Chapter 3

Thesis Details

Chapter 3  Thesis Details ................................................................. III-2

3.1  Background ................................................................. III-2

3.2  Protocol Overview ......................................................... III-4

3.3  Packets Formats .......................................................... III-5

   3.3.1  Hello Packet ......................................................... III-5

   3.3.2  Request Packet ......................................................... III-7

   3.3.3  Reply Packet .......................................................... III-8

3.4  Routing Table .............................................................. III-10

3.5  ID Management ............................................................. III-11

3.6  References ................................................................. III-12
THESIS DETAILS

3.1 BACKGROUND

Any multimedia application to transport over ad hoc wireless network is a very challenging task because of the unreliable links. There is no guarantee for the link to remain stable for the required time period. In case of wired networks the links are fixed so we can trust on the links. The unreliability is due to the frequent path breaks due to the dynamic mobility of the nodes and the dynamic topology created.

Typical multimedia applications such as streaming require higher reliability connection than that provided by the single link. In a network consisting of mobile nodes, the connection between a source and destination may break down and has to be updated regularly. When the links breaks the time is needed to setup a new path for the destination. This time requirement is unacceptable in cases where destination is far away from the source node and it will temporarily disrupt the multimedia transmission. Instead of this single path if multiple paths are maintained than the time needed to setup a new path is in parallel to the packets transmission. Only the time is requiring during the first search. So the time for the source to wait for the data transmission is reduced.

Unlike traditional protocols that will choose a single path through the network, the protocol used must deliver multiple paths from a source to a destination. Also, the Quality of Service (QoS) aspects of each path (e.g. delay, bandwidth, and cost) must be taken into account. Finally, care must be taken that the different routes do not share nodes other than the source and destination nodes.
The advantages of this are given below:

First, Multipath transport distributes traffic load in the network more evenly. For example, a large burst of data, e.g., an Intra or I video frame, can be partitioned into several smaller bursts, each transmitted on a different path. A high rate video can be partitioned into several sub-flows, each with a lower rate and sent on a different path. Such balanced load results in less congestion inside the network. Thus the video packet losses caused by router buffer overflow can be effectively reduced.

Second, Multipath transport provides a larger aggregate capacity for a multimedia session. In an ad hoc network, since the available link bandwidth may be limited and time varying, a high rate flow may not find enough available capacity on a single path. With Multipath transport, the flow can be partitioned into several thinner sub-flows, each of which can be accommodated by a path.

Third, if a set of disjoint paths are used in Multipath transport, losses experienced by the sub-flows may be independent to each other. When a path is down because of a link failure, which happens more often in an ad hoc network than in a wired network, it is likely that some other paths are still in good condition. Thus the receiver can always receive some data during any period.

To summarize, the use of Multipath transport for real-time multimedia applications in ad hoc networks can effectively reduce packet losses, provide better scalability, and provide un-interrupted display of video even with the presence of frequent link breaks.

The protocol must provide multiple, loop-free, (preferably) node-disjoint paths from a source to a destination. Since multimedia streaming is primary goal, the multiple routes need to be used simultaneously.
3.2 PROTOCOL OVERVIEW

The protocols for ad hoc wireless network are different from that of the wired or static network. The main difference is due to the dynamic topology. There is no central coordinator or administration that maintains all the nodes in the network. The routing is to be done by the node itself and the information related to the network is also to be transmitted and maintained by the nodes. So the node itself acts as the router and transmits the packets towards the destination by looking into the routing table.

The new Multipath routing protocol is based on the approach used in AODV i.e. on-demand basis and it reactive routing protocol. In this protocol different control messages are transmitted before the actual transmission takes place. The control information transmitted by the nodes are the REQUEST, REPLY, ERROR and the HELLO packets and all the packets are used for different purpose.

On-Demand Routing

In the on-demand routing scheme, a node builds up a route by flooding a query to all nodes in the network. The request packet "picks up" the IDs of the intermediate nodes and stores them in a path field. On detecting the query, the destination or any other node that has already learned the path to destination answers the query by sending a "reply" response packet back to the sender. Since multiple responses may be produced, multiple paths may be computed and maintained. After the paths are computed, any link failure will trigger another query/response so the routing can always be kept up to date. They introduce excessive control overhead since they require frequent flooding, especially when mobility is high and traffic is dense and
uniformly distributed. As a result, on-demand routing protocols are only suitable for wireless network with high bandwidth, small packet transmission delays.

3.3 PACKET FORMATS

The different packet format used in the Multipath routing protocol are describe below with their field types. These packets are called the control packets and are not used in the actual data transmission. These packets are used only for the management of the network at each node. All these packets have different formats and functions are different for all of them.

3.3.1 Hello Packet

Hello packets format is as shown in Figure 3.1. These packets are broadcasted by every node in the network at regular interval of time. The Hello packets are usually used to find out the neighbors in the networks by the nodes. The neighbors are used during the Request for any destination. The hello packet contains the information of the sent node address and the time to live field. Hello packets have the lifetime of only 1, so as soon as it is received by the neighbors it extracts the required information and than the hello packet is dropped. The hello packet when received by the node it first finds the information about which node has sent the packet, after that it adds the address of the sent node in the neighbor list of the receiving node. If the information is already in the receiving node it simply updates the information of expire time of the neighbor. When expire time reaches its limit, the neighbor is automatically deleted which is handled by the other timer. Expire timer is required in the ad hoc network because of the node mobility.
The following are the fields that are used in the hello packet:

- **Type**: The type field specifies the type of packet i.e. whether it is hello, request, reply or other control packets.

- **Hop Count**: This field specifies the number of hop it has to traverse in the network. It is set to 1, so that it is received only by the neighboring node and by receiving this hello packet it updates its neighbor list entry.

- **Sequence Number**: This field is set so that the node receives the update packets. When the node receives the hello packet, it compares it with the latest sequence number it has received. If the sequence number of the received packet is less than the previously received sequence number than the information is not valid.

- **Originator IP address**: This field specifies the originator of the hello packet. By this the receiving node identifies the neighbor and adds in the neighbor list.

- **Lifetime**: This field is set to 1 so that the packet does not travel in the network.
3.3.2 Request Packet

The protocol is on-demand routing protocol, so the path from the source to destination is searched only when there is data to send. Whenever any node wants to send data packets it first waits until the path is found to reach the destination.

![Request Packet format](image)

Figure 3.2: Request Packet format

Figure 3.2 shows the packet format of Request packet. The fields of the request packet are described below:

- **Type**: The type field specifies the type of packet. For this it is "Request" type.
- **Hop Count**: This field specifies the number of hop it has to traverse in the network. It is set to 1, so that it is received only by the neighboring node and by receiving this hello packet it updates its neighbor list entry.
- **RREQ ID**: This field describes the ID that is unique for the entire route that is request.
- **Destination IP address**: The destination IP address for which the request is to be made.
• Destination Sequence Number: The latest sequence number received in the past by the originator for any route towards the destination.

• Originator IP address: This field specifies the originator of the hello packet. By this the receiving node identifies the neighbor and adds in the neighbor list.

• Originator Sequence Number: The current sequence number to be used in the route entry pointing towards the originator of the route request.

When path to the destination is found by the REQUEST packet, the reply packet is sent by the destination node towards the source node. The source node sends the request packet by broadcasting the request packet to all its neighbors. The number of request is dependent on the number of neighbors it has in the neighbor list, which is filled by the hello packet.

When the request reaches to the neighbor, they broadcast this request into the network. All the intermediate nodes that come towards the destination also broadcast the request until it reaches the destination.

When the request comes to the intermediate node, the node adds the reverse path into the routing table of its own. This reverse route can then be used by the reply message that will come from the destination. The intermediate node if received the request from the same source, it will discard the request and drop the packet.

3.3.3 Reply Packet

REPLY packet as shown in Figure 3.3 is sent when the REQUEST packet reaches to destination node of the source request. Reply packet replies to all the neighbors from
which it has received the route request with taking into consideration that the reply is not to be sent for the request ID whose reply has been sent.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>

**Figure 3.3: Reply Packet Format**

- **Type**: The type field specifies the type of packet. For this it is "Reply" type.

- **Hop Count**: This field specifies the number of hop it has to traverse in the network. It is set to 1, so that it is received only by the neighboring node and by receiving this hello packet it updates its neighbor list entry.

- **RREP ID**: This field is the ID for the route request for whom the reply is sent.

- **Destination IP address**: The destination IP address for which the request is to be made.

- **Destination Sequence Number**: The latest sequence number received in the past by the originator for any route towards the destination.

- **Originator IP address**: This field specifies the originator of the hello packet.
By this the receiving node identifies the neighbor and adds in the neighbor list.

- Life time: The time for which nodes receiving the Route Reply consider the route to be valid.

If the reply packet is received by the intermediate node it simply forwards the packet by looking into the routing table which it had developed during the route request. The route reaches the destination; the destination node sends the data that it has queued.

### 3.4 ROUTING TABLE

Multipath routing protocol deals with the routing table management. The routing table shown in Figure 3.4 is to be updated when even the packets reaches to the node. The different fields that are used in the routing table are described below.

```
+---------------------------------------------+
| Dest_IP | ID   | Seq_No | Hops   | Next_Hop | Expire_Time | Flags | +---------------------------------------------+
+---------------------------------------------+
```

Figure 3.4: Routing Table Format

- **Dest_IP**: - Destination IP Address. This is 32-bit unsigned integer which identifies the destination address and the port number for the routing table entry. For the ad hoc network it is set to 255 for the ad hoc network.
- **ID**: - Broadcast ID which is unique to every route. It is 32-bit unsigned integer which identifies the unique route from the many routes available to the destination
- **Seq_No**: - Sequence number of the route, which is used for update. It is 32-bit unsigned integer. This field is used to identify the updated packet
that is received by the nodes.

- **Hops:** This field identifies the number of hops to reach the destination. It is unsigned 8-bit integer.

- **Next Hop:** This field identifies the next hop to which it has to forward the packet when the destination address of the packet is the first field i.e. the Destination IP in the network to reach the destination. It is 32-bit address and the port number and port address is always 255.

- **Expire Time:** Expire time of the route after which the route is no longer valid. It is 64-bit floating-point number. When this time reaches the routing table entry is deleted.

- **Flags:** This is used for whether the route is UP or DOWN; If the route is UP than the packets can be forwarded to this route else if DOWN than can't. It is 8-bit unsigned integer.

- **Multipath routing protocol deals with the routing table management. The routing table is to be updated whenever the packet reaches to the node.**

### 3.5 ID MANAGEMENT

The ID table shown in Figure 3.5 is maintained at each node. This table is update\added at each time of request\reply packet received at the node. The fields used this table are described below.

```
+-----------------------------+-----------------------------+-----------------------------+-----------------------------+
<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>ID</th>
<th>Reply Sent</th>
</tr>
</thead>
</table>
+-----------------------------+-----------------------------+-----------------------------+-----------------------------+
```

Figure 3.5: ID table format

- **Source:** Source address of the request
• Destination: Destination address of the request
• ID: Unique ID that is sent at each request by the source node
• Reply Sent: a Boolean variable that is set false when the request is sent and update to true when the reply is received by the nodes. The nodes here are the source, intermediate and the destination.

3.6 References


[6] International Telecommunication Union (ITU_T) Study Group 12:

"Performance and quality of service - Lead Study Group on Quality of
Service and performance",


[7] Tsu-Wei Chen, "Efficient Routing and Quality of Service Support for Ad Hoc
Wireless Networks", University of California, Los Angeles, 1998.

[8] Ian D. Chakeres and Elizabeth M. Belding-Royer, "AODV Routing Protocol
Implementation" Design Dept. of Electrical & Computer Engineering
University of California, Santa Barbara.

Destination Sequenced Distance Vector Routing [DSDV] for mobile

[10] David B. Johnson, David A. Maltz, and Josh Broch, "DSR: The Dynamic
Computer Science Department Carnegie Mellon University Pittsburgh, PA
15213-3891 http://www.monarch.cs.cmu.edu/