http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (19): 5036-5043, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.5036.5043

An Improved Spray and Wait Routing Algorithm Based on Node Performance in Delay Tolerant Networks

Chenqu Dai, Jianbo Li, Lei You and Jixing Xu
Information Engineering College, Qingdao University, Ningxia Road 308, Qingdao 266071, Shandong, China

Abstract: Delay Tolerant Networks (DTNs) are one of the emerging mobile wireless networks that cannot set up the end-to-end communication path between the source and destination node pairs in most of time. Due to the main characteristics of intermittent connectivity and long delay, the traditional routing algorithms for the Internet and Mobile Ad-Hoc Networks do not perform well in DTNs. In this paper, we propose an improved Spray&Wait Routing Algorithm Based on Node Performance (NPSW), which is mainly evaluated by the number of encountered nodes in a period. In addition, we also design a utility metric to select next hop node(s) with the consideration of the combination of encountering count to the destination node and the last contact time. Extensive simulations have been conducted and the results show that our proposed method can achieve a higher bundle delivery ratio and incur much lower communication overheads. As compared with other existing DTN routing protocols, the results also indicate that the NPSW scheme achieve a better routing performance.

Key words: Delay tolerant networks, routing algorithm, node performance, encounter frequency

INTRODUCTION

Delay Tolerant Networks (DTNs) are a kind of emerging mobile wireless networks that experience frequent network partitions and extremely long end-to-end delays, where there does not exist a instantaneous complete end-to-end path between pair nodes (Fall, 2003). The concept of DTN has a broad range of applications in many real scenarios such as Military Networks (Malladi and Agrawal, 2002), Deep Space Networks (Burleigh *et al.*, 2003), Wildlife Tracking System, Vehicular Communication (Burgess *et al.*, 2006) and Internet Access in rural areas etc. In such networks, a path from a source to a destination can be highly unstable and change or break frequently before it has been discovered and the delay would become unpredictable and very large.

Many potential DTN applications and communications are influenced by challenges including high node mobility, low node density, environmental interference and obstruction, short range radio and malicious attacks (Prodhan *et al.*, 2011). In general, the conventional TCP/IP or Ad-Hoc (Xie *et al.*, 2002; Bae *et al.*, 2000) network routing protocols have been done much. However those routing schemes are so difficult to directly be applied to DTNs because of its characteristic of intermittent connectivity and lack of highly stable paths between nodes.

There are many efficient routing algorithms proposed to enable messages transmit in those opportunistic networks environment. For example, the Epidemic protocol presents that when two nodes encounter, they exchange message vector each other. This scheme would get a high delivery rate in a short run, but at the expense of high overhead ratio and large dropped messages due to the limited buffer size. PROPHET adopts history of node encounters to predict the delivery probability to destinations. However, it still waste a lot of network resources. Spray and Wait restrict the initial number of copies of messages in the spray phase and directly transmit messages in the wait phase to improve flooding-based schemes.

To this end, we propose a novel improved protocol NPSW based on Spray and Wait, which could dynamically control the direction of message flow with the consideration of history encounter information. The main idea embodied two parts. On the one hand our scheme adopts a specific utility metric to select a better candidate node as the next hop. The purpose is to enable messages delivery more efficiently since the initial number of messages copies has been set. On the other hand, this scheme designs a dynamic spray method according to calculating the encountering count with other nodes of these adjacent nodes. And the candidate node(s) can help to spray the message quickly, especially in extreme environment. In addition, we also allow nodes with

Corresponding Author: Jianbo Li, Information Engineering College, Qingdao University, Ningxia Road 308, Qingdao 266071, Shandong, China

single message copy to forward to other relay node(s) in the wait phase, since it doesn't add the copy number actually.

Our main contributions are listed as following:

- An improved spray routing scheme is proposed based on the node performance. In our protocol, source or relay nodes carrying information choose the next hop node based on node performance. And the node performance is expressed by the ability of contact frequency with other nodes in history and the potential to spread message copies as fast as possible. Especially in special network scenarios, where nodes are inhomogeneous distribution, our protocol can achieve a good behavior
- In this study, we apply the contact frequency given in Definition 1 to forecast the odds of successful transmission to destination. However, we also set a threshold value τ of contact frequency $v_{k,i}$. When two nodes meet with lower $v_{k,i}$ than τ, we design another metric the last encounter time to measure the deliver probability
- A novel asymmetric spray mechanism of allocating copy tickets to relay nodes is adopted in this paper, based on the node activeness that explained in Definition 3. The set E_a contains nodes which n_a once contacted in history, so the greater weight of set E_a indicates that n_a is more active. Therefore, our spray scheme distributes more message copy number to who has a better activeness in spray phase
- In this study, we also improved the wait phase. Although the copy number is left only one, the carrier is allowed to forward to another node that has a higher potential to destination

The rest of study is organized as follows. In section II, we give a detailed description of our NPSW scheme. The simulation evaluations and analysis of results are presented in Section III. And finally in section IV we conclude our paper and give future research guidelines.

NODE PERFORMANCE-BASED SPRAY AND WAIT SCHEME

In this study, we propose a novel improved Spray and Wait algorithm based on node performance, named NPSW. Our NPSW scheme purely relies on node encounter history information rather than the global

Table 1: Mathematical Notations

Notation	Meaning
N	The set of all the nodes in the network
M(a)	The set of all message of node na
nk	The node named nk
mi	The message named mi
Vk	The vector consisted of vk,i
Tk	The vector consisted of tk,i
Ek	The vector consisted of ni
vk, i	The contact frequency between node nk and node ni
tk,i	The last encounter time between node nk and node ni
TTL(mk)	Time-to-live for the message mk
t	The threshold of vk,i

knowledge or the summary of link states of the network topology, which actually makes NPSW be more flexible and practical. As known, the most notable feature of DTNs is the characteristic of frequent partitions and intermittent connectivity intrinsic, so the network topology is always changing and becomes very difficult to be catched.

In the first section, we give the mathematical notations used in this chapter in Table 1. Then we also design definitions of contact record and the way to collect information at section A. Next we present the execution procedure of NPSW that is how to choose relay nodes and how many message copies should be delivered in spray phase. Finally, we explain the new forwarding method in wait phase.

Node performance metric: We first define several utility metrics for each node to denote the node performance. The node performance contains the delivery probability to forward messages to destination and the node activeness which depicts how fast nodes spread message copies. Different from PROPHET, we adopt the contact count with destination node as the delivery probability. We assume there is a vector in each node recording encounter information, including last encounter time, contact counter and so on. And we apply node activeness to describe the ability to got connect opportunities. In other ways it can be said how many different nodes it encountered once. In many DTN scenarios the mobility of nodes is not complete random, such as in Wildlife Tracking System, animal nodes get used to gather together. In Deep Space Networks, planet nodes move regularly according to the given orbit. Therefore, the asymmetric spray we proposed is more suitable for such DTNs. Besides, we show the protocol how to collect all the routing information need by a node n_a in Algorithm 1 as follows:

Network model: Node set: N= $\{n_I|1<i<|N|\}$
Message set: M(a) = $\{m_i|1<I<|M(a)|\}$

where, N is a node set which contains all nodes in the network. And M(α) in each node is a message set that notes all messages carried on the node buffer space. Commonly, each set N of each node is the same, whereas every M(α) in every node is mostly different each other.

Contact records: Definition 1. Contact frequency record. Each node n_k maintains a vector:

$$V_k = [v_{k,1}, v_{k,2}, \ldots, v_{k,n}]$$

where $v_{k,1}$, means the contact frequency between node n_k and node n_i and we define the initial value $v_{k,i}$ is zero before the first touching between pairware nodes. Every time node n_k meets node n_i, the value of $v_{k,i}$ plus one. The set vector V_k is consist of $v_{k,i}$ and maintained in every node, which used to predict the delivery probabilistic to other nodes.

Definition 2. Last encounter time record: Each node n_k maintains a vector:

$$T_k = [t_{k,1}, t_{k,2,\ldots,} t_{k,n}], t_{k,1>0}$$

where, $t_{k,i}$, is the last encounter time between node n_k and node n_i. In the beginning, all elements $t_{k,i}$, are set zero. Once node n_k, meets n_i the vector updates the current time as the last encounter time $t_{k,i}$. This variant is another measuring standard that use for selecting relay node(s), only when both $v_{k,d}$ and $v_{k,\beta}$ are less the threshold τ. Who has a bigger $t_{k,i}$, is a better candidate node.

Definition 3. The set of encountered nodes: Each node n_k, maintains a set:

$$E_k = \{n_i|\ v_{k,i} \neq 0 \text{ and } 1<I<n\},\ 0<|E_k|<|N|$$

where, this set records every node it meets. A pair of nodes should firstly examine whether E_k includes each other, when they contacts. If node n_a meets a new node n_b that is not contained in E_a, then add n_b into the set E_a. Meanwhile, n_a should also be added into set E_b. The initial weight of E_k is zero and the value of $|E_k|$ is not greater than that of |N|, namely the number of all nodes in network. We employ this metric to allocate message copy tickets to the next hop node(s) in spray phase.

Calculation and update of metric: When a pair nodes n_a and n_b encounter, they firstly exchange message vectors M(a) and update those utility metrics each other, for example $v_{a,b}$, $t_{a,b}$,E_a and $v_{b,a}$, $t_{b,a}$, E_b. To begin with, node n_a needs to check if n_b is in its E_a set. If not, then add n_b into the set. Besides $v_{a,b}$ pluses one in turn and $t_{a,b}$ is reset by current time when they contact. Similarly, node n_b should also do the same thing like n_a. The detailed procedure is illustrated in Algorithm 1.

```
Algorithm 1. Collecting Information
Require: current_time
Ensure:
for ∀ node n_a _ N do
 if v_a,b ≠ 0 then
E_a←E_a _¾{n_b}
 end if
 v_a,b← v_a,b +1
t_a,b←current_time
end for
```

ADAPTIVE SPRAY PHASE

In this process, we describe how our NPSW scheme chooses the next hop node and delivers multi-message-copies. In our NPSW scheme, the spray phase is divided into two aspects. First, we offer an adaptive technique dynamically to select high performance relay nodes. This technique is not only helpful to spreading copy tickets as quickly as possible, but also ensures that those nodes which are more likely to contact destination carry messages. Next, we calculate to distribute the number of message copies that should be delivered to candidate node (s). Those nodes that seem to meet other nodes frequently will be allocated more message copies.

When the resource node or relay node n_a which is conveying message m_i to destination n_d encounters node n_b, they firstly exchange their information vectors V_a, T_a, E_a and V_b, T_b, E_b for each other. Primarily we must be sure that there is not the m_i contained in the storage space of n_b and update parameters. Then the carrier node chooses relay node(s) to asymmetrically replicate message copies.

Node selecting scheme: Normally, for each node, the delivery potential to the same destination node is always different and the contact frequency $v_{k,i}$ that is calculated by encounter count between a pair of nodes varies, especially in extreme environment where nodes move non-randomly. Therefore, we propose an approach to help carrier nodes to select candidate node (s) for each message in order to improve the successful deliver ration. In detail, when the value of $v_{b,d}$ is greater than that of $v_{a,b}$, the node n_a sprays message copies to n_b. It is because the more encounters in history indicate the higher meeting potential in future.

Nevertheless, the contact frequencies of both nodes might be a little low so that $v_{k,i}$ can't be good and accurate enough to forecast the delivery potential. It is hard to determine to replicate message. For instance, Node n_a and n_b might meet destination node n_d only one time in a period time. In such condition, it seems hardly to make decisions due to the unsteady and little data. Based on the above analysis, another utility metrics $t_{k,i}$ and τ is proposed, where $t_{k,i}$ means the time when node n_k and node n_i contacted and τ is the threshold of $v_{k,i}$. With the moving of nodes, n_k and n_k have not yet encountered from $t_{k\text{-}i}$ to now, so we apply $t_{k\text{-}i}$ to express the "distance" between n_i and n_i. The bigger $t_{k,i}$ is, the farther the distance is. As a consequence, if and only if both $v_{i,d}$ is littler than τ, we decide to compare the last encounter time. That is a carrier node would spray message copies to nod who has a higher value of $t_{k,i}$ than itself.

Replicas distributing scheme: The key idea in this section is how to allocate message replicas to candidate node (s). Although the buffer size of nodes is the same, the movement of nodes is different. Therefore, some nodes are able to meet more nodes, otherwise some nodes have little encounter opportunities. If we distribute the same count message copies to two kinds of nodes, a part of message copies cannot be spread fast for a period time. Besides, to unequally spray the message copies without any consideration is unreasonable.

To this end, we adopt the variant, node activeness, a main metric of node performance. We apply the metric E_k mentioned in the Definition 3 to express node activeness. The larger weight of set E_k presents this node contacted more nodes. Furthermore, the higher weight indicates the better node performance since more encounter opportunities would happen to this node. That is to say that a node who is more active is likely to spread message copies more quickly. Thus, we should replica more copies to such kind nodes.

Therefore, we propose a replica distributing scheme with the help of metrics as shown in formula (4), based on which we presents a novel copy ticket spray mechanism. Node n_a carrys L_a copies of message m_i whose destination is n_d and L_a isn't equal to one. If node n_a chooses node n_b as the next hop, node n_a will spray L_b copy tickets of message m_i to node n_b. According to our analysis above, whose activeness is better should be allocated more number of copies and lower activeness node gains less copies. We utilize the ratio of the weight of set E_a and the sum total of weight of E_a and E_b as allocation proportion to node n_a. The reminder message copies are forwarded to n_b. The Eq. 4 shows us the detailed calculation process. In addition, the specific process of spray mechanism we present is illustrated in Algorithm 2:

$$\left\{\begin{matrix} L_{anew} = \left[\dfrac{|E_a|}{|E_a|+E_b}\right].L_{aold} \\ L_{anew} = L_{aold} + L_{bold} - L_{anew} \end{matrix}\right\} \tag{4}$$

where, L_a is the number of message copies. L_{aold} presents the number of copies before carrier node deliver to next hop. And L_{bnew} shows the number of copies that should to be allocated to n_b:

```
Algorithm 2. Spray Routing Protocol
Require: n_a meets n_b
Ensure:
exchange V_A,E_a, T_a and V_b,E_b, T_b for each other
for ∀ message m_k∈ M(a) do
 d_©destination node of m_k
 L_anew ← |E_a| / |E_a + E_b| L_aold
 L_bnew = L_aold+L_bold- L_anew
 if max{v_a,d, v_b,d}<ô AND t_a,d< v_b,d then
 forwardList.add(m_k)
 else if v_a,d < v_b,d then
 forwardList.add(m_k)
 end if
 if L_bnew = 0 then
 forwardList.deleteMessage(m_k)
 end if
 forwardList.get(m_k).update(L_bnew)
 buffer.get(m_k).update(L_anew)
 forwardList.sort(ascending, TTL)
 if L_anew =0 then
 buffer(n_a).deleteMessage(m_k)
 end if
end for
```

FORWARD PHASE

In Spray and Wait algorithm, when a node only has a single copy of a message left, it switches to the wait phase, in which it allows the node to forward the message directly to destination node. However, this process might bring large end-to-end latency, especially when the initial copy ticket number is small. For instance, the relay node is carrying one message copy, while the relay node is far from destination. On this occasion, it is suitable for the relay node to forward message copy to another node that has a higher contact frequency $v_{k,d}$ with destination. In fact, the main motivation to design such scheme in wait phase is to improve the delivery ratio without the increase of message copies number in the network. In addition, the whole flow chart of our proposed protocol NPSW is displayed in Fig. 1.

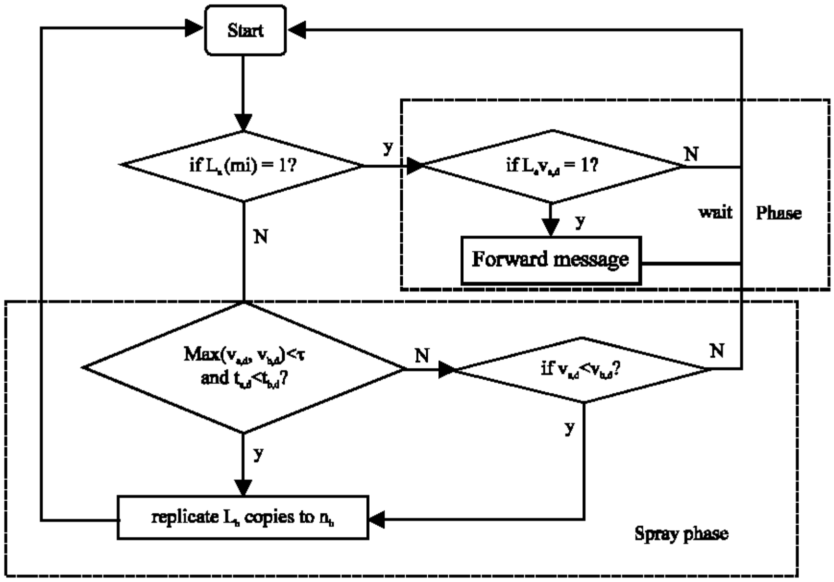

Fig. 1: Flow chart of the NPSW procedure

SIMULATION AND RESULTES

In this section, we evaluate the performance of our NPSW algorithm by implementing it via the ONE simulator (Keranen *et al.*, 2009). Besides, we also have focused on evaluating the Epidemic, Prophet and Binary Spray-and-Wait for performance comparison. In this simulation experiment, we use four performance metrics in order to measure the performances of our algorithm as follows.

We investigate the variations of these metrics with consideration to the buffer size of nodes, the live time of a message and the message generated time interval. The main parameters are listed in Table 2.

Figure 2(a) shows the impact of the node buffer size on the simulation of the delivery ratio. As shown in Fig. 2(a), the deliver rate of four routing schemes all increases in the wake of the enlargement of buffer size, because a bigger buffer size could carry more messages. We also observe that our protocol achieves a higher delivery ratio than other algorithms regardless of the different buffer size, since our scheme takes account of different performance of nodes to forward messages and copies of messages are distributed unequally in the movement of nodes.

Figure 2(b) describes the variation in overhead ratio with respect to the buffer size. It significantly shows our proposed scheme and Binary Spray and Wait work more effectively than others, since the number of message copies has been limited initially but Epidemic and Prophet might produce large copies in network during routing procedure.

Table 2: Simulation settings in Random Walk model

Parameter	Value
Area size	500×500 m
Number of nodes	20
Transmit radius	100m
Message size	500K
Message interval	20,30,40,50,60(s)
Transmit speed	250Kbps
Moving speed	0.5-1.5 (m/s)
Node buffer size	2M,4M,6M,8,10M
Time-To-Live(TTL) (min)	60,90,120,150,180
Simulation time	12 h

However, in Fig. 2(c) the average latency of our scheme is a little higher than other three protocols as the increase of buffer size. Since in spray process of our NPSW, source or relay nodes need to gather information of adjacent nodes to estimate their performance. From this figure, we can conclude that no matter how the buffer space enlarges the overhead of NPSW and BSW just change slightly.

As can been seen from the Fig. 2(d), the average hopcount of our NPSW is bigger than Prophet and BSW whereas smaller than Epidemic totally. This is due to the novel forwading protocol mechanism during wait phase, which increases the average count of hop. Nevertheless, at the cost of network resource and average hopcount, our NPSW has a higher delivery ratio.

The results in Fig. 3 are simulated in the condition of buffer size as 8MB. As illustrated in the Fig. 3(a), it is obvious to find our proposed algorithm perform better than other schemes regarding to higher delivery ratio. The delivery ratios of BSW and NPSW grow slowly along with the change of messages' TTL(Time-To-Live). Since the

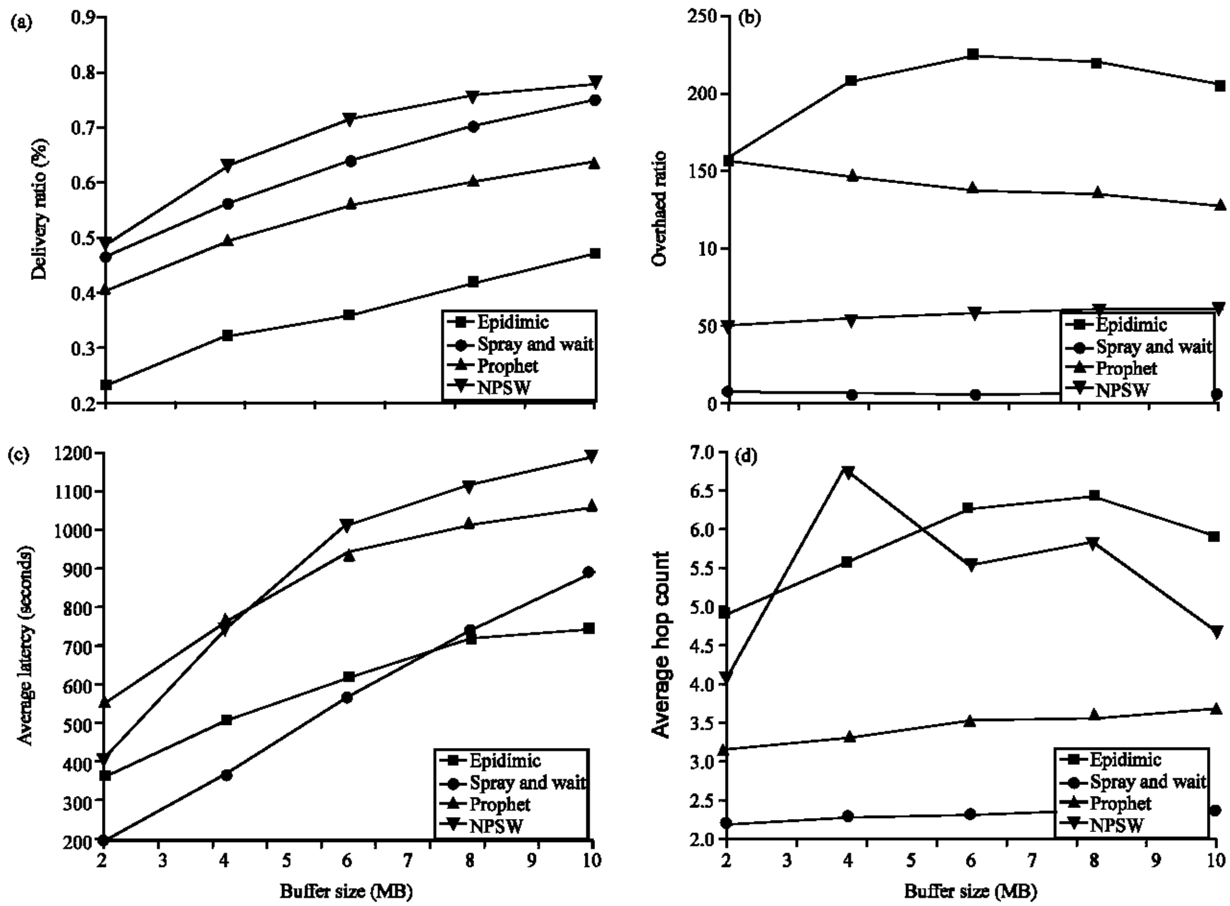

Fig. 2(a-d): Buffer size vs delivery ratio, overhead, average latency, average hopcount (a) Buffer size vs delivery ratio (b) Buffer size vs overhead (c) Buffer size vs latency (d) Buffer size vs hopcount

buffer size and network resource are limited, there will be more copies of messages as the time goes by.

In Fig. 3(b), we can easily observe that the overhead of our proposed algorithm still outperforms the compared algorithms Epidemic and PROPHET because of its decision mechanism and dynamic spray phase. NPSW spends more cost than BSW due to the addition of forwarding protocol in wait phase. Although messages live longer, the overhead of NPSW and BSW almost remains the same. It attribute to the restriction for message copy number, which is the biggest advantage of Spray and Wait.

Considering the result in Fig. 3(c), it should be noted that NPSW also get a little higher than Epidemic, Prophet and Binary Spray&Wait in the respect of average latency. Our scheme NPSW needs to collect information relatively and quantize node performance for comparing, which greatly increases the average delay time. When messages' TTL gains bigger, all algorithms' latencies become larger.

In Figure 3(d), we can see that messages TTL influences the average hopcount of NPSW dramatically compared with other three schemes. That might be due to the conflict caused by wait phase. In this stage, the TTL is not short or long enough so as to messages may be cyclic transmitted. Therefore, the wave of our scheme average hop count is unstable.

Figure 3 investigates the performance comparison in terms of delivery ratio, overhead, average delay and hop count as to the message generated interval time. A phenomenon that can be observed from Fig. 3(a) is that the delivery ratio of NPSW is higher than BSW. This is because that NPSW adopts asymmetric spray mechanism. Though BSW spreads message copies quickly, it doesn't take the differences between nodes into account. The behind reason is that in the same simulation experiments, nodes produce less messages than before as time interval becomes bigger. The less the amount of resources is consumed to transmit information, the higher the delivery ratio is.

In addition, our algorithm seems to have a good behavior on overhead than that of Prophet and Epidemic in Fig. 4(b). This is due to the fact that Epidemic and Prophet replicate more message copies when routing the information. On the contrast, our scheme and BSW maintain the overhead in a steady level because NPSW and BSW control the number of copies effectually. Nevertheless, Subfigure 4(c) shows that NPSW achieves a larger average latency than other three protocols, especially after time interval increases to 30s. What else shown in Subfigure 4(d) is that NPSW jumps more count in order to transfer the information successfully. It is because of the complexity of our protocol, such

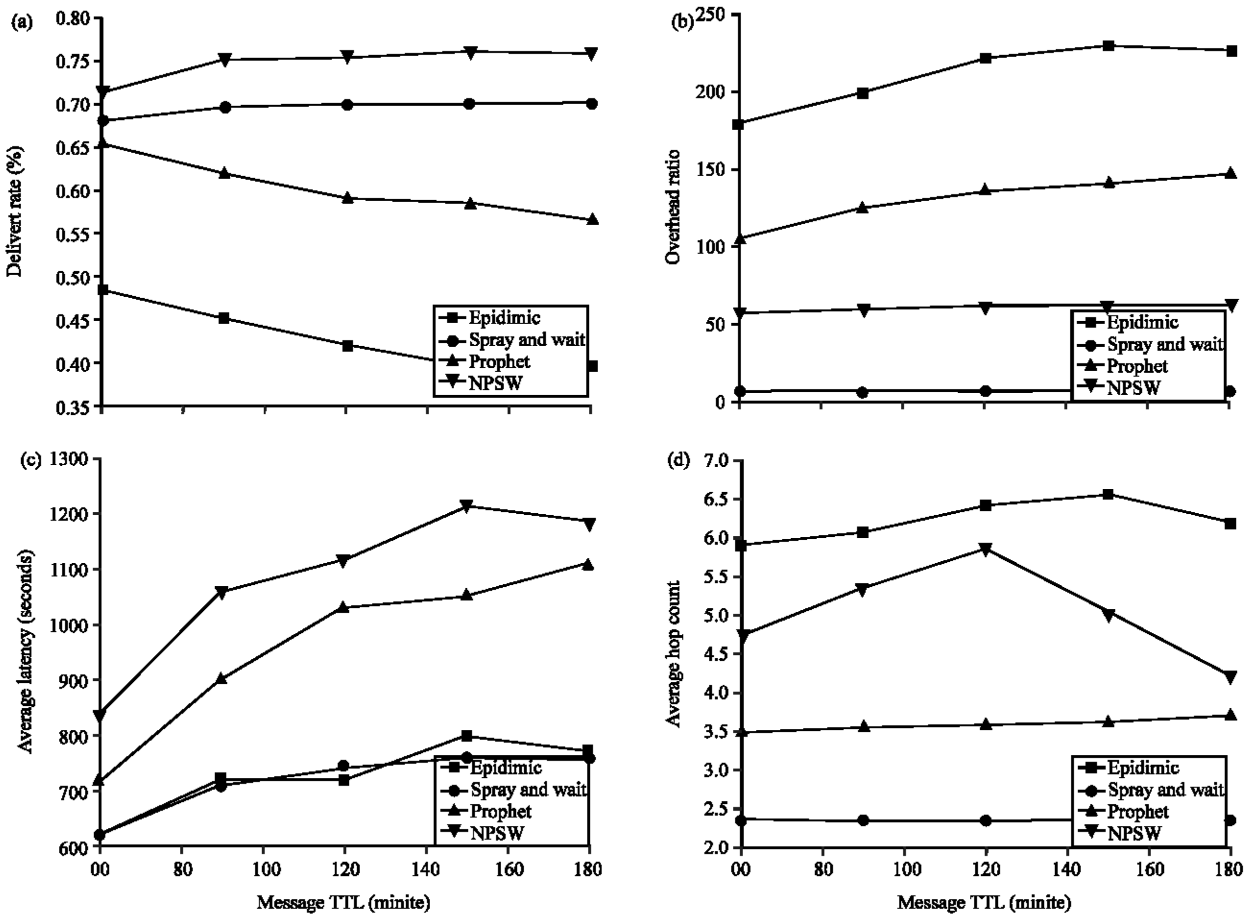

Fig. 3(a-d): TTL of messages vs delivery ratio, overhead, average latency, average hopcount(a) TTL vs delivery ratio (b)TTL vs overhead (c)TTL vs latency (d)TTL vs hopcount

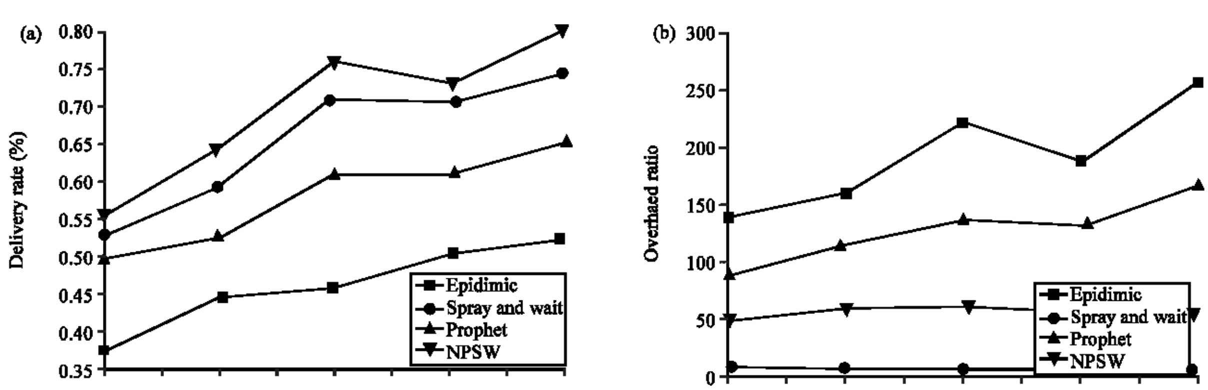

Fig. 4(a-d): Continue

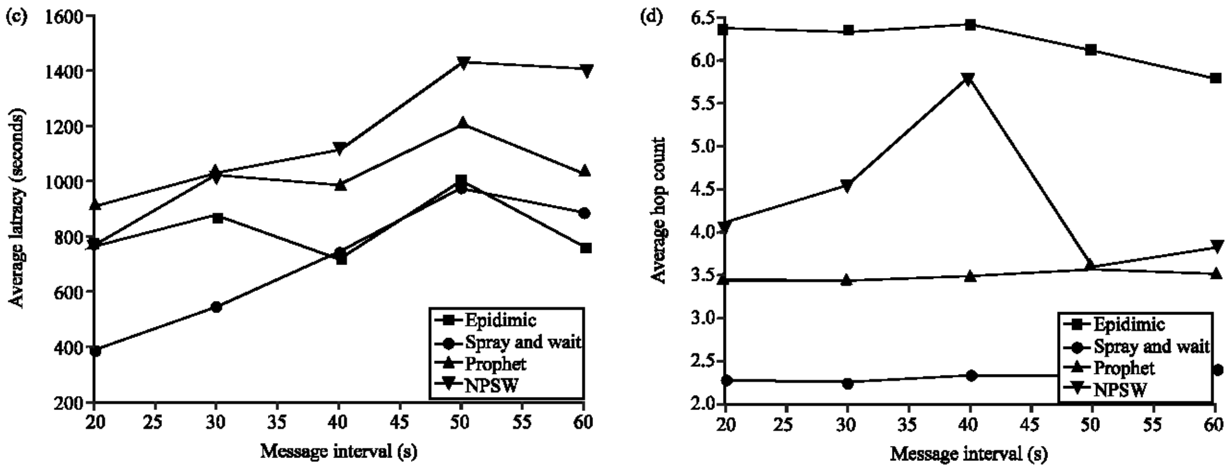

Fig. 4(a-d): Interval of messages vs delivery ratio, overhead, average latency, average hopcount (a) Interval vs delivery ratio (b) Interval vs overhead (c) interval vs latency (d) Interval vs hopcount

collecting information, calculation contact frequency, comparing performance metric and so on.

CONCLUSION

In this study, we proposed a novel improved Spray and Wait routing protocol based on node performance for Delay Tolerant Networks. In our NPSW scheme, we first define a performance metric v_{ki} for each node using the history encounter information to select a better candidate node as next hop. Moreover, our scheme design a dynamical spray method based on calculating the touching frequency with adjacent nodes. The simulation performed above shows that NPSW scheme achieves a better performance than Binary Spray and Wait in terms of delivery ratio. However, the average delay of NPSW is slightly higher than other state of the art algorithms because the additional selecting mechanism may need more time to choose candidate nodes. In the future work, we will find other methods to address the weakness of our scheme meanwhile with keeping high delivery ratio and low communication overhead.

ACKNOWLEDGMENTS

This research was supported in part by Foundation research project of Qingdao Science and Technology Plan under Grant No.12-1-4-2-(14)-jch and Natural Science Foundation of Shandong Province under Grant No.ZR2013FQ022.

REFERENCES

Bae, S.H., S.J. Lee, W. Su and M. Gerla, 2000. The design, implementation and performance evaluation of the on-demand multicast routing protocol in multihop wireless networks. IEEE Network, 14: 70-77.

Burgess, J., B. Gallagher, D. Jensen and B.N. Levine, 2006. Maxprop: Routing for vehicle-based disruption-tolerant networks. Proceedings of the International Conference on Computer Communications, April 23-29, 2006, Barcelona, Spain, pp: 1-11.

Burleigh, S., A. Hooke, L. Torgerson and K. Fall, 2003. Delay-tolerant networking an approach to interplanetary internet. IEEE Communi. Magazine, 41: 128-136.

Fall, K., 2003. A delay-tolerant network architecture for challenged internets. Proceedings of the Conference on Applications, Technologies, Architectures and Protocols for Computer Communications, August 25-29, 2003, Karlsruhe, Germany, pp: 27-34.

Keranen, A., J. Ott and T. Karkkainen, 2009. The ONE simulator for DTN protocol evaluation. Proceedings of the 2nd International Conference on Simulation Tools and Techniques, March 2-6, 2009, Rome, Italy, pp: 1-5.

Malladi, R. and D.P. Agrawal, 2002. Current and future applications of mobile and wireless networks. Communi. ACM, 45: 144-146.

Prodhan, A.T., R. Das, H. Kabir and G.C. Shoja, 2011. TTL based routing in opportunistic networks. J. Network Comput. Appl., 34: 1660-1670.

Xie, J., R.R. Talpade, A. Mcauley and M. Liu, 2002. AMRoute: Ad hoc multicast routing protocol. ACM/Baltzer Mobile Networks Appl., 7: 429-439.