Utility-based Distributed Routing in Intermittently Connected Networks

Ze Li and Haiying Shen
University of Arkansas
Department of Computer Science and Computer Engineering
Fayetteville, AR 72701
{zxl008, hshen }@uark.edu

Abstract

Intermittently connected mobile networks don't have a complete path from a source to a destination at most of the time. Such an environment can be found in very sparse mobile networks where nodes meet only occasionally or in wireless sensor networks where nodes always sleep to conserve energy. Current transmission approaches in such networks are primarily based on: multi-copy flooding scheme and single-copy forwarding scheme. However, they incur either high overheads due to excessive transmissions or long delay due to possible incorrect choices during forwarding. In this paper, we propose a A Utility-based Distributed routing algorithm with Multi-copies called UDM, in which a packet is initially replicated to a certain number of its neighbor nodes, which sequentially forward those packets to the destination node based on a probabilistic routing scheme. Some buffer management methods are also proposed to further improve its performance. Theoretical analyze and simulations show that compared to Epidemic routing, Spray and Wait routing, UDM routing scheme provides a nearly optimal delay performance with a stable packet arrive rate in the community mobility model.

1 Introduction

Over the past years, tremendous development of techniques such as IEEE 802.11 and low power radios has produced a significant stimulus to wireless ad hoc networks. In a wireless ad hoc network, packets can be forwarded by the intermediate nodes if a pair of source node and destination node can not communicate directly. One of the most basic requirement for the wireless ad hoc network is that a continuous end-to-end path between a pair of source and destination nodes always exits. However, in intermittently connected mobile networks, i.e. Delay Tolerant Networks (DTN), such continuous connections can not be guaranteed due to mobility [1], power management [2], wireless range, sparsity [3], or malicious attacks [4]. Examples of DTN include wildlife monitoring sensor networks [2], interplanetary communication networks [5], vehicular ad hoc networks (VANETs) [1], terrestrial wireless networks, ocean sensor networks [6, 7]. Therefore, conventional internet routing protocols (e.g. RIP, OSPF) as well as ad hoc network routing schemes, such as DSR [8] and AODV [9] that assume a complete source-destination path cannot be applied to DTN directly.

One group of routing schemes proposed for DTN is based on flooding [2, 10, 11]. Despite their increased robustness and low transmission delay, flooding-based routing schemes (i.e. epidemic routing schemes) consume much energy, bandwidth, and memory space that are crucial to the performance of wireless network applications. In particularly, under high traffic loads, they suffer from severe resource contention and packet drops, which significantly degrade their scalability performance. On the other hand, another group of routing schemes proposed for DTN is single-copy based routing, such as two hop direct routing [12], predicted routing [13]. In the two hop direct routing, the source node spreads the packets to several mobile nodes. These mobile nodes keep these packets until meet the destination node. In predicted routing, the packets are forwarded to mobile nodes that have higher probability to meet the destined node. Although these schemes bring about much lower overhead for packet transmission, which consequentially save a considerable amount of node resources, they are likely to suffer from severe transmission delay if a wrong path for the delivery are chosen .

In this paper, we present the design, implementation and evaluation of a utility-based distributed routing algorithm with Multi-copies (UDM) for intermittently connected network. Taking advantage of current connectivity information and predictions of future connectivity information, UDM "stores and forwards" the packet to the destined node in a distributed manner. The basic idea of UDM is that when a source node want to transmit a packet to a destination node, it initially replicates a packet to a certain number of its neighbor nodes. These nodes independently "store and forward" the packet copies to another node that has a higher utility (the possibility to meet the destination node) of the packet's destined node. This process will continue until one of the packet copies arrives at the destination. Fundamentally, the benefit of UDM is that it allows the transmission to be spread over multiple relays while using a constrained amount of overhead. It makes the transmission much more robust to relay failures or some bad choices caused by the single copy predicted routing, leading to a high transmission performance. Moreover, based on some buffer management methods such that replace low utility packets (back up packets) with other high utility packets (core packets), delete out-dated packets and give higher transmission priority to the delaying or emergent packets, UDM outperforms traditional flooding-based and single copy based routing schemes in respect to the delay, congestion avoidance, and packet receiving rate.

Spyropoulos *et al.* [10] also proposed a multi-copies routing method with constrained number of routing copies called Spray and Wait (SW). UDM differs SW mainly in two aspects: (1) UDM uses predicted routing after replication phrase while SW uses direct routing. (2)Several buffer management approaches are employed in UDM to deal with the packets congestion in the buffer while SW only uses a simple TTL scheme to management buffer.

Simulations results confirm that UDM improves the performance of SW in terms of the number of received packets and transmission delay. In the next section we go over existing related work. Section 3 presents UDM routing algorithm. Simulation results are presented in section 4. Finally, section 5 concludes the paper.

2 Related Work

Although numerous routing protocols for wireless ad hoc networks have been proposed [9, 14, 8], traditional routing protocols are not appropriate for DTNs that are sparse and disconnected. These protocols don't work well even if the network is only "slightly" disconnected [12].

An intuitive idea to deal with connectivity disruptions in DTNs is to reinforce connectivity on demand by sending out a number of specialized nodes (e.g. robots, satellites) which are assigned to fill the "communication gap" when a disconnection happens [15, 16]. However, this approach is not applicable in a highly dynamic self-organized networks needed to be "fixed" at all time.

Predicted routing is another approach for DTN [17, 3, 5, 3, 13]. They determine the routing path before transmission. In [2], nodes record the history of past encounters in order to make fewer but more informed decisions. Those routing paths are predicted either by statistics of a mobility module or by a historical moving path record. However, these schemes reduce the transmission overhead of flood-based routing at a significant penalty on delivery delay. In [13], the author points out that consulting the age of the last node encountered when making forwarding decision results in superior performance than flooding. [3] propose a forwarding algorithm to minimized the average delay of packet delivery using oracles about the current network topology. However, Balasubramanian in [18] argue that, even the simplest oracle in [3] is very hard to be implemented because the connection opportunities are affected by many factors in practice such as weather, radio interference and etc.

The third transmission approach for DTN is opportunistic routing. A simplest approach is direct routing that lets the source or a moving relay node carry the packet all the way to the destination [19]. Although these schemes can achieve high throughput performance, the delay will be very long, especially when base stations are sparse in the system. A faster way to perform routing in DTN is epidemic routing [11] which is based on packets flooding. This scheme can guarantee a optimal short delay with a infinite buffer by locating a shortest routing path at the cost of high network resource consumption. There are some improved approaches proposed to reduce the overhead of the epidemic routing [20–23]. In [20], a packet is "gossiped" to other nodes instead of flooding in which a packet is forwarded to partial neighbors. In [22], nodes remove redundant copies of certain packet when that packet has been transmitted by exchanging a "metadata" containing the ID of delivered node. Network coding [23, 21, 24] have also been used to improve the performance of the flood routing. In [24], the author use erasure coding technology to achieve the desired data delivery ratio with minimum overhead. He also implement a fault tolerance message which indicates the importance of the messages. The decisions on message transmission and dropping are made based on fault tolerance for minimizing transmission overhead. Chen et al. in [25] proposed a hybrid routing method which fully combine the robustness of erasure coding based routing techniques, while preserving the performance advantages of reputation techniques.

Although all these schemes can improve the performance of epidemic routing to a certain extent, they still inherits the shortcomings caused by flooding and can not significantly reduce transmission delay.

3 The UDM Protocol

In this section we describe the UDM protocol. We start off by describing the goals of UDM design and then discuss the approaches to achieve the goals. We will also discuss the various aspects of the protocol in details.

To be an optimized protocol for DTN, UDM has the following goals and corresponding approaches.

(1) In order to reduce the delay of existing single-copy schemes and make the transmissions more robust to relay failures or bad choices, UDM replicates a certain number of copies of a source packet to other nodes.

(2) In order to reduce the forwarding delay of traditional direct forwarding scheme, UDM adopts probabilistic routing to "guide" a packet forwarding in a corrective path. Even in the situation where nodes carrying packets fail to make right forwarding choices at all the time, UDM will degenerate to "Spray and Wait"scheme [10] which has been proved to outperform most current routing methods.

(3) In order to improve the transmission performance in a high loaded system, UDM uses buffer management methods to efficiently manage the buffer by replacing the low utility packets with other high utility packets, detecting out-dated packets and giving higher transmission priority to the delaying or emergent packets.

Since the replication of one unique packet does not affect the replication of another packet, all the transmissions can be regarded as states in a Markov chain. UDM routing spreads a number of copies generated per packet, therefore a number of transmissions can perform in the entire network. Specifically, UDM has three phases listed below:

(1) Replicate phase: For every packet originating at a source node, N_c packet copies are initially spread and replicated to N_c distinct closed relays. If the destination node is among these nodes, the transmission of this packet is completed. Otherwise, it goes to the forwarding phase.

(2) Forwarding phase: For every node in the system, it will hold a utility vector that include the meeting possibility value for every other node it has met. The utility vector indicates how likely this node will be able to deliver a packet to the other node. Based on these utility information, each copy of the packet is forwarded to the node with a higher utility value intentionally until one of those copies arrives at the destination node. If the buffer is full, the packets with higher utility value will replace the packets with low utility value.

(3) Clear phase: After the transmission is completed, the destination node sends a message or piggybacks it on other packets back to the system. This message includes the identifiers of the offloaded packets received in destination node. The nodes in the system will updated the information of offloaded packets and delete those packets in their buffers if they have them.

3.1 Mobility Models

The traditional popular simulation scenarios such as random walk, random way-point model [26] assume that each node may move equally frequently to every network location with identical, and independently distributed mobility process. However, numerous recent studies based on mobility traces from real networks (e.g. university campuses, conferences, etc.) have demonstrated that these two assumptions rarely hold in real-life situations [12]. For this reason, we simulated UDM under a more realistic mobility model, called "Community-based Mobility Model" that better resembles real node movement [27]. In reality, every people have their living habits and moving routines . That is, they usually go to some places with high probability but other places in low probability. Therefore, in the community mobility model, a node selects a destination and moves to it at a selected speed, and then repeats this process. If the node is at home community, it will go to a gathering place (e.g. in reality it can be a mall for the people, feeding ground for the animals, or certain bus stop) with a high possibility, but it can still go to other places. If it at a gathering place, it is very likely that the next destination of the node is a home community. Moreover, if the node is in other places, it will definitely go back to the home community.

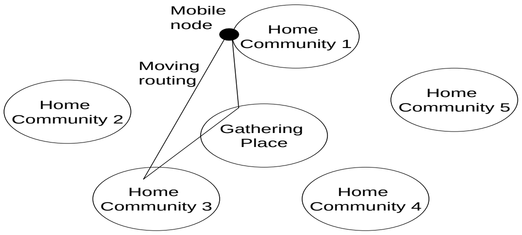


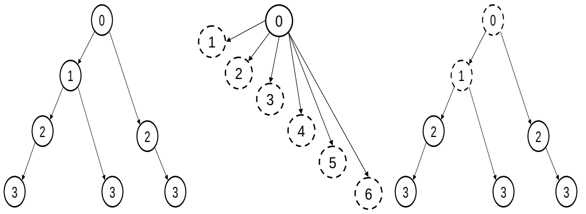
Figure 1. Community mobility model

3.2 Delivery Utility Calculation

There are serval factors that can be used to determine a node's transmission utility to others according to different systems based on different mobility models, such as time aging and transitivity utility using in [13] and the distance utility adopting in [12]. However, in DTN, the communication time between two nodes is limited. After one interaction, it will take a while to forward the packets to the destined node again. In this paper, we devise a new utility called meeting time possibility in the community model. The packets are always forwarded to the nodes with higher utility to the destined node. The calculation of the meeting possibilities has three steps. The first thing to do is to update the utility whenever a node is encountered, so that nodes frequently encountered have a high meeting predictability. The calculation is shown below.

$$P_{(i,j)} = \frac{T_{(i,j)}}{T_{(i)}},$$

where $P_{(i,j)}$ denotes the utility of node i meets node j, $T_{(i,j)}$ is the total meeting time between $node_{(i)}$ and $node_{(j)}$ in a time interval $T(i)$. $T(i)$ denotes a time period between $node_{(i)}$ leaving home community in consecutive two times. For example, the first time when $node_{(i)}$ leaves home com-



(a) (b) (c)

Figure 2. Packets replication tree

munity is at t_0, and the second time it leaves at t_1, then $T(i) = (t_0 - t_1)$. However, when calculating the current utility of the node, the old utility value of the node should be also brought into account, since the latest acquiring utility may not accurately reflect the meeting probability of two nodes because of the signal interference or link break. Therefore, every time when nodes leave home, they will recalculate $P_{(i,j)}$ as

$$P_{(i,j)_{current}} = \alpha P_{(i,j)} + (1 - \alpha) P_{(i,j)_{new}}$$

and then clear $T_{(i,j)}$ and T_i in order to record the next round meeting possibility $P_{(i,j)_{new}}$, where α is a weighing constant ($\alpha \in (0, 1)$). The packets forwarding are based on the value $P_{(i,j)_{current}}$.

3.3 Packet Replication

UDM uses replication strategy to increase the probability that a packet copy is offloaded to a destination node. Too much replicas will result in traffic congestion or unnecessary node energy consumption in the system. Therefore, UDM generates a small number of replicas to be transmitted in the system. The number of replicas should be chosen based on the optimized trade-off between energy consumption and robustness provision, influenced by system size, mobility model or number of nodes in the system [10]. The replicated packets are sent to randomly selected neighbor nodes.

Suppose that there are N_c packet copies in the system. These packet copies should be replicated as fast as possible as well as the replication process can be terminated when $N_c - 1$ copies are presented. Therefore, an approach is needed to ensure a node knows the number of replicated packets. A replicate distributing approach called locally-optimal tree [22] is adopted by UDM to replicate the copies. Figure 2(a) shows that at time $t = 0$, node n_1 knows there are N_c copies required in the system. Thus, when node n_1 comes across node n_2, it entitles n_2 to replicate $\frac{N_c-1}{2}$ copies to other nodes while itself remains $\frac{N_c-1}{2}$ transmissions. Sequentially, if $node_{(2)}$ meets $node_{(3)}$, each node will be entitled to have half of the remaining transmissions, i.e. $\frac{N_c-2}{4}$. The process is continuous until each node has only one copy of the packet. Figure 2(b) shows a source tree algorithm (only source node can replicate copy to other). Compared to the source tree which need $O(N)$ time steps to replicate the copies, optimal tree algorithm only needs $O(\log_2 N_c)$ time steps. Meanwhile, Figure 2(c) shows a binary routing tree algorithm (each node can only replicate copies to two other nodes). It is easy to find that the optimal tree algorithm can replicate the packet to another node with possibility of 1 at all time, while the binary tree algorithm only has a possibility of $5/7$ in this case. Although the binary routing tree algorithm also has a replication time step in the order of $O(\log_2 N_c)$, the replication process is still slower than the optimal tree algorithm.

3.4 Buffer Management

Buffer size has a critical impact on the performance of wireless sensor network. The limited buffer size will lead to serious traffic congestion, deteriorating network performance. In UDM, several buffer management methods are adopted to release the burden of the buffer and increase network throughput with reduced transmission delay.

3.4.1 Copy Management in the Buffer

In section 3.3, a packet is replicated for N_c nodes with different utility values to the destined node. However, since the replicated nodes are randomly chosen, some selected nodes may have low utilities to the destination node. In order to save the buffer resources, the buffer slots taking up by copies should be assigned to more promising packets, that is, the packets which are more liked to by transmitted to their destined node. Therefore, a utility threshold is assigned to the buffer to classify the packets into two categories: *core-copy* and backup-copy. If the utility of the packet's destination node is lager than the threshold, the packet is regarded as a *core-copy*, otherwise a *backup copy*. Pasztor [28] also proposed a classifying method that each kind of packets only have one master copy storing in the neighbor node with highest utilities until it reach the destination node. However, their method is not applicable here, since in UDM, the packets will be further forwarded after replication process. Therefore, which node can be the master copy can not be guaranteed in a long term.

When two nodes inquire each other's utilities for the packets forwarding, they also exchange the information of the number of the empty buffer slots and number of *backup-copy* in their buffers [28]. Packet copies forwarded to the other nodes firstly use available empty slots and then overwrite the slots of the *backup-copy*. Since *backup-copy* has a lower possibility to meet destination node in a short time, and the *core-copy* of this certain packet is still remain in a certian other node in the system, replacing this *backup-copy* in the buffer will not degrade the delay performance of transmission.

3.4.2 Copy deletion from the Buffer

Although the *backup-copy* and *core-copy* packet management can significantly release the burden of the buffer, when the buffer is full of *core-copy*, some promising packets from other nodes will be dropped. Therefore, we also propose a copy deleting approach to guarantee the robustness of the transmission.

When a node is about to load off packets to the destination node, the destination node sends a hash table containing the identifiers of the already received packets to that node at first. After that, the node deletes the packets indicated in the hash table if it has and forwards the remain packets to the destination node. After receiving the new packets, the destination node updates its hash table. This hash table periodically exchanges with this neighbor nodes or piggybacks them on other packets. The neighbor nodes that receive the hash table message will update their own hash table and get rid of the unnecessary packets.

3.4.3 Maintain the Forwarding Sequence

Since the transmission between two mobile nodes occurs only when they are within transmission range of each other, the time for packet transmission is especially in a highly dynamic situation. To ensure short transmission delay, a packet should be delivered to the destination node as soon as possible. However, a rely node may have several packets with various destinations at a time. In addition, when a rely node meets another relay node along its way, it can hand over very few packets, since the duration during which they are in each other's communication range is very small. Hence, UDM needs a scheme to decide which packets have higher priority to be transmitted during communication time.

In addition to the traditional field such as the IDs for source node and destination node, GSR includes two new fields into each packet's head : *priority* and *time_stamp*. *Priority* is used to indicate the delivery urgency of packets indicated by the applications. *time_stamp* is used to record the elapsed time since packet creation. In a node's buffer, the packets are arranged in decreasing order of *priority*. Within each level of priority, the packets are sorted in decreasing order of *time_stamp*. When two nodes meet each other, the bundles are delivered based on the sequence in the buffer. The employing of *time_stamp* guarantees that the longer a bundle stays in a buffer, the higher priority it has to be delivered. It avoids the worst delay in the communication in which a packet always stays in a buffer.

4 Analysis of the Performance of UDM

There are two kinds of routing methods for DTN currently, namely single-copy routing and multi-copies routing. However, although single-copy routing can reach a high throughput, it will lead to a much longer delay compared to multi-copies routing.

Theorem 4.1 *Signal-copy transmission can not achieve an average delay of less than O(M).*

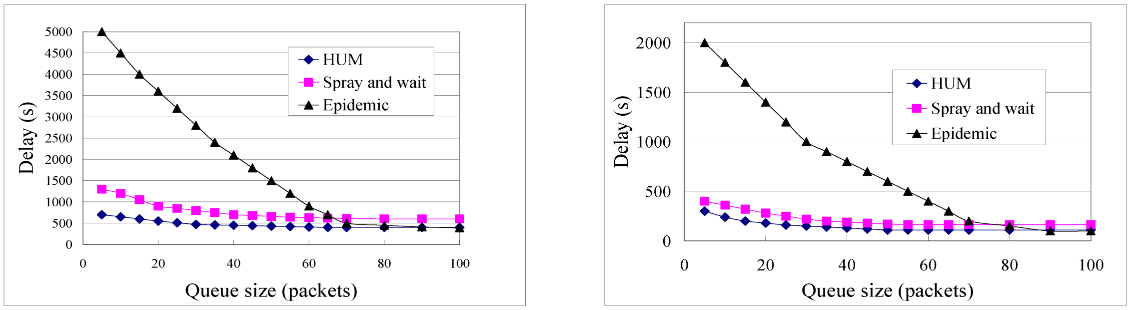
Proof Suppose a packet is transmitted in an optimal linearly path to the destination node. Therefore, $ET_d = L/v$, where ET_d denotes the transmission delay and L is the distance between a source node and a destination node; v is the average transmission speed. Because in any time, the packet is not duplicated and are held by at most one user, no further relay will happen in the transmission. Suppose $d = M/C$, where M denotes the number of nodes; C is the number of cells, therefore, d denotes the nodes' density in the networks. From the formula above, we can get that $L = O(C)$, $ET_d = O(L)$ and $C = O(M)$, therefore $ET_d = O(M)$.

Epidemic routing can reach an optimal delay performance in a lightly loaded transmission environment since a node can find a shortest path to the destined node by flooding the copies to all the nodes it meets before one of the copies is offloaded. However, flooding based transmissions consume significant energy and resources, which is precious to some micro-devices such as wireless sensor. T. Small [22] indicates that restriction of transmission traffic can save energy in the network. We give the proof below.

Theorem 4.2 *Given a large N_c, adding more copies will lead to constant energy consumption but negligible delay decrease.*

Proof Given the same amount of energy for each node. Suppose the average packet offloading possibility of each node is p and there are N_c copies of the packets in the DTN. The average energy consuming of each transmission is E. Therefore, to replicate N_c copies, $N_c - 1$ transmission are needed, consuming $(N_c - 1) \cdot E$ energy. If N_c were constant over the entire lifetime of the packets, offloading delay would be a geometric process with mean $\frac{1}{N_c \dot{p}}$. With another packet replication, the reduction of the packet offloading delay is $\frac{1}{N_c \dot{p}} - \frac{1}{(N_c+1)\dot{p}} = \frac{1}{(N_c+1)(N_c)\dot{p}}$. Therefore, with the increase of copies in the system, packet offloading possibility increases less and less while the energy consuming rate increases constantly at rate of E. ■

Therefore, in the UDM, a packet from a source node is replicated to a certain number of N_c ($N_c \ll M$) nodes to increase the possibility of a packet to be delivered to its destined node as well as reduce the overheads in the networks. Although, each of N_c relays looks for routing path independently in the forwarding phrase, delay in replication phrase will inevitably affects the delay performance in forwarding phrase and sequently deteriorate the whole transmission process.



(Range = 20m) (Range = 40m)

Figure 3. Transmission delay

Theorem 4.3 *Delay in the replication phrase will significant affect the delay performance in the forwarding phrase.*

Proof Suppose the average delay in the replication phrase is T time period. T. Spyropoulos [10] showed that the meeting time of randomly selected node i and j is exponentially distributed with average ET_d, and the expected duration of the forwarding phrase ET_f is $ET_f = \frac{ET_d}{N_c}$. Therefore, the average number of replications in each time slot is $\frac{ET_d}{(N_c/T)t}$ where $t \in [1, T]$. Then, the average delay is $ET_f = \sum_{t=1}^{T} \frac{ET_d}{(N_c/T)t}$. The average total delay in a packet transmission is approximately

$$T + ET_f = T + \sum_{t=1}^{T} \frac{ET_d}{(N_c/T) \times t}$$

$$= T + \sum_{t=1}^{T} \frac{ET_d \times T}{N_c \times t}$$

$$= \frac{ET_d \times T}{N_c} O(lnt)$$

The equation shows that as the delay in the replication prase T increase, the total transmission delay will increase in the order of O(ln t). ∎

That is also why in the replication phase, the source node in UDM would like to replicate the packet to its neighbor nodes as soon as possible regardless their utilities, rather than only replicate the packet to the node with higher utility as the forwarding phase do, because it is very likely that the node initially with low utility to the destined node will meet a intermediate node with high utility node later.

Theorem 4.4 *In the UDM routing method leads to delay performance* $O(\sqrt{N}) > ET_d > O(log(N))$

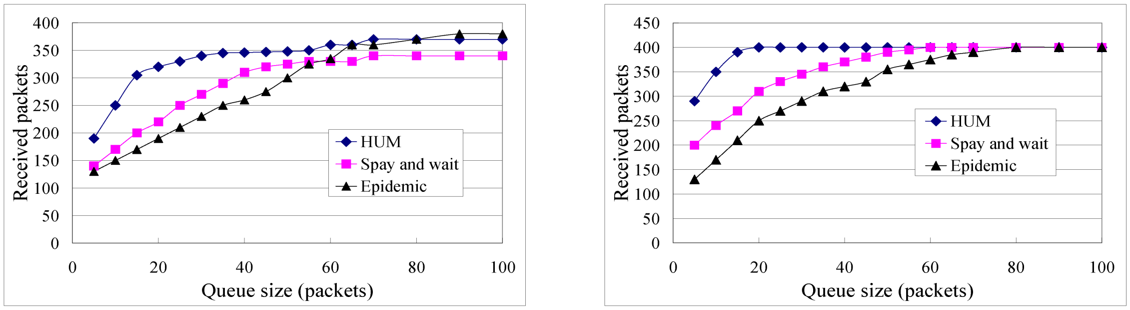
Proof In [29], Neely proved that no algorithm (with or without redundancy) that restricts packets to 2-hop paths can provide an average delay better than $O(\sqrt{N})$. Therefore, in the worst situation of the forwarding phrase in the UDM where nodes with copies can not find a higher utility node for the relaying, UDM will deteriorate to a two-hop multi-copies routing, the delay of which is in the order of $O(\sqrt{N})$. On the other hand, if packets in UDM can always find a better relay node (with a higher utility) to , the delay performance of the UDM is like an optimal redundancy multi-hop transmission [29], whose delay performance is O(log(N)) ∎

5 Performance Evaluation

This section demonstrates the distinguishing properties of UDM through simulation built on a custom discrete event-driven simulator [12] in comparison with Epidemic routing [11], and Spray and Wait routing [10]. We used "Community Model" [13] as the simulation scenario. This scenario consists of a $500 \times 500m$ area where 100 nodes are identical, independent distributed placed. Every 10 nodes share a home community and a gathering place. The home communities and gathering places are identical distributed in the area. The mobile nodes move in the scenario at speeds of $(0-20m/s)$, with pause time $0-5s$. Nodes are randomly chose to generate a new packet for a randomly selected destination with transmission rate 1 packet per node for 2000s. The hop count of Epidemic routing is 5 hops. The number of replicas of UDM and Spray and Wait routing is 16. When a node is in a home community, it will go to a gathering place with possibility of 0.8, and go to other randomly choose places with possibility of 0.2. When a node is in a gathering place, it will then go home with possibility of 0.5, other places with possibility of 0.5. When node is at other places, it will go back to home community directly. All results were average over 5 runs. A warm up period of $500s$ is used in the beginning of the simulations to initialize the utility of UDM.

We used two metrics in the simulation:

(1) Packet delivery delay: the average time that it takes a



(Range = 20m) (Range = 40m)

Figure 4. Number of Received Packets

packet to be delivered.

(2) Packet delivery ability: the number of the packets that is able to be delivered to the destination.

5.1 Packet Delivery Delay

Figure 3 (a) and figure 3 (b) present the packet delivery delay with different transmission ranges 20m and 40m. Comparing these two figures, we can notice that the transmission delay rapidly decrease with the increase of the transmitting range of the node. The reason is that a larger transmission range makes it easier to find other neighbor nodes, which may be the potential receivers or promising relay nodes. Moreover, the speed of the electromagnetic wave moves much faster than the moving node, thus data transmission with larger transmission range is faster. The figures also show that with the increase of the buffer size, transmission delay is decreased. It is because a larger buffer size enables more packets to be buffered, thus packets have low possibility to be thrown away.

The figures also indicate UDM has the best delay performance with a small buffer size. Although in UDM routing scheme, a packet is replicated to several nodes as Spray and Wait scheme does, some buffer slots taking up by the *backup-copy* can be replaced by other promising packets if the buffer is full, and the *core-copy* with the highest possibility to be delivered still remain in the system. With the buffer management approaches, the transmission in the UDM will not be significantly affected by the buffer size. However, Relying on flooding, the Epidemic routing suffer from severe congestion because of the limited buffer size.

Moveover, figure 3 shows UDM outperforms Spray and Wait routing where the nodes are equipped with large buffer size. It is due to the reason that in the forwarding phase, UDM uses a probabilistic routing while Spray and Wait just adopts direct routing leading to a little more delay as $O(\frac{N}{N_c})$.

The factor that delay of the UDM is almost the same as Epidemic routing with a large buffer size can also be found in the figure, which means UDM can reach the lower bound of the delay performance[11]

5.2 Packet Delivery Ability

The figure shows that the UDM is less sensitive to the buffer size changes than Epidemic routing for its significant buffer management approaches, and the Epidemic routing with flooding nature is still manifest to suffer from the congestion severely especially under high load. Figure 4 also indicates in a low load, where the butter is large enough for all the packets, the transmission performance of UDM can still reach the upper bound as Epidemic routing does.

Figure 4 also shows that as the buffer size increases, so does the number of the packets delivered to their destination. A larger buffer size means more packets can be buffered, and the possibility that a packet is thrown away decreases. Therefore, the number of packets received by the destination nodes will increase.

Since a shorter transmission range of a node leads to a smaller possibility of nodes to meet neighbor nodes, that is why the number of received packets are reduced with the decrease of transmission range. Meanwhile, in the Spray and Wait routing scheme, direct transmission with a Time To Live (TTL) threshold is adopted in the forwarding phase. However, it is very likely that the packets are dropped when TTL expires. That is why it is so obvious in the figures that Spray and Wait suffers more than other routing schemes from the decrease of the transmission range.

6 Conclusions

Traditional routing scheme in wireless ad-hoc networks can not achieve a good performance in Delay tolerate networks (DTN), since DTN can not guarantee a end-to-end link established all the time. Epidemic routing and Spray and Wait routing are two representative routing schemes for DTN. The former is based on flooding routing and the later is a hybrid routing combining direct routing and multi-copies routing. In this paper, we proposed a A utility-based distributed routing algorithm using multi-copies for DTN, namely UDM. UDM replicates a new packet to a certain number of nodes. These nodes hold the copies until they meet another node with a higher utility for the packet's destination. The packets is forwarded in this way until one of the copies reaches the destination. Simulation results based on community mobility model show that UDM outperforms the Epidemic routing scheme and Spray and Wait scheme in terms of packet delivery delay and packet delivery ability. In the future work we intend to implement UDM to other mobility modules such as random walk model and random way point model to see whether UDM can still keep high performance.

References

[1] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter. MDDV: Mobility-centric data dissemination algorithm for vehicular networks. In *Proc. of ACM on VANET*, 2004.

[2] P. Juang, H. Oki, M. Martonosi Y. Wang, L. S. Peh, and D. Rubenstein. Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with zebranet. In *Proc. of ASPLOS*, 2002.

[3] S. Jain, K. Fall, and R. Patra. Routing in a delay tolerant network. In *Proc. of ACM SIGCOMM*, 2004.

[4] Delay tolerant networking research group. http://www.denrg.org.

[5] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, B. Durst V. Cerf, and K. Scott. Delay-tolerant networking: an approach to interplantary internet. *IEEE communication magzine*, 2003.

[6] J. Partan, J. Kurose, and B. N. Levine. A survey of practical issues in underwater networks. In *Proc. of WUWNet*, 2006.

[7] A. Maffei, K. Fall, and D. Chayes. Ocean instrument internet. In *Proc. of AGU*, 2006.

[8] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. *IEEE Mobile Computing*, 1996.

[9] C. Perkins, E. Belding-Royer, and S. Das. RFC 3561: Ad hoc on demand distance vector (AODV) routing, 2003.

[10] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In *Proc. of WDTN*, 2005.

[11] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University.

[12] T. Spyropoulos, K. Psounis, and C. Raghavendra. Efficient routing in in termittently connected mobile networks: The single-copy case. *ACM/IEEE Transactions on Networking*, 2007.

[13] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mobile Computing and Communication Review*, 2003.

[14] E. P. Charles and P. Bhagwat. Highly dynamic destination sequenced distance vector routing (DSDV) for mobile computers. In *Proc. of SIGCOMM*, 1994.

[15] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *Proc. of Mobihoc*, 2004.

[16] Q. Li and D. Rus. Communication in disconnected ad hoc networks using message relay. *ACM/IEEE Transactions on Networking*, 2003.

[17] E. P. C. Jones, L. Li, and P. A. S. Ward. Practical routing in delay-tolerent networks. In *Proc. of SIGCOMM*, 2005.

[18] B. N. Levine A. Balasubramanian and A. Venkataramani. Dtn routing as a resource allocation problem. In *Proc. of SiGCOMM*, 2007.

[19] H. Shen and Z. Li. MDR: Market-based distributed routing scheme in hybrid wireless networks. Technical Report TR-2007-07, CSCE Department, University of Arkansas, 2007.

[20] X. Zhang, G. Neglia, J. Kurose, and D. Towsley. Performance modeling of epidemic routing. In *Proc. of IFIP*, 2006.

[21] J. Widmer and J. Y. L. Boudec. Network coding for efficient communication in extreme networks. In *Proc. of WDTN*, 2005.

[22] T. Small and Z. Haas. Resource and performence tradeoffs in delay-tolerant wireless networks. In *Proc. of WDTN*, 2005.

[23] Y. Wang, S. Jain, M. Martonosi, and K. Fall. Erasure-coding based routing for opportunistic networks. In *Proc. of SIGCOMM*, 2005.

[24] Yu Wang and Hongyi Wu. Delay/fault-tolerant mobile sensor network (dft-msn): A new paradigm for pervasive information gathering. *IEEE Transactions on mobile computing*, 2006.

[25] L. J. Chen, C. H. Yu, T. SUN, Y. C. Chen, and H. H. Chu. A hybrid routing approach for opportunistic network. In *Proc. of SiGCOMM Workshop*, 2006.

[26] N. Bansal and Z. Liu. Capacity, delay and mobility in wireless ad-hoc networks. In *Proc. of INFOCOM*, 2003.

[27] M. McNett and G. M. Voelker. Acm mobile computing and communication review. In *Access and mobility of wireless pda users*, 2007.

[28] B. Pasztor, M. Musolesi, and C. Mascolo. Proc. of mass. In *Opportunistic mobile sensor data collection with Scar*, 2007.

[29] M. J. Neely and E. Modiano. Proc. of ciss. In *Improving Delay in Ad-hoc Mobile Networks Via Redundant Packet Transfers*, 2003.