Backup Routing for Multimedia Transmissions over Mesh Networks

Chungui Liu1,2, Yantai Shu1 and Lianfang Zhang1
1. Department of Computer Science, Tianjin University, Tianjin, 300072, China,
2. Department of Computer Science, Datong University, Datong, 037006, China

Maode Ma
School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Avenue, Singapore, 639798

Abstract—A significant issue in Mesh networks is to support multimedia transmissions while providing Quality of Service (QoS) guarantees to mobile users. For real-time multimedia streaming, unstable throughput or insufficient bandwidth will incur unexpected delay or jitter, and it remains difficult to provide comprehensive service guarantees for wireless mesh environment. In this paper, we target the problem of providing multimedia QoS in wireless mesh networks. We have designed and implemented a campus wireless mesh network testbed, and propose an available bandwidth estimation algorithm plus a QoS backup route mechanism to accommodate multimedia traffic flows in mobile wireless mesh networks. Moreover, to validate the correctness of our proposed algorithm, we have implemented the algorithm on the campus wireless mesh network testbed. Our implementation and experiments show that our mechanisms can improve the network stability, throughput, and delivery ratio effectively, while decreasing the number of route failures.

Keywords- Multimedia; mesh networks; QoS; dynamic source routing (DSR); backup source routing (BSR)

I. INTRODUCTION

A wireless mesh network is a mesh network created through the connection of wireless access points installed at each network user’s locale. Each node is also a provider, forwarding data to the next one. Wireless mesh network could allow people living in remote areas and small businesses operating in rural neighborhoods to connect their networks together for affordable Internet connections. However, real-time multimedia traffic such as voice and video typically demand high data rate and more stringent service requirements, whereas wireless nodes generally cannot provide service guarantees due to the fluctuating nature of the wireless channel. It is complex and difficult to provide QoS routing in a wireless mesh network (due to the unsteadiness of wireless links) and a fixed network path from the source to the destination is often not a good choice for multimedia streaming. The goal of this paper is to design and verify the effectiveness of wireless protocols in supporting delay critical multimedia traffic on an actual testbed environment.

In this paper, we propose a path bandwidth estimation algorithm. In order to decrease the computational complexity of the estimation algorithm, we linearize the intra-flow and inter-flow interference and calculate the path bandwidth based on each node’s available bandwidth estimation. Moreover, we propose a QoS backup route mechanism for multimedia transmission. Combining these two mechanisms together, effective multimedia streaming over wireless mesh networks can be enabled.

The majority research in wireless networks has been conducted using simulations which offer an efficient and flexible means to evaluate new protocols. However, in simulations, MAC protocol and channel models are often simplified, ideal wireless channels are assumed without consideration of background noise, random interference and the usual utilization of various unrealistic traffic traces. Consequently, the performance evaluation through simulation may not reflect the fact in real networks [1]. As a result of simulation inaccuracies, many researchers have begun deploying actual multi-hop mesh network testbed for wireless network protocol development and testing. A testbed experiment is challenging due to the effort required to install, configure and manage the hardware [2]. Furthermore, the highly varying characteristics of wireless links can often lead to unstable and unrepeatable results. Significant effort is necessary to enable repeatable tests and to establish adequate methods for collecting and analyzing the testbed data. Yet, testbed results remain more meaningful than simulation evaluations.

Several wireless mesh network testbeds have been built by various research labs: the MIT Roofnet [3], the APE testbed in Uppsala University [4], and the testbed in Microsoft Research [5], and the UCSB MeshNet [6]. In this paper, we present the design a mesh network testbed environment (TJU MeshNet testbed) around our campus. We implement our proposed algorithms on the testbed through an improved DSR protocol. Our implementation and experiments show that our mechanisms can effectively improve the network stability, throughput, delivery ratio, while decreasing the route invalidation ratio, and can guarantee the fluent transmission of multimedia streams.

The rest of the paper is organized as follows. Section II briefly reviews related work. Section III introduces our wireless mesh network and testbed environment. Our proposed available bandwidth estimation algorithm and the QoS backup route mechanism were described in section IV and section V. Section VI analyzes our experimental results. Concluding remarks were presented in section VII.

II. RELATED WORK

Dynamic Source Routing (DSR) [7] is an “on-demand” routing protocol for wireless networks. It uses connection
oriented source routing instead of relying on the routing table at each intermediate device. The DSR is suitable for audio/video communications over the wireless channels. However, in the event of fatal route, the route rediscovery will bring long time delay and high data loss rate which will block the multimedia communications. Most of the reactive on-demand routing protocols use single path for routing.

Multipath routing is one of the favorite mechanisms to balance network traffic and to provide fault tolerance and quality of service (QoS). Without the use of routing tables to keep track of routes, mobility management and scalable design can be relatively easy to manage. A new idea in [8] is to generate two node-disjoint paths during the query phase in the route discovery process by restricting the way a query packet is flooded and to keep two node-disjoint routes in the route cache. The approach can reduce the frequency of invoking the routing discovery process. The DSR protocol [9] also has an option of maintaining multiple routes so that an alternate path can be used upon failure of the primary one. In the DSR, the quality of routes can be easily evaluated and the best (i.e., the shortest) one can be selected. But too many routes are maintained in a trivial manner without any regard to their ultimate usage. In [10], a multipath routing was proposed by constructing node-disjoint paths. The destination keeps a record of the first arrived packet. The subsequent packets will be discarded until a packet with a node-disjoint path with respect to the first one arrives. This scheme can implicitly control the total number of replies preventing a reply flood. [11] proposed a scheme to improve existing on-demand routing protocols by creating a mesh and providing multiple alternate routes without yielding any extra overhead. [12] proposed a novel on-demand routing protocol, named as Backup Source Routing (BSR), to establish and maintain backup routes that can be utilized after the primary path fails. The idea is to use the route reliability to provide the basis for the backup path selection and to use a heuristic cost function to develop an analytical model for an approximation. However, all the related research does not consider supporting of multimedia transmissions over wireless mesh networks. Furthermore, these mechanisms are usually verified by simulations only without performance measurements on corresponding testbed. Furthermore, [13] proposed an interference-aware QoS multipath routing protocol for QoS-constraint multimedia or real-time applications in ad hoc wireless networks. With available bandwidth and interfering susceptibility pre-evaluation, it can significantly reduce the call dropping rate and improve the QoS stability. However, the proposed mechanism lacks corresponding tested experiment results because the scheme only evaluates the stability and throughput improvement via simulations.

III. WIRELESS MESH NETWORK TESTBED

In this section, we introduce the details of our wireless mesh network and the testbed environment.

A. TJU Wireless Mesh Network

A wireless mesh network on our campus square has been designed, which is composed of two access points, three mesh routers and eight wireless nodes. In our current implementation, the routing protocol is mainly based on the DSR. Yet, the DSR scheme doesn’t provide Internet connectivity to all of the nodes by the design. There is no gateway concept in the original DSR protocol. We thus extend the function of the original DSR with the concept and implementation of mobile gateways and mobile gateway protocol to connect a MANET to Internet. Additionally, we also introduce a wireless LAN management protocol to facilitate the automatic management of our wireless multimedia system [14]. We implement all functions that are necessary for Internet access in our test-bed, specifically, the support of the DSR and the NAT at the gateway nodes, and a gateway discovery protocol by non-gateway nodes in MANET. We conducted extensive experiments to study the performance of using the mobile gateways and we found that it provides very good performances for majority of applications including supporting a fluent transmission of video streams. Figure 1 shows the topology of the network.

B. Testbed Setup and Experiments

Base on the TJU wireless mesh network, we have setup the TJU MeshNet testbed. In order to support multimedia transmission with QoS requirements, we improve the wireless routing protocol on the testbed with a dynamic ACK mechanism, which is used to balance the throughput and the quality of transmission [15]. Additionally, we introduce a dynamic mechanism to change the multimedia coding rate dynamically at the source node according to the available bandwidth [16]. Moreover, we also made improvement on the admission control protocol to facilitate our experiments.

We implement our QoS backup routing mechanism in the testbed environment. Figure 2 shows the topology of our experiment scenario. There are six mobile nodes in our experiments allocated within a 100m × 150m square without any obstacles. There is a nummular fountain in the square center, and it is used as a reference point for relay nodes’ motion. The nodes are laptops such as DELL 1150N (2.8GHz Intel Pentium 4 processor with 512MB of memory). Each node has a 802.11b adapter (D-link dwl-650) with omnidirectional antenna. All adapters are set to DCF mode, 5.5Mbps data rate, All nodes run on Linux Red Hat 9.0 operating system with original DSR and improved MSR testbed modules.

In the experiment scenario, there are two video streams: node 7 transmits video packets to node 3 at 150kbps via node 5 (Stream A), and node 4 to 6 at 300kbps via node 9 (Stream B). The formats of both video streams are 320 × 240, 10-15 frames per second.

![Figure 1. TJU Wireless Mesh Network](image-url)
that share this common channel. Notice that this available bandwidth will be shared by this node and all its neighboring nodes as shown in (1).

Notice that the above estimation has taken into account the inter-path interference. The intra-path interference will be taken care of as follows. Assume that there is a path \( P = v_1, v_2, \ldots, v_k \), where \( v_i (1 \leq i \leq k) \) is a node on the path \( P \). Let \( b_i \) be the estimated available bandwidth of node \( v_i \).

\[
\begin{align*}
b_i &= (1-\alpha) \times \rho \times C - \sum_{j=1}^{m} \left( \sum_{t=j+1}^{T} L_t / T \right) \\
\rho &= E[S] / (t_f - t_i)
\end{align*}
\]  

where we set a variable \( L_i \) for each communication node to periodically (generally, 1 second) record the number of bytes sent by the node. \( m \) is the number of node and all its neighboring nodes that share this common channel. \( E[S] \) is the average transmission time of one MAC data packet. \( (t_f - t_i) \) is the transmission period of the channel. \( C \) is the data rate of the channel. \( \rho \) is the utilization ratio of the channel (about 0.68). \( \alpha \) is the interference coefficient and equals to 0.2 according to our test bed experiments [18].

When the path has at most 3 hops, the path bandwidth is estimated as follows:

\[
B(P) = \min \{ b_i \mid 1 \leq i \leq k \} / h(P)
\]  

The reason is that all links on this path will interfere with each other. \( h(P) \) is the hop-count of path \( P \). When the path has more than 3 hops, the path bandwidth is estimated as follows:

\[
B(P) = \min \{ b_i \mid 1 \leq i \leq k \} / 4
\]  

This is due to the fact that, from our experiments, a link will interfere with only at most 4 links. This estimation simplifies channel sharing under the complicated inter-path and intra-path interference.

To implement the above algorithm, a node has to have the knowledge of the estimated occupied bandwidth of all its neighboring nodes. In our design, similar to AODV, we extend the Hello packet in the DSR routing protocol. We introduce the Neighbor State Exchange and Update (NSEU) mechanism [16] into DSR protocol. The difference is that the Hello packets in the AODV routing protocol are used to create the routing table, while we use the Hello packets to collect the occupied bandwidth of neighboring nodes without affecting the DSR routing protocol. Each node will periodically broadcast a Hello packet to announce its occupied bandwidth. Then its neighboring nodes can estimate their available bandwidth based on the information kept in these packets. We point out that this Hello packet will not be flooded to the network. But it is only sent to its one-hop neighbors.

In our testbed implementation, when a source node requests a new multimedia transmission, it first calculates the required bandwidth for this flow and the available bandwidth of itself. If available bandwidth is large enough, it sends out a route request packet, which contains the bandwidth requirement of the flow and its own available bandwidth as current path bandwidth. When an intermediate node receives this request, it first retrieves the path bandwidth information up-to this node and then estimates its available bandwidth. It will discard the request if its available
bandwidth is less than the requested flow bandwidth. Otherwise, it updates the supportable path bandwidth and forwards the request. When the target node received the request, it computes the final path bandwidth of a path $P$ as follows:

$$AB(P) = \frac{B(P)}{\min(4, h(P))}$$  \hspace{1cm} (5)$$

where $AB(P)$ is the available bandwidth of path $P$ and $B(P)$ is the minimum node bandwidth of path $P$. It then chooses the path with the maximum path bandwidth and informs the source node of the selected path.

V. USING BSR TO SUPPORT MULTIMEDIA

A key advantage of using a backup path is the reduction of the frequency of route discovery flooding, which is recognized as a major overhead in on-demand protocols, particularly, in mobile wireless networks where routes are disconnected frequently due to mobility and poor wireless link quality. In this section, we will study the performance using BSR compared with routing using primary path only. The BSR algorithm, which extends DSR by selecting a backup path piggybacked with the primary path to achieve the more reliable routes among communicating mobile nodes. Through performance evaluation via both simulations and testbed experiments, we show that our routing strategy has a couple of advantages: 1) in the less stressful situations with lower mobility, BSR has similar performance as DSR, and 2) in more challenging situations with higher mobility, BSR can improve the performance significantly.

A. Mechanism

BSR is an extension of DSR, which uses the concept of backup route to improve the route reliability. BSR consists of two modes of operation: (a) Route Discovery and (b) Route Maintenance. Route Discovery is only invoked when needed while Route Maintenance operates only when the route is used actively for data transmission.

Route Discovery: In ad hoc wireless networks, a destination node will calculate the available bandwidth of the path when it receives a Route Request from a source node via intermediate nodes. If the path’s available bandwidth is larger than the requested one, the destination node will send a Route Reply back to the source node via the chosen routes. The source node may receive multiple Route Reply packets corresponding to multiple available paths. And then, it selects the primary path and the backup path in routing cache. When the source node receives the first Route Reply packet, a timer is set. All Route Reply packets arrived within timeout period are accepted and the routes contained are sorted according to their available bandwidths. The one with the maximum available bandwidth is selected as the primary path and the second most one is used as the backup path. Timeout value of the timer (ROUTE_PENDING_TIME) should be chosen appropriately. If it is too short, some Route Reply packets may not be returned. On the other hand, the delay of multimedia traffic will be affected. Some Route Reply information that has already been returned may become invalid. We set ROUTE_PENDING_TIME to 500ms after many outdoor experiments. So far, except available bandwidth, there are not any other feasible schemes to judge the quality of paths in real experiments. So the only metric we use in selecting routes is the value of available bandwidth in our implementation.

If, after Route Discovery period, a source node just finds one primary path, with no backup path, it will initiate Route Request periodically to look for a backup route. To avoid the flooding effect of Route Request and to lower overhead of transmitting DSR control packets, the period we select must be larger than DSR_REQ_TIMEOUT. The experiments’ result shows that it is better to be set as 4 seconds. Therefore, in our improved DSR protocol, the period of issuing Route Request is 4 seconds.

Route Maintenance: In BSR, when a node fails to deliver data packets to next hop of the route, it considers the link to be disconnected and sends a ROUTE_ERROR (RERR) packet to the stream node of the route that is still using the reverse backup routes. The idea is to reduce the probability of losing RERR packets. A RERR packet contains the backup routes to the stream node, and the immediate upstream and downstream nodes of the broken link. Upon receiving these RERR packets, the source node will remove the broken link in its cache and reconstruct its backup routes using the updated routing information. If there is not enough information, it will initiate a new route recovery process.

B. Primary/Backup Routes Switching

Generally, the multimedia traffic flows are served via the primary path. But if the primary path fails detected by the ACK mechanism of the DSR protocol and an available backup path has been already found, the backup path will be activated and transmit the rest data packets. At the same time, DSR begins to search for a new backup path.

Moreover, in order to avoid the route fluctuating that is brought by frequent switching, the primary path will never be changed when it works well, even if its bandwidth value is less than the new backup path’s. Figure 3 shows the switch process of primary/backup path.

\[\text{Figure 3. primary/backup path state switch diagram}\]

VI. PERFORMANCE EVALUATION

Our evaluation is based on our implementation of BSR protocol with QoS support. In order to validate the usability of backup routing mechanism and compare the performance between improved DSR and original DSR with considering the battery power at the laptops, we tested the scenario with the original DSR for only 13 minutes and with the improved DSR for 50 minutes. The results are shown in Figure 4 and 5. The details of Y-axis are shown in Table 1 and 2.
Figure 4 shows the details of the route transformation from node 7 to node 3 with the original DSR, and the 5, 6, 7, 8, 9 points on Y-axis represent all 3-hop routes. Confessedly, the more hops a route has, the more probability of route invalidation there are. Figure 4 also shows that the time of successful data transmission via 3-hop relay is short. Due to the lack of backup routing mechanism, it will take the system a lot of time to rediscover a new available route when the old route failed. At this phase, the video images that collected by node 7 cannot be transmitted to the destination node. From the results, we found that the routing reply is often not received by the requesting nodes in 3 seconds. For real-time video communications, it is not acceptable.

At the initialization state, the route 730 was chosen as the primary path because the available bandwidth of single-hop route is the most. According to the principle of backup route selection, the route 763 was chosen as the backup path. The reply of route 753 was not the first one among all reply that node 7 received. It is because the signal of wireless adapter is gradual attenuation and it does not have strict transmission range in real environment. In usual cases, discovering a single-hop relay route will take more time than two-hop relay route discovery and the single-hop reply will be received earlier than two-hop reply because the two-hop route request and reply packets must be transferred by relay nodes. And, the return sequence of two-hop route reply is stochastic. However, it can be made sure that the two-hop routing path is steady because both the source and destination node are near to the relay node.

In Figure 2, the single-hop path (730), only activated for about 9 seconds before it was broken because the distances among all nodes were too long and the transmission signal was weak. Since there was a backup path (763) in the route cache of node 7, the path switched to the backup one when the primary path broken. At this phase, the route would not be rediscovered, and the lost packets which were sent by the fault route would not lose forever too. When data transmitted for 90 seconds, the route between node 7 and 6 was broken because node 7 was far away from node 6 and the quality of signal was poor. Because of using the backup routing mechanism, node 7 had found a backup path (753) to node 3 before the route between node 7 and node 6 broken. As the high quality of route 753, node 7 used this route from 9:59:43 to 10:10:31.

We also evaluated the performance of improved DSR protocol in mobile scenarios. We recorded the changes of routes and quality of the videos during the movement. Node 5 moved towards to node 7. This action broke the route between node 5 and 3 and the route switched to the backup one (730). The route changed to route 743 and 763 after a short period because of the long distance and the poor signal quality. Due to the long distance between either node 4 and node 3 or node 6 and node 7, the route via node 4 and node 6 was unstable and changed to route 793. The route changed to route 753 when node 5 moved back to its original position. From Figure 5, we found that the movement of node 5 did not interrupt the video communications from node 7 to 3 and this process could prove the validity of the backup routing mechanism. Then node 5 moved towards to west for 500 seconds. When node 5 went out of the transmission range of node 7 and 3 at 10:14:23, the route changed to route 793. After that, the video delivered via route 793.

From Figure 5 and Figure 2, we found that the length of time a route could be used is decided by the stability of the route. Due to the long distance between either node 4 and node 3 or node 6 and node 7, the signal quality of route 743 and route 763 is poor. Therefore, the route is estimated as link failure by DSR protocol and it brings short employ time of the route. The stability of the single-hop route 730 is poor and the employ time is short. However, in our current testbed environment, we are not able to find the quality of physical signal especially in route selection arithmetic.

The data delivery ratio case of video stream A has been improved accordingly. In the original DSR, Node 7 sent 20,168 packets, among which 17,259 packets (85.28%) have a valid route to node 3 at the time. Node 3 received 15,113 packets, implying a delivery ratio 91.48%. In the improved DSR, Node 7 sent 83,043 packets, among which 82,033 packets

![Figure 4](image-url)

**Figure 4.** The change of path between node 7 and 3 with original DSR

![Figure 5](image-url)

**Figure 5.** The change of path between node 7 and 3 with improved DSR

**TABLE I. THE DETAILS OF Y-AXIS IN FIGURE 4**

<table>
<thead>
<tr>
<th>Y-axis</th>
<th>Means</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Path: 0. There is no available route path between node 7 and 3</td>
</tr>
<tr>
<td>1</td>
<td>Path 1: 192.168.2.7 -&gt; 192.168.2.4 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>2</td>
<td>Path 2: 192.168.2.7 -&gt; 192.168.2.9 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>3</td>
<td>Path 3: 192.168.2.7 -&gt; 192.168.2.9 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>4</td>
<td>Path 4: 192.168.2.7 -&gt; 192.168.2.6 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>5</td>
<td>Path 5: 192.168.2.7 -&gt; 192.168.2.4 -&gt; 192.168.2.9 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>6</td>
<td>Path 6: 192.168.2.7 -&gt; 192.168.2.4 -&gt; 192.168.2.6 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>7</td>
<td>Path 7: 192.168.2.7 -&gt; 192.168.2.9 -&gt; 192.168.2.6 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>8</td>
<td>Path 8: 192.168.2.7 -&gt; 192.168.2.5 -&gt; 192.168.2.4 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>9</td>
<td>Path 9: 192.168.2.7 -&gt; 192.168.2.4 -&gt; 192.168.2.5 -&gt; 192.168.2.3</td>
</tr>
</tbody>
</table>

**TABLE II. THE DETAILS OF Y-AXIS IN FIGURE 5**

<table>
<thead>
<tr>
<th>Y-axis</th>
<th>Means</th>
</tr>
</thead>
<tbody>
<tr>
<td>700</td>
<td>Path 700: There is no available route path between node 7 and 3</td>
</tr>
<tr>
<td>730</td>
<td>Path 730: 192.168.2.7 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>743</td>
<td>Path 743: 192.168.2.7 -&gt; 192.168.2.4 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>753</td>
<td>Path 753: 192.168.2.7 -&gt; 192.168.2.5 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>763</td>
<td>Path 763: 192.168.2.7 -&gt; 192.168.2.6 -&gt; 192.168.2.3</td>
</tr>
<tr>
<td>793</td>
<td>Path 793: 192.168.2.7 -&gt; 192.168.2.9 -&gt; 192.168.2.3</td>
</tr>
</tbody>
</table>
(98.78%) have a valid route to node 3 at the time. That is, only 1.22% of packets get lost because no path is found. Node 3 received 77,637 packets, implying a delivery ratio 93.49%, and end-to-end loss rate is 6.51%. The backup route mechanism works well and only a small fraction of packets get lost. There are new 4899 control packets added. The new control overhead introduced is 5.89%, making the improved DSR tested with a higher delivery ratio. The failure cases of primary, backup path during the whole transmission process of video stream A with improved DSR (about 50 minutes) were observed. There were total of 59 link failures, 44 occasions where the backup routes failed, and 15 occasions where both routes failed simultaneously. However, during the 800 seconds of the original DSR experiment, there are total of 71 link failures. It shows that it is very likely that there is no available path between source and destination nodes. In this case, new Route Request should be initiated and the corresponding delay is larger resulting in inconsistency of video stream at the receiver. The QoS of multimedia delivery is not guaranteed.

Although we have set the network data transmission rate at 5.5Mbps fixed, the achieved transmission rate fluctuated from time to time. Sometimes it even dropped to 0, i.e., there is no route from source to target. Figure 6(b) shows the change of bandwidth at node 7. We can see that the throughput of node 7 is always about 150kbps. But it was 0 during the period between 10:14:25 and 10:15:05. The reason is that the video process software of mpeg4ip cannot send packets during that time and is not caused by route breaks. In fact, the transmission is quite stable. By comparing Figure 6(b) and (c), we can see that the changes of the throughput are similar at node 3 and node7. The experimental processes and results of video stream B are similar to that of stream A.

VII. CONCLUSIONS

In wireless mesh networks, by the DSR mechanism, the Route Discovery is initiated after link failures can incur large delay and high packet loss rate, which brings up great challenges in transmitting real time multimedia traffic. This paper has presented an implementation of a simple path bandwidth estimation algorithm and integration of the backup route mechanism with the original DSR protocol. Results from the experiments show that the proposed solution can decrease numbers of route failures and increase the delivery rate in the wireless mesh network consisting of a small quantity of low speed mobile nodes. The results also indicate that the backup route scheme is effective to service of multimedia traffic over wireless mesh networks. In the future, we are going to work on the backup route mechanism to support a more intensive higher speed mobile nodes environment.

ACKNOWLEDGMENT

This research was supported in part by the National Natural Science Foundation of China (NSFC) under grant No. 60472078 and No. 90604013. We would like to thank Zhi Liao, Zhibin Dou and etc for their help. Thanks also to Tony Sun and Xin Wang for helpful language and grammar.

REFERENCES