Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2014, 6(5):2032-2036

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

A social selfishness compatible routing protocol in opportunistic networks

Li Liu

College of Information Science and Electrical Engineering, Shan Dong Jiaotong University, Jinan, China

---

ABSTRACT

More and more mobile devices are emerging in our lives. And the wireless applications and services based on mobile devices bring more flexible lives. However, the heavily overhead to 3G network obstacle the applications. Opportunistic Networks is an open field to settle this problem recently. It enables mobile devices to communicate in a circumstance where end-to-end connections are unavailable or unstable. Message routing and forwarding are based on nodes' cooperation through store-carry-and-forward manner by use of short message transmission technologies such as Bluetooth or WiFi. The existing of selfishness nodes make the performance decreased heavily. In this paper, based on Prophet, we consider social selfishness compatible routing protocol (SS-Prophet) in DTNs. Simulation and experiment demonstrate that SS-Prophet obtain higher performance in selfishness circumstance.

Keywords: DTNs, routing and forwarding, selfishness, social selfishness.

INTRODUCTION

With the proliferation of mobile devices with short-range networking interfaces, such as smart phones and personal digital assistant, more and more applications and services based on mobile devices are popular recently. However, the overhead of 3G network is heavily which is a challenging problem in wireless applications. Recently, many researchers found that Opportunistic Networks [1] is an effective method to settle this problem. In Opportunistic Networks, end-to-end paths are unavailable or unstable due to the mobility of nodes and lack of infrastructure. Messages are transferred from source nodes to destination nodes in a store-carry- and-forward manner by use of opportunistic meet opportunities. The routing and forwarding protocols are mainly based on the cooperation of nodes. However, in reality, most of nodes are selfishness due to limited resource such as power, bandwidth and buffer size, and private consideration [2]. Many researches show that a small proportion of nodes that do not forward messages, performance is heavily degraded [3-7].

Li et al. divide selfishness nodes into individual selfishness and social selfishness [8]. Individual selfishness nodes are unwilling to help other nodes forward messages. While social selfishness nodes are willing to help parts nodes which have some social relationships with themselves such as friendships, classmates or family members.

In this paper, we present a social selfishness compatible routing protocol based on Prophet (SS-Prophet). Prophet is presented by Lindgren in [9] which record and analyze history contact records and predict future meet probability. Nodes with social relationships such as friendship, family members, and classmates always meet frequently. On one hand, SS-Prophet considers social relationship between nodes which are willing to forward messages with each other. On the other hand, SS-Prophet utilizes history contact records to predict future contact opportunities.

The rest of this paper is organized as follows. We give a brief survey on recent incentive mechanism to individual selfishness and social selfishness in Section 2. We introduce a network model and architecture of SS-Prophet in Section 3 and give a detailed description of implementation of SS-Prophet in Section 4. In Section 5, we establish simulation and experiments to evaluate the performance of SS-Prophet, compared with Prophet in different proportion of selfishness nodes. Finally, we conclude the paper in Section 6.

2. Related Work

Most recent researches focus on incentive mechanism for individual which are generally classified into three categories: reputation-based, credit-based, barter-based.

Reputation based incentive mechanism relies on identifying misbehaving node and excluding them from the network according to nodes' reputation records. Nodes can build up their good reputation records by forwarding packets for others, and are rewarded with higher priority when transferring their own packets. The reputation records decrease when it misbehaves. The bad reputation node is detected and excluded when its reputation is decreased below the threshold.

Credit-based incentive mechanisms introduce virtual currency or credit to regulate the packet-forwarding relationships among different nodes. Nodes need credit to pay for message delivery requirement and earn credits by forwarding packets from other nodes.

Barter-based incentive mechanisms are mainly based on game theory which is found to be fit to tackle the selfishness considering fair exchange process, which is also called barter based process.

Social selfishness is presented by Li et al. in [8]. They consider users' willingness and present a social selfishness aware routing (SSAR) from philosophy of "design for user". In SSAR, a node only forwards packets for those with social ties. And it gives priority to packets received from those with stronger social tie when there are not enough resources. To improve performance in further, SSAR formulates the forwarding process as a Multiple Knapsack Problem with Assignment Restrictions (MKPAR). It provides a heuristic-based solution that forwards the most effective packets for social selfishness and routing performance.

3. Overview of SS-Prophet

Network Model. We design a network model to represent nodes' movement and social relationships. We assume a network model with 60 nodes which are divided 4 groups. Nodes in one group have social relationships such as friendship. That is, one group represents a community. Nodes in different groups have no social relationships. Generally, nodes in a community meet frequently than out of communities. We assume nodes move in 10×10 areas. And we divide area into 4 sub-areas. Each group selects a sub-area as its sensitive sub-area where nodes in this group appears more frequently. For social selfishness, nodes in same community are willing to forward for each other. However, nodes in different community are unwilling to forward for each other.

Architecture of SS-Prophet. SS-Prophet combines social selfishness and Prophet into together. Form social selfishness point, nodes are willing to forward to their friend nodes. From contact point, nodes are willing to select higher meet probability with destination nodes as forwarders. SS-Prophet consists of three main parts: social relationship and social tie, forwarding algorithm and message priority control.

Social relationship and social tie maintains social information between nodes in same community. Generally, nodes will record their friend nodes or nodes with social relationships. By use of Prophet, nodes can record contact history record, then analyze how the social tie strong. If two nodes contacts frequently, their social tie will be big. On the contrary, the social tie will be small if two nodes contact scarcely. The social relationship and social tie information are important information to select forwarder.

Forwarding algorithm decides the forwarding messages list. Due to the existing of selfishness nodes, the messages may be rejected by selfishness nodes. Thus, the forwarding algorithm in SS-Prophet is more complicated. It needs be judged from 4 conditions. We will give a detailed description in the following section.

Message priority control is responsible for messages' transmission sequence in order to improving forwarding performance in further. Considering the limited transmission time due to the movement of mobile nodes, the transmission sequence has big influence on the forwarding performance. We wish the messages which have higher meet probability be transmitted firstly. It will increase the probability of successful delivery.

4. Implement of SS-Prophet

Social Relationship and Social Tie. We assume that nodes know each other as friends in community. Nodes in same community are willing to forward messages for each other, even if they are selfishness nodes. This is social selfishness. However, the social tie strengths among nodes are different. Prophet records nodes' connect record which indicate the future meet probabilities. At the same time, social tie strength among nodes in community is computed by using (1) as follows. Where T is a time window, n is the meet times between two nodes in T. An evaporation process is necessary for the degree by (2). γ and k is evaporation factor.

$$\deg = \sum_{t=0}^{T} n \quad (1)$$

$$\deg\_new = \deg\_old \times \gamma^{\kappa} \quad (2)$$

Forwarding Algorithm

Whether to forwarding about a message, SS-Prophet relies on 4 conditions. For simplify description, we assume that there are two nodes NA and NB in transmission range.

Con.1: NA is the destination of Message. When NA is the destination of message in NB's message list, NA must accept the message whether it is selfishness node or not.

Con.2: P(A,D)>P(B,D). The function P(A,D) indicate the meet probability between NA and ND (destination node) in the future. While P(B,D) indicate the meet probability between NB and ND (destination node) in the future. If NA has more probability to meet ND, it is more fitful for forwarder than NB. Thus, NA will forward the message in theory. However, in reality, it needs to consider whether NA is a selfishness node.

Con.3: NA is selfishness. If NA is a selfishness node, it only forward for nodes in same community. And it will not forward for nodes in different communities. If NA is not a selfishness node, it will forward for all other nodes.

Con.4: NA and NB is in same community. This condition is worked when NA is selfishness node and judge whether NA forward for social selfishness nodes.

When two nodes are in transmission range, they exchange message according to SS-Prophet, which consists of 6 steps. And we give the detailed introduction from NA's aspect.
1) NA firstly checks whether they are in same community. For example, if NB is one of friends, NA maintains the social information for NB and meet information for future meet prediction. Otherwise, NA only maintains meet information for NB.

2) Exchange message list for each other between NA and NB. The message list consists of the destination information (ND) and current meet probability. For example, message in message list of NB has a meet probability between NB and ND (supposed P(B, D), which is mainly based on degree in (ND, degree) in NB.

3) For each message of NB, NA checks whether NA is the destination. If NA is the destination, the message is delivered to NA. Thus, this message must be added to forwarding list. Otherwise, if NA is not the destination, go to

4) For each message, SS-Prophet computes the meet probability between NA and ND in the future, supposed P(A, D). If P(A, D)> P(B, D), the message is decided to deliver from NB to NA according to 5). Otherwise, the message does not be forwarded by NA.

5) If NA is selfishness node and NA is a friend of NB, the message is added to forwarding list. If NA is selfishness node and NA is not a friend of NB, the message will be rejected to forward by NA. If NA is not a selfishness node, the message will be added to forwarding list.

6) Decide the transmission sequence and Start to transmission.

Table 1. SS-Prophet routing protocol is outlined in Algorithm 1

Algorithm 1 Pseudo-code of SS-Prophet routing protocol
NA meet NB
If (NB and NA are in same community)
maintain social information about NB in NA
Endif
Maintain meet inforamtion about NB
For (message M in message list of NB)
If (ND of M = =NA)
Add M to forwarding list
Else
If (P(A, D)>P(B,D)
If (NB and NA are in same community)
Add M to forwarding list
Endif
Else
Do not add M to forwarding list
EndElse
EndIf
Else
Do not add M to forwarding list
EndElse
Endfor
order the delivery list
For (M in delivery list)
transimite M from NB to NA
Endfor

Message Priority Control. When delivery list is decided, the transmission sequence will be ordered according to message priority control.
1) Messages from nodes in same community have highest priority. In the same circumstance, the nodes with bigger social tie have higher priority.
2) Messages from nodes in different community will be ordered by future meet probability function. The bigger values will have higher priorities.

5. Simulation and Evaluation

The experiment is carried out through the Opportunistic Network Environment (ONE) Simulator [10]. In the simulations, we use delivery ratio and average latency as the performance metrics. We compare the effectiveness of SS-Prophet with Prophet in 30% selfishness nodes, 70% selfishness nodes and 100% selfishness nodes. The values of simulation parameters are illustrated in Table1.

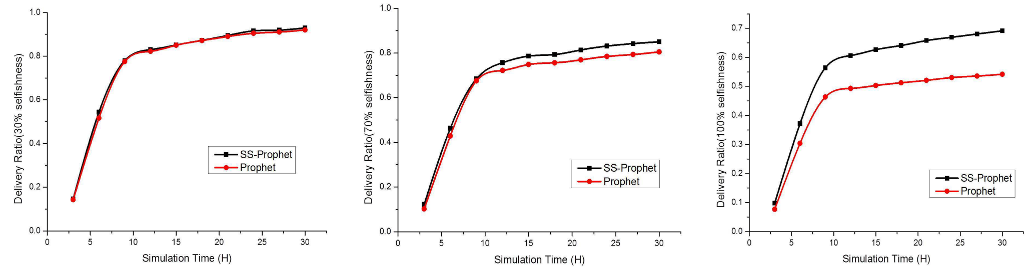
Fig.1 shows the delivery ratio of SS-Prophet and Prophet in 30% selfishness nodes, 70% selfishness nodes and 100% selfishness nodes respectively. As we have seen, SS-Prophet obtain higher delivery ratio than Prophet in every conditions. For instance, in 27 hours with100% selfishness, SS-Prophet's delivery ratio is 68.06% which is higher than Prophet with 53.57%. Compared to best performance (all cooperate), SS-Prophet's performance is decreased with the increasing number of selfishness nodes. However, it improves performance at some extent.

Fig.2 shows the average latency of SS-Prophet and Prophet in 30% selfishness nodes, 70% selfishness nodes and 100% selfishness nodes respectively. As we have seen, the average latency of SS-Prophet and Prophet are similar when selfishness nodes are 30% and 70%. However, the delivery ratio of SS-Prophet is similar to Prophet in 30% selfishness nodes and higher than Prophet in 70% selfishness nodes. In Fig 3(c), the average latency of SS-Prophet is longer than Prophet when 100% selfishness nodes. However, the delivery ratio of SS-Prophet is far higher than Prophet.

In a word, the forwarding performance of SS-Prophet is higher than Prophet in different portion of selfishness nodes exist.

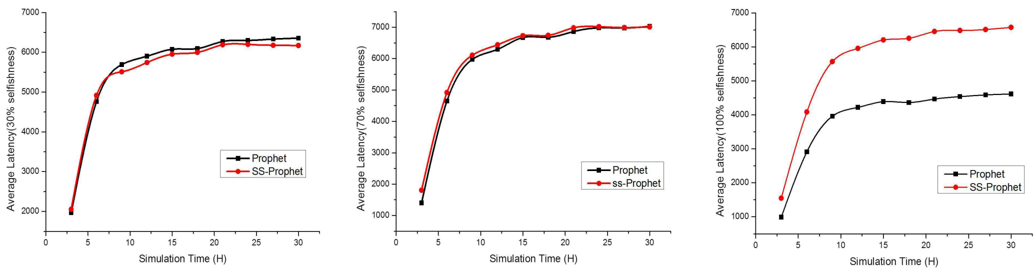
Table 2. Simulation Parameters

Simulation Parameters	Values
Simulation Time	30Hours
Warm Time	5000seconds
Nodes Number	60
Interface Type	Bluetooth
Transmit Speed	250KB
Transmit Range	10M
Message size	10M
Message TTL	5Hours
Nodes' Buffer	10MB
Prophet	Pinit = 0:75; α= 0:25; β= 0:98



(a) 30% selfishness nodes. (b) 70% selfishness nodes (c) 100% selfishness nodes

Fig. 1 Delivery ratio comparison



(a) 30% selfishness nodes. (b) 70% selfishness nodes (c) 100% selfishness nodes

Fig .2 Average Latency comparison

CONCLUSION

In this paper, we present a social selfishness compatible routing protocol based on Prophet (SS-Prophet). SS-Prophet combine social selfishness information and contact information into together in order to decide whether messages are forwarded by selfishness nodes. Simulation shown that the delivery ratio of SS-Prophet is higher than Prophet when some portion of selfishness nodes are in the network.

Acknowledgements

This work was financially supported by the Shandong Jiaotong University Science Research Foundation (Z201305, Z201212).

REFERENCES

[1].M. Conti, S. Giordano, M. May, and A. Passarella, *Communications Magazine*, **2010**;48(9):126-139.

[2]. Jingwei Miao, Omar Hasan, Sonia Ben Mokhtar, et al. *International Journal of Information Management*, **2013**;33(2):252-262.

[3]. Y. Li, P. Hui, D. Jin, L. Su, and L. Zeng, *Communications Letters, IEEE*, **2010**;14(11):1026-1028.

[4]. Y. Li, G. Su, D. O. Wu, D. Jin, L. Su, and L. Zeng, *IEEE Transactions on*, **2011**;60(5):2224-2238.

[5]. Y. Li, P. Hui, D. Jin, et al. **2011** *IEEE International Conference* on, 2011:1-5.

[6]. A. Panagakis, A. Vaios, and I. Stavrakakis, On the effects of cooperation in dtns, in Communication Systems Software and Middleware, **2007**. 2nd International Conference on, 2007:1-6.

[7]. Y. Li, G. Su, and Z. Wang, AEU-*International Journal of Electronics and Communications*, 2012;66(1):62-67.

[8]. Q. Li, W. Gao, S. Zhu, and G. Cao, Ad *Hoc Networks*, **2012**;10(8):619-632.

[9].A. Lindgren, A. Doria, and O. Schel´en, *Comput. Commun. Rev*., **2003**;7(3):19-20.

[10]. A. Ker¨anen, T. K¨arkk¨ainen, and J. Ott, *Journal of Communications*, **2007**;5(2):92-105.