $O(N)$.

(2) Receiving Node. If the social link is higher than a threshold, receiving node j will obtain the acknowledgment $\mathrm{ACK}(m)$ and delete the copies of m. Node j needs to traverse the message queue to find the same copies of m, and then it drops these redundant copies. If the length of message queue is M, the time complexity of traversal is $O(M)$. Therefore, the time complexity of node j to delete copies is $O(M)$.

3.4.2. Updating Overhead. The updating of social link is made in each encounter. Each node locally maintains its social link table to record the historical information of the social links with neighbor nodes. When encountering node j, node i will traverse its social link table to search for the historical social link with node j and update the information. As a result, the updating overhead of social link is $O(N)$, where N is the number of neighbor nodes.

4. Performance Evaluation

This section gives the simulation of this algorithm using ONE [22]. The simulation environment is set as follows. The Helsinki Map [17] is used as the mobile area. There are 150 mobile nodes deployed in $4500\,\mathrm{m} * 3400\,\mathrm{m}$ area. They are divided into three categories to represent walker, runner, and rider. There are 50 nodes for each category and the radio communication range of each node is set to be 10 m. The channel bandwidth is 250 kbps. The nodes buffer space is changing from 1 M to 10 M. The mobility model is the default Helsinki Map Model. The mobile speed of nodes is set as 1.34 m/s, 4 m/s, and 8.94 m/s separately to indicate the three average mobile speeds of walking, running, and riding [23]. The message size is 50 KB, and generating time interval is set as 10 s, 20 s, 30 s, 40 s, 50 s, 60 s, 70 s, 80 s, 90 s, and 100 s. The simulation running time is set to be 12 H.

In the simulation, four metrics are used to verify the performance.

(1) Buffer Occupancy. It is the ratio of the used buffer size to the total buffer size. Buffer occupancy $= \sum_{i=1}^{N}(\mathrm{occupiedSize}_{n_i}/\mathrm{bufferSize}_{n_i})/N$.

(2) Delivery Probability. It is the ratio of the delivered messages to the total number of messages. Delivery probability = Delivered/created.

(3) Overhead Ratio. It is the ratio of the forwarded but unsuccessfully delivered messages to all delivered messages. Overhead ratio = (Relayed − delivered)/delivered.

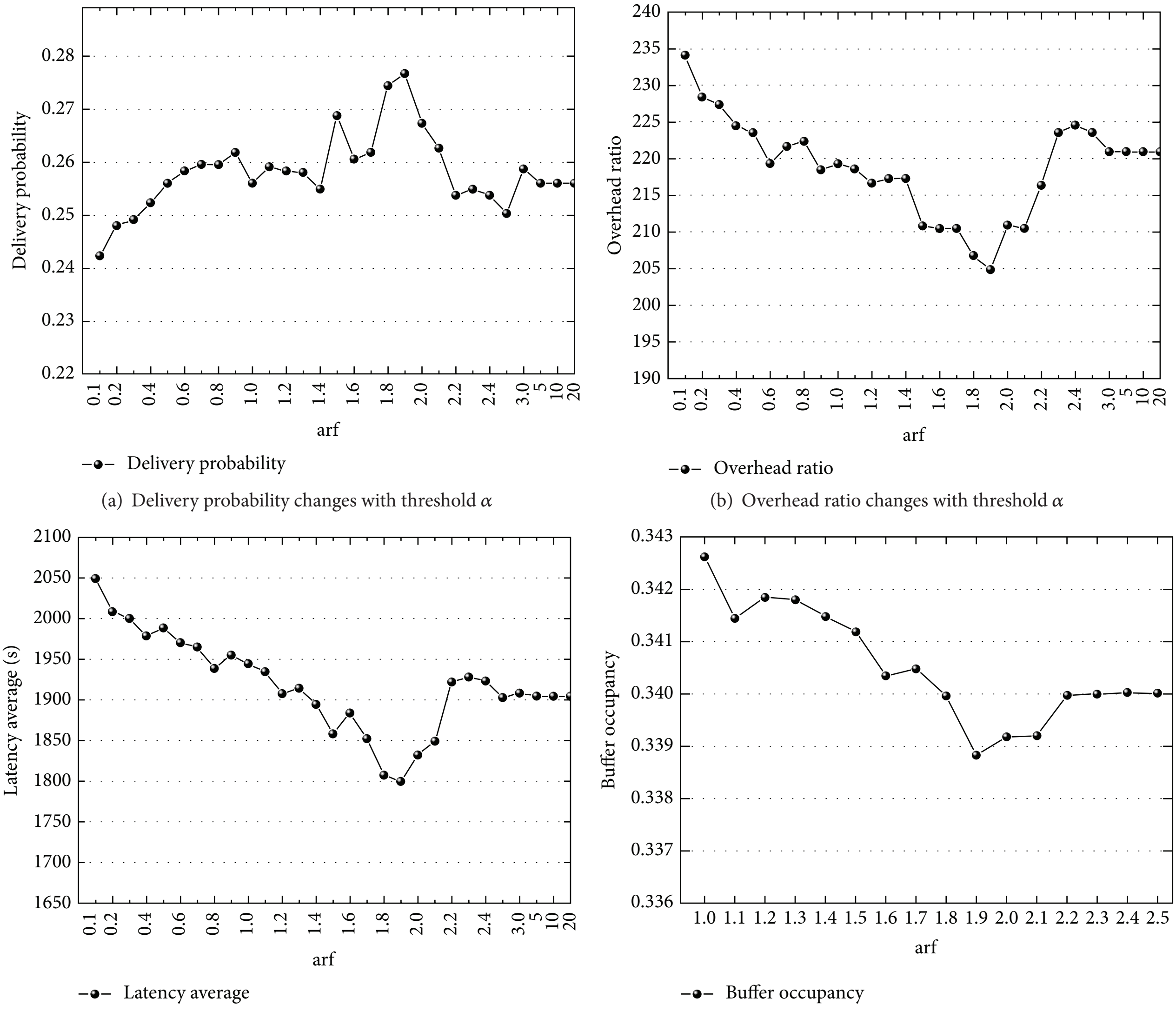
(4) Latency Average. It is the average latency of all the delivered messages. Latency average $= \sum_{i=0}^{n-1}$(receiving time of message i − sending time of message i)/number of messages $= \sum_{i=0}^{n-1}(t_{ir} - t_{is})/n$.

4.1. Selection of Threshold α. In our proposed SAFM, social link is compared with the threshold α to decide if the acknowledgment should be sent. If the value of α is too high, the sending speed of acknowledgment is too slow to reduce redundancy effectively. In other parts, if the value of α is too low, a large number of acknowledgments will occupy the buffer space to decrease resource utilization. Therefore, to reach the best algorithm performance, the value of α is chosen by the simulation results.

In this simulation, the buffer size is 2 M and the message generation time interval is set to be 50 s. The value range of α is in $[0.1, 10]$. We change the value of threshold α and observe the changes of delivery probability, overhead, and latency.

Figure 3(a) shows the change of delivery probability with different threshold α. When the value of α is increasing, nodes have strong social relation and more meeting opportunities. It is highly possible that the encountered node is carrying a copy of received message, so destination node will generate ACK. In this way, acknowledgments are more likely to be sent. Therefore, the redundant copies are dropped more effectively so that the delivery probability improves. When threshold α keeps climbing up, more and more acknowledgments are generated. The acknowledgments occupy the buffer space, which decreases the delivery probability and then tends to be stable. It is shown that the optimal delivery probability can be obtained when the value of α is in $[1.0, 2.5]$.

Figure 3(b) illustrates the change of overhead when threshold α is different. When α is low, there are a lot of redundant copies so the overhead is high. With the increment of α, acknowledgments are spread in the network to inform other nodes to delete same copies, so the overhead reduces. When α keeps growing, the overhead also increases because



(a) Delivery probability changes with threshold α

(b) Overhead ratio changes with threshold α

(c) Latency changes with threshold α

(d) Buffer occupancy changes with threshold α

Figure 3: Performance metrics change with threshold α.

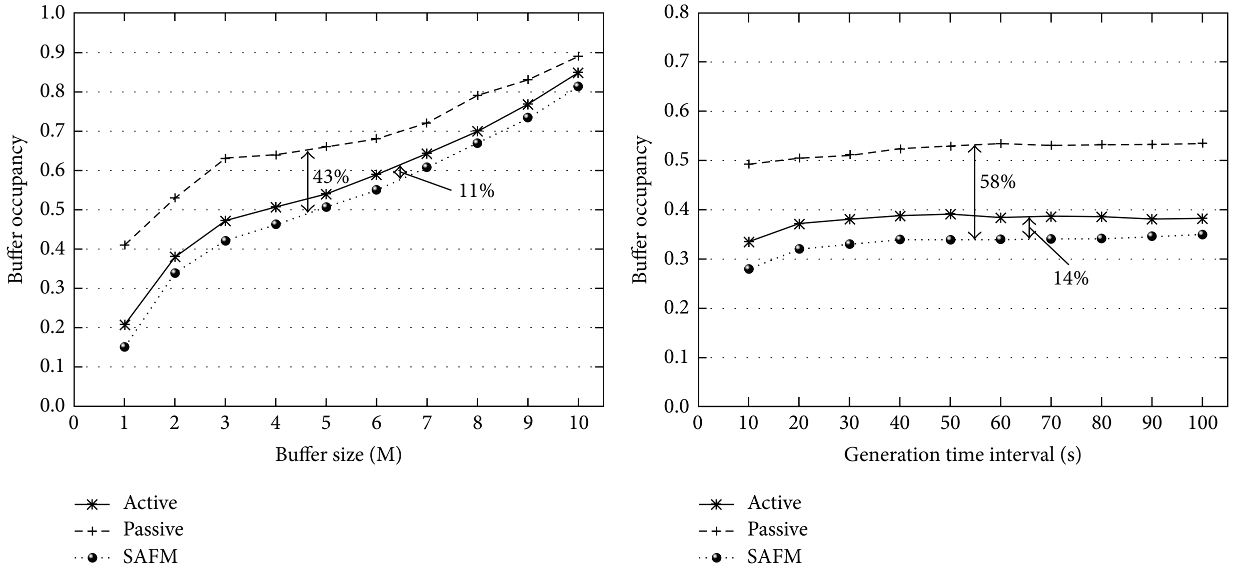
of a large number of acknowledgments and finally tends to be stable. It is shown that the overhead is the lowest when the value of α is in $[1.0, 2.5]$.

Figure 3(c) shows the change of latency with threshold α. When α is increasing, the number of redundant copies is cut down so that the waiting time is shorter. Therefore, the latency decreases. When α keeps growing, there are more acknowledgments in buffer, which increases the latency. The shortest latency can be obtained when the value of α is in $[1.0, 2.5]$.

It is shown in Figure 3(a) to Figure 3(c) that the optimal algorithm performance can be obtained when the value of α is in $[1.0, 2.5]$. In order to improve the resource utilization, we will change the value of α in $[1.0, 2.5]$ to verify the change of buffer occupancy.

The change of buffer occupancy with different value of α is shown in Figure 3(d). The approximate changing range is from 0.33 to 0.34. When α is small, acknowledgments are sent with low probability, resulting in high buffer occupancy. Subsequently, if α keeps growing, more acknowledgments are sent to reduce the redundancy. Therefore, the buffer occupancy is decreasing and finally tends to be stable. The lowest buffer occupancy can be obtained when α is 1.9. As a result, we choose the value of threshold α to be 1.9 in our algorithm.

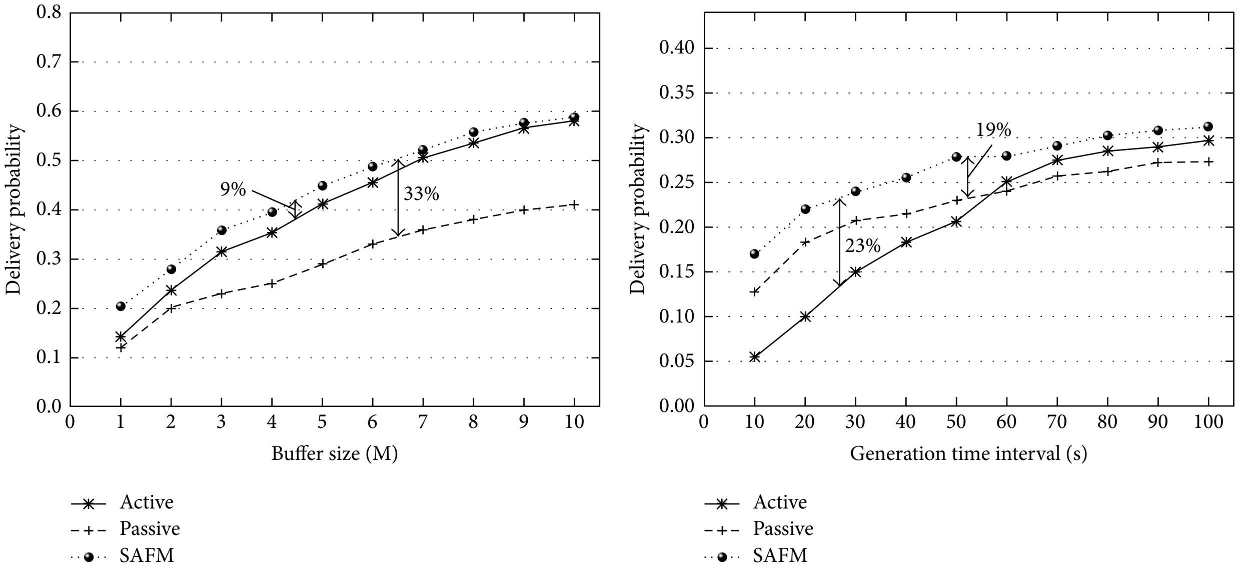
4.2. Performance Verification of SAFM. In this part, our proposed feedback mechanism SAFM is compared with active receipt and passive receipt [4]. In active receipt, destination node generates a corresponding acknowledgment of the message after receiving the message. Then, the acknowledgment is actively forwarded to the encountered nodes in the flooding way. Every encountered node receives the acknowledgment and then deletes the copies of the received message. However, this active receipt can cause great overhead. In passive receipt, the implicit acknowledgment of the received message is



(a) Buffer occupancy versus buffer size

(b) Buffer occupancy versus generation time interval

Figure 4: Buffer occupancy.



(a) Delivery probability versus buffer size

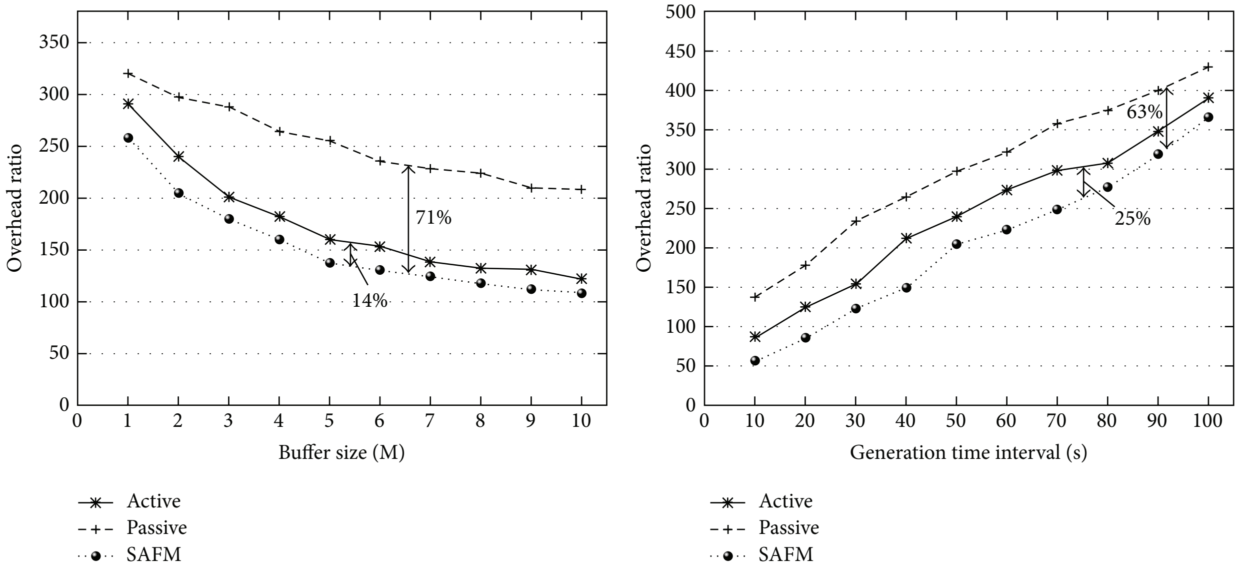
(b) Delivery probability versus generation time interval

Figure 5: Delivery probability.

generated. Different from active receipt, the acknowledgment will not be sent to encountered node until the encountered node tries to send the copy of received message. Therefore, the overhead caused by acknowledgment is low. But passive receipt is also inefficient in decreasing redundancy.

Two groups of experiments are made in our simulation to verify the performance of SAFM. In group A, the message generation time interval is fixed as 50 s. The nodes buffer size is changing from 1 M to 10 M to observe the changes of four performance metrics. Simulation results are shown in Figures 4(a), 5(a), and 6(a). In group B, the buffer size is set to be 2 M. Then, the generation time interval is increasing from 10 s to 100 s to verify the network performance with three feedback mechanisms. Simulation results are shown in Figures 4(b), 5(b), and 6(b).

In Figure 4(a), the buffer occupancy of three feedback mechanisms is changing when buffer size is increasing from 1 M to 10 M. Figure 4(b) shows the changes of three buffer occupancies with generation time interval. The buffer occupancy of SAFM is lower than that of active and passive receipt. This is because passive receipt generates the least acknowledgments. There are still redundant copies so the



(a) Overhead ratio versus buffer size

(b) Overhead ratio versus generation time interval

Figure 6: Overhead ratio.

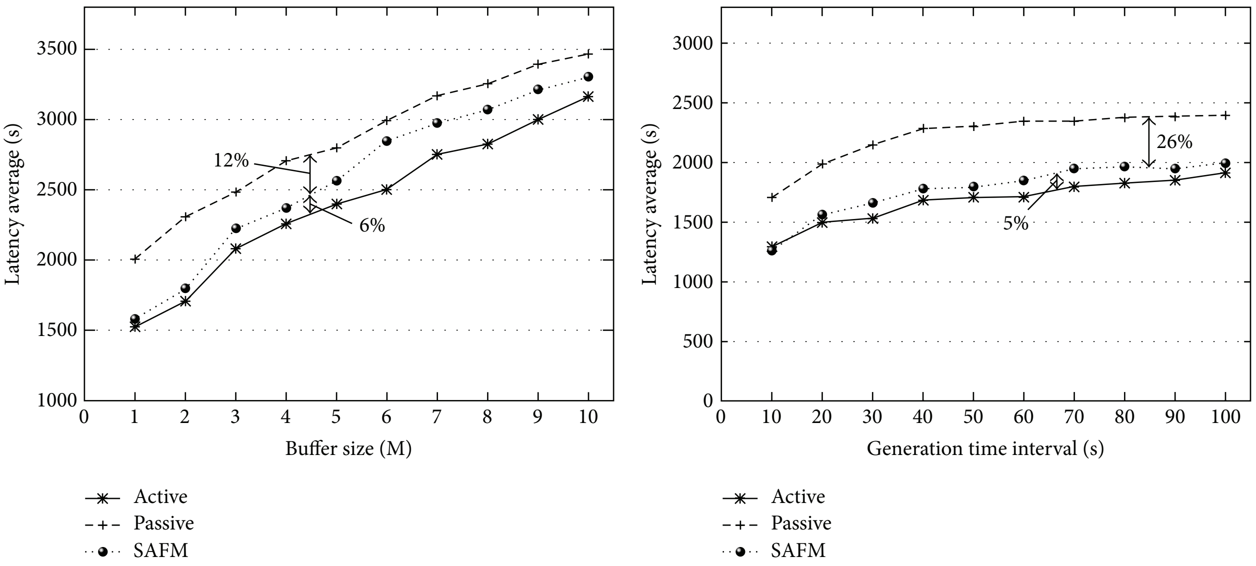
buffer occupancy is high in passive receipt, while in active receipt so many acknowledgments also occupy buffer space. So the resource utilization is also low. In SAFM, the number of acknowledgments is controlled based on social links, and the redundancy copies also are deleted at the same time. Therefore, SAFM can effectively reduce buffer occupancy. Figure 4(a) shows that the buffer occupancy of SAFM is 11% lower than active receipt and 43% lower than passive receipt. Figure 4(b) shows that the buffer occupancy of SAFM is 14% lower than active receipt and 58% lower than passive receipt.

Figures 5(a) and 5(b) show the changes of delivery probabilities with three feedback mechanisms. When the message generation time interval is short, the data density in network is high. Active receipt generates a lot of acknowledgments, so its delivery probability is the lowest one. With the increasing time interval, data density is decreasing and buffer space is becoming sufficient to avoid congestion. Redundant copies are effectively deleted in active receipt. So its delivery probability is growing to be higher than passive receipt. In SAFM, the number of acknowledgments is controlled. The congestion problem can be released especially when buffer size is limited. Therefore, its delivery probability improves obviously. Figure 5(a) shows that the delivery probability of SAFM is 9% and 33% higher than active and passive receipt. Figure 5(b) shows that the delivery probability of SAFM is 23% and 19% higher than active and passive receipt, respectively.

When buffer size is changing from 1 M to 10 M and generation time interval is increasing from 10 s to 100 s, the changes of overhead in three feedback mechanisms are shown in Figures 6(a) and 6(b). The overhead of passive receipt is the highest one since there are the least acknowledgments and most redundancy. The overhead of SAFM is lower than active receipt. This is because there are also plenty of acknowledgments in active receipt. When the buffer space is limited or data density is high, these acknowledgments occupied in the buffer space cause the message dropping, which increases the overhead. SAFM effectively controls the number of acknowledgments and reduces the redundancy at the same time. Therefore, the overhead largely reduces. Figure 6(a) shows that the overhead ratio of SAFM is 14% and 71% lower than active and passive receipt, respectively. Figure 6(b) shows that the overhead ratio of SAFM is 25% and 63% lower than active and passive receipt, respectively.

It is illustrated in Figures 7(a) and 7(b) that the latencies of three feedback mechanisms are changing with buffer size and generation time interval. In passive receipt, there are more redundant copies. Besides, the waiting time of message in buffer is long, so its latency is longer than the other two mechanisms. When buffer space is limited and generation time interval is short, the data density in network is high. The latency of SAFM is close to active receipt. When buffer space is becoming sufficient and generation time interval is longer, the latency of SAFM is a little increasing and growing longer than that of active receipt. This is because the number of acknowledgments is controlled according to social links of SAFM. Therefore, the number of redundant copies is more than that of active receipt. Redundant copies occupy the buffer space and increase the waiting time and transmission latency. However, the increasing latency is acceptable in DTSN. Figure 7(a) shows that the average latency of SAFM is 12% shorter than passive receipt and 6% longer than active receipt. Figure 7(b) shows that the average latency of SAFM is 26% shorter than passive receipt and 5% longer than active receipt.

4.3. Summary of Simulation. The simulation results verify that SAFM can achieve a higher delivery probability than that of active and passive receipt with lower buffer occupancy and overhead ratio under most circumstances. On average,



(a) Latency versus buffer size

(b) Latency versus generation time interval

Figure 7: Latency.

delivery probability of SAFM is 16% and 26% higher than that of active and passive receipt. Buffer occupancy is 12.5% and 50.5% lower than that of active and passive receipt. Meanwhile, overhead ratio of SAFM is 19.5% and 67% less than that of active and passive receipt. Besides, the average latency of SAFM is 19% shorter than that of active receipt, but only 5.5% longer than that of passive receipt.

5. Conclusion

In this paper, we focus on the feedback mechanism in the social applications of Delay Tolerant Networks and propose an improved social awareness based feedback mechanism called SAFM. This paper defines a new concept of social link to indicate the social relationship between nodes pair according to the encountering history, reflecting the social features of nodes' interaction in DTSN. Besides, SAFM is to reduce redundancy with the utilization of social links of nodes, reaching a tradeoff between overhead and delivery efficiency. Simulation results show that, in an acceptable range of delay, the proposed feedback mechanism improves the delivery probability, decreases the buffer occupancy, and reduces the overhead.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank NSFC (61572262, 61100213, 61170276, 61201160, 61201222, and 61401107); SFDPH (20113223120007); NSF of Jiangsu Province (BK20141427); Huawei Innovation Research Program (YB2014010048); NUPT (NY214097, XJKY14011); Open Research Fund of Key Lab of Broadband Wireless Communication and Sensor Network Technology (Nanjing University of Posts and Telecommunications), Ministry of Education (NYKL201507).

References

[1] Y. Yun, Y. Xia, B. Behdani, and J. C. Smith, "Distributed algorithm for lifetime maximization in a delay-tolerant wireless sensor network with a mobile sink," *IEEE Transactions on Mobile Computing*, vol. 12, no. 10, pp. 1920–1930, 2013.

[2] K. Fall, W. Hong, and S. Madden, "Custody transfer for reliable delivery in delay tolerant networks," Tech. Rep. IRB-TR-03-030, Intel Research, 2003.

[3] A. Pentland, "Socially aware computation and communication," *Computer*, vol. 38, no. 3, pp. 33–40, 2005.

[4] K. A. Harras and K. C. Almeroth, "Transport layer issues in delay tolerant mobile networks," in *NETWORKING 2006. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*, vol. 3976 of *Lecture Notes in Computer Science*, pp. 463–475, Springer, Berlin, Germany, 2006.

[5] P. R. Pereira, A. Casaca, J. J. P. C. Rodrigues, V. N. G. J. Soares, J. Triay, and C. Cervelló-Pastor, "From delay-tolerant networks to vehicular delay-tolerant networks," *IEEE Communications Surveys and Tutorials*, vol. 14, no. 4, pp. 1166–1182, 2012.

[6] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Tech. Rep. CS-200006, Duke University, 2000.

[7] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proceedings of the ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN '05)*, pp. 252–259, IEEE, August 2005.

[8] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the

multiple-copy case," *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 77–90, 2008.

[9] K. Wang and H. Guo, "An improved routing algorithm based on social link awareness in delay tolerant networks," *Wireless Personal Communications*, vol. 75, no. 1, pp. 397–414, 2014.

[10] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 7, no. 3, pp. 19–20, 2003.

[11] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: routing for vehicle-based disruption-tolerant networks," in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, pp. 1–11, Barcelona, Spain, April 2006.

[12] S. He, X. Li, J. Chen, P. Cheng, Y. Sun, and D. Simplot-Ryl, "EMD: energy-efficient p2p message dissemination in delay-tolerant wireless sensor and actor networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 9, pp. 75–84, 2013.

[13] A. Takahashi, H. Nishiyama, N. Kato, K. Nakahira, and T. Sugiyama, "Replication control for ensuring reliability of convergecast message delivery in infrastructure-aided DTNs," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 7, pp. 3223–3231, 2014.

[14] E. Bulut, Z. Wang, and B. K. Szymanski, "Cost efficient erasure coding based routing in delay tolerant networks," in *Proceedings of the IEEE International Conference on Communications (ICC '10)*, pp. 1–5, May 2010.

[15] J. A. F. F. Dias, J. J. P. C. Rodrigues, J. N. G. Isento, and J. Niu, "The impact of cooperative nodes on the performance of vehicular delay-tolerant networks," *Mobile Networks and Applications*, vol. 18, no. 6, pp. 867–878, 2013.

[16] V. N. G. J. Soares, F. Farahmand, and J. J. P. C. Rodrigues, "Performance analysis of scheduling and dropping policies in vehicular delay-tolerant networks," *International Journal on Advances in Internet Technology*, vol. 3, pp. 137–145, 2010.

[17] F. Xiao, X. Xie, Z. Jiang, L. Sun, and R. Wang, "Utility-aware data transmission scheme for delay tolerant networks," *Peer-to-Peer Networking and Applications*, 2015.

[18] A. Ali, T. Chahed, E. Altman, M. Panda, and L. Sassatelli, "A new proposal for reliable unicast and multicast transport in Delay Tolerant Networks," in *Proceedings of the IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '11)*, pp. 1129–1134, September 2011.

[19] D. O. Seo and D. H. Lee, "The end-to-end reliability algorithms based on the location information and implicit ACK in delay tolerant mobile networks," in *Future Generation Information Technology*, vol. 7105 of *Lecture Notes in Computer Science*, pp. 229–239, Springer, Berlin, Germany, 2011.

[20] Y. An, J. Huang, H. Song, and J. Wang, "A Congestion Level based end-to-end acknowledgement mechanism for Delay Tolerant Networks," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM '12)*, pp. 1574–1579, December 2012.

[21] E. Bulut and B. K. Szymanski, "Exploiting friendship relations for efficient routing in mobile social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 12, pp. 2254–2265, 2012.

[22] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques (ICST '09)*, pp. 1–10, March 2009.

[23] J. Ghosh, S. J. Philip, and C. Qiao, "Sociological orbit aware location approximation and routing (SOLAR) in MANET," *Ad Hoc Networks*, vol. 5, no. 2, pp. 189–209, 2007.

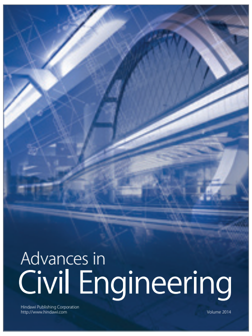
Journal of
Engineering

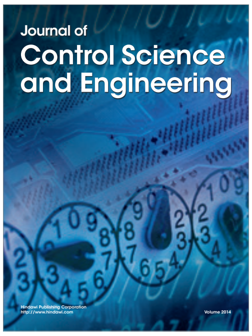
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

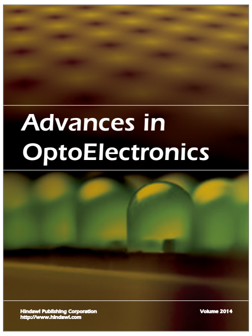
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

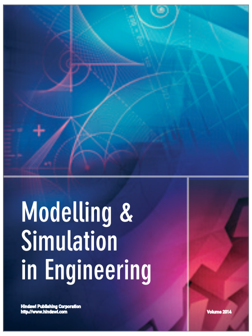
Journal of
Robotics

Journal of
Electrical and Computer Engineering

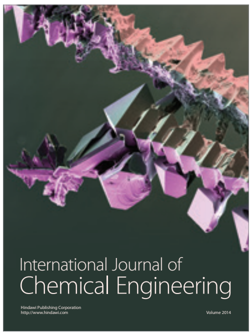

Advances in
OptoElectronics

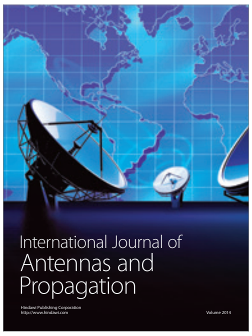
VLSI Design

International Journal of
Navigation and Observation

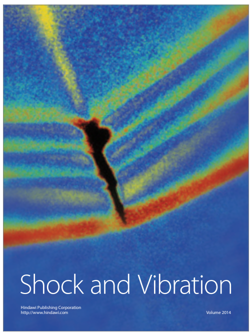

Modelling & Simulation in Engineering

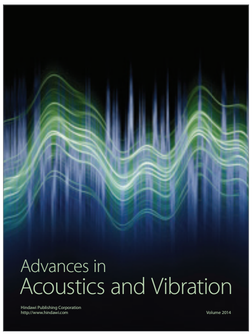
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration