

## DECLARATION

I, Alexa Morris, based on my personal knowledge and information, hereby declare as follows:

1. I am Managing Director of the IETF Administration LLC and have held that position since the LLC was formed in August 2018. Prior to that, starting on January 1, 2008, I was the Executive Director of the Internet Engineering Task Force, which was an activity of the Internet Society. Since the business of IETF did not change in any materially relevant manner with the formation of the LLC, I will collectively refer to both the activity and the LLC as IETF.

2. One of my responsibilities with IETF has been to act as the custodian of Internet-Drafts and records relating to Internet-Drafts. I am familiar with the record keeping practices relating to Internet-Drafts, including the creation and maintenance of such records.

3. I hereby declare that all statements made herein are of my own knowledge and information contained in the business records of IETF and are true, and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements may be punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

4. Since 1998, it has been the regular practice of the IETF to publish Internet-Drafts and make them available to the public on its website at [www.ietf.org](http://www.ietf.org) (the IETF website). The IETF maintains copies of Internet-Drafts in the ordinary course of its regularly conducted activities.

5. Any Internet-Draft published on the IETF website was reasonably accessible to the public and was disseminated or otherwise available to the extent that persons interested and ordinarily skilled in the subject matter or art exercising reasonable diligence could have located it. In particular, the Internet-Drafts were indexed and searchable on the IETF website.

6. Internet-Drafts are posted to an IETF online directory. When an Internet-Draft is published, an announcement of its publication that describes the Internet-Draft is disseminated. Typically, that dated announcement is made within 24 hours of the publication of the Internet-Draft. The announcement is kept in the IETF email archive and the date is affixed automatically.

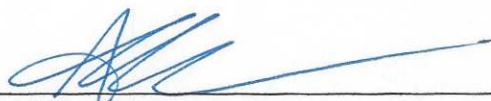
7. The records of posting the Internet-Drafts in the IETF online repository are kept in the course of the IETF's regularly conducted activity and ordinary course of business. The records are made pursuant to established procedures and are relied upon by the IETF in the performance of its functions.

8. It is the regular practice of the IETF to make and keep the records in the online repository.

9. Exhibit 1 is a true and correct copy of an announcement of the publication of draft-ietf-manet-olsr-00.txt, titled "Optimized Link State Routing Protocol". I have determined that an announcement of the publication of this Internet-Draft was made on November 21, 1998. Therefore, based on the normal practice of the IETF, that Internet-Draft was reasonably available to the public within 24 hours of that announcement. At that time, the Internet-Draft would have been disseminated or otherwise available to the extent that persons interested and ordinarily skilled in the subject matter or art, exercising reasonable diligence, could have located it.

Pursuant to Section 1746 of Title 28 of United States Code, I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct and that the foregoing is based upon personal knowledge and information and is believed to be true.

Date: 13 August 2019

By:   
Alexa Morris

# Exhibit 1

Internet Draft  
IETF MANET Working Group  
Expiration : 18 May 1999

Philippe Jacquet  
Paul Muhlethaler  
Amir Qayyum  
INRIA Rocquencourt, France  
18 November 1998

Optimized Link State Routing Protocol  
draft-ietf-manet-olsr-00.txt

Status of this Memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

To view the entire list of current Internet-Drafts, please check the "l1d-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), ftp.nordu.net (Europe), munnari.oz.au (Pacific Rim), ftp.ietf.org (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this memo is unlimited.

Abstract

This document describes a routing protocol for mobile ad hoc networks. The protocol is based on the link state algorithm. It uses the multi-point relays to calculate the route towards any destination in the network. The protocol is particularly suitable to the large dense networks with high nodal mobility and topological changes.

This routing protocol is developed to work with the IMEP protocol. It uses the different functionalities provided by IMEP, like link status sensing with neighbors, multi-point relay forwarding, etc.

The protocol exchanges topology information with other nodes of the network at regular intervals. It uses the multi-point relays to do efficient flooding of its control messages in the network. These are only the multi-point relays which are used to form the route towards any destination in the network.

Multi-point relays are selected among the one hop neighbors with "symmetric" i.e. bi-directional link. Therefore, selecting the route through multi-point relays automatically avoids the problems associated with data packet transfer on uni-directional links; like the problem of getting the ACKnowledgement for the data packets at each hop, which cannot be received if there is a uni-directional link in the selected route.

## 2. Terminology

connection

A communication channel or medium \*on the same physical

interface\*, over which the nodes can communicate with each other.

#### link

A logical communication channel between two nodes over which they can communicate with each other.

#### node

A MANET router that implements this Link State Routing protocol.

#### multi-point relay (MPR)

A node which is selected by its one-hop neighbor node X to "re-transmit" all the broadcast packets that it will receive from X, provided that the same packet is not already received, and the hop count field of the packet is greater than zero.

#### asymmetric link

A uni-directional \*link\* (not connection) between two neighbor nodes, i.e. node X can hear node Y, but node Y can not hear node X. So the link X-Y is asymmetric from node X's perspective, and this link does not exist from node Y's perspective (as Y is not hearing X).

#### symmetric link

A bi-directional \*link\* (not connection) between two neighbor nodes, i.e. node X and node Y, both can hear each other. This bi-directional link can be a union of two oppositely-directed uni-directional connections using different interfaces.

#### holding time

The lifetime associated to an entry in any table. That entry is kept in the table for a period equal to its holding time. If the entry is not refreshed during this period, it is removed from the table when its holding time expires.

#### refreshing an entry

The holding time of the entry in the table is updated to a specified value `<table_name>_HOLDING_TIME`.

### 3. Applicability Section

This section dictates the characteristics of the OLSR protocol, as

specified in the Applicability Statement draft.

### 3.1 Networking Context

The protocol is best suited to the large "dense" networks, as the optimization done using the multi-point relays works well in this context. More the network is dense and large, more optimization is achieved as compared to the normal link state algorithm. The thing which may restrict here is the size of the topology table which is  $O(N^2)$ , where  $N$  is the number of nodes in the network. (Although we have developed an algorithm which reduces its size to  $O(N)$  only).

This protocol is suited for the networks where the traffic is random and sporadic between "several" nodes (and NOT between a small specific set of nodes) of the network, and still better if these <source, destination> pairs change with time. (These changes will initiate enormous traffic (Query flooding) in case of source routing, but nothing in OLSR, as the routes are maintained for each destination all the time).

### 3.2 Protocol Characteristics and Mechanisms

\* Does the protocol provide support for unidirectional links? (if so, how?)

No. It uses only bi-directional "links", but these links may be composed of oppositely directed uni-directional "connections".

\* Does the protocol require the use of tunneling? (if so, how?)

No.

\* Does the protocol require using some form of source routing? (if so, how?)

No. The protocol uses hop-by-hop routing.

\* Does the protocol require the use of periodic messaging? (if so, how?)

Yes. Periodically, all nodes in the network send a message containing the addresses of its multi-point relays. This information helps other nodes to build routes to that node through these multi-point relays.

\* Does the protocol require the use of reliable or sequenced packet delivery? (if so, how?)

No. As the TC packet is sent periodically, it needs not to be sent reliably. TC packet also contains the sequence number of



the "most-recent-information" (MPR SEQ NUM), so UN-sequenced delivery of packets will also not create any problem.

\* Does the protocol provide support for routing through a multi-technology routing fabric? (if so, how?)

Yes. It provides support for routing through a multi-technology routing fabric by using Router IDs (see IMEP draft). RID may be associated with more than one IP addresses, representing different physical interfaces.

\* Does the protocol provide support for multiple hosts per router? (if so, how?)

Yes. As mentioned in the preceding question, RID may be associated with more than one IP addresses, and these IP address may represent different hosts associated to that router.

\* Does the protocol support the IP addressing architecture? (if so, how?)

The OLSR protocol uses RIDs, but the mapping of RIDs to IP addresses is provided by IMEP through its NARP service. So in this sense, OLSR supports the IP addressing architecture.

\* Does the protocol require link or neighbor status sensing (if so, how?)

Yes. The protocol requires the link status sensing. It needs to be notified when a bi-directional link fails with a neighbor, or when a neighbor with a bi-directional link is added. This service is provided by IMEP to OLSR.

\* Does the protocol have dependence on a central entity? (if so, how?)

No. All the routers in the network have their own routing tables and does not depend on any specific node (source, etc).

\* Does the protocol function reactively? (if so, how?)

No. But it decreases and increases the interval (within certain limits) of sending the TC packet periodically, depending upon the rate of link changes in its neighborhood.

\* Does the protocol function proactively? (if so, how?)

Yes. It periodically sends the information about its multi-point relay set, which helps other nodes to build routes to it.

\* Does the protocol provide loop-free routing? (if so, how?)

As the protocol uses link state algorithm, so implicitly, the routing is loop-free.

\* Does the protocol provide for sleep period operation? (if so, how?)

Yes, it can provide support for sleep period operation. To enable this feature, the protocol should select its multi-point relays from only among the "p-supporters" (power conservation supporters) i.e. the nodes which can (or agree to) store its packets while it is in sleep mode.

\* Does the protocol provide some form of security? (if so, how?)

No, not itself. But it uses IMEP which provides authentication and security.

\* Does the protocol provide support for utilizing multi-channel, link-layer technologies? (if so, how?)

Yes.

#### 4. Protocol Overview

This protocol is developed to work with the IMEP protocol. It uses various functionalities of the IMEP, specially the link status information with the neighbors, multi-point relays selection and forwarding, etc. It needs from the IMEP protocol to provide the information about

- its neighbors with which the status of the link is symmetric i.e. bi-directional
- the list of multi-point relays along with the sequence number of their selection.

This information is kept in the Neighbor Table, and is updated accordingly whenever the up to date information is passed from IMEP to this routing protocol (OLSR).

The protocol is pro-active in nature, and by periodic exchange of messages, it establishes its routing table in which it stores the information of the route to each destination node in the network. This information is updated when

- a change in the neighborhood is detected concerning a symmetric link; or
- a route to any destination is expired (because the corresponding topology entry is expired) or
- a better (shorter) route is found for a destination.

The protocol relies on the multi-point relays, and calculates the routes towards each destination through these nodes. To implement this, each node in the network periodically broadcast the information of its multi-point relays. Upon receipt of this MPR information, each node calculates (or updates) the route towards each destination. So these are the multi-point relays which form the route towards any destination.

The protocol does not do the source routing, instead it performs hop by hop routing.

## 5. Multi-point relays

The idea of multi-point relays is to minimize the flooding of broadcast messages in the network by reducing/optimizing the duplicate re-transmissions in the same region. Each node in the network selects a set of nodes in its neighborhood, who will re-transmit its packets. This set of selected neighbor nodes is called the multi-point relays of that node. Each node selects its multi-point relay set in a manner to cover all the nodes that are two hop away from it. The neighbors which are not in the MPR set, do read and process the packet but \*do not\* re-transmit this broadcast message. For more details about the multi-point relays, refer to the IMEP protocol [1], where the multi-point relays selection and forwarding is implemented.

## 6. Interface with IMEP

### 6.1 OLSR registration

To be operational, OLSR will be registered with IMEP, as specified by the IMEP protocol, with the protocol type value equal to OLSR, a handle to receive objects from IMEP (implementation dependent), an epitaph object as null (for the time being) and the Link-level mode of IMEP signaling support. It also request IMEP protocol to provide

LCSS (Link Connection Status Sensing) and MPR (Multi Point Relay) support.

## 6.2 IMEP Signalling Support

For OLSR to perform its task of calculating the routes to any destination in the network, it does not require to know if a link is composed of bi-directional connections on the same physical interface or it is using two (or more) oppositely directed uni-directional connections on different interfaces to form a symmetric link between two one-hop neighbor nodes. All it requires to know is whether a link with a neighbor node is asymmetric or symmetric. So at the time of registration, the "Link-level" signaling support will be indicated to IMEP, and OLSR will take the advantage of the links composed of multiple interface connections.

## 6.3 Connection Notification mode

OLSR is sensitive only to the symmetric links with its one-hop neighbors. OLSR computes the routes using the multi-point relays, which are selected among the one hop neighbors with symmetric link only. So the UNI-directional connection notification mode will be demanded from IMEP.

Jacquet, Muhlethaler and Qayyum

[Page 8]

Internet draft

Optimized Link State Routing

18 November 1998

## 6.4 Broadcast signaling modes

OLSR will select Multiple Interface (MI) mode, in order to be able to establish route to/through the nodes having different physical interfaces or technologies, but are operating in the same MANET.

## 6.5 Neighbor Broadcast Reliability

OLSR will request BROADCAST (implicit) delivery for its control packets (TC packets) as it does not need the reliable mode. The TC packets are sent periodically so if a packet is lost, the information is assumed to be passed in the successive packets.

## 7. Information data bases

To perform the task of routing the packets, each node manages some tables. Each node maintains four tables: Duplicate table, Neighbor table, Topology table and Routing table.

### 7.1 Duplicate Table

In the duplicate table, each node records the information about the packets it has already received. This information helps the node to identify the already received packets, so that it does not waste time in processing again the packet, and silently discards the packet. The information is recorded in the duplicate table as a duplicate entry.

The duplicate table has the following format to record these entries:

```
1.   D_addr   D_seq   D_time
2.   D_addr   D_seq   D_time
3.   "        "        "
```

Each entry in the table consists of `D_addr`, `D_seq` and `D_time`, which specifies that the packet with sequence number `D_seq` is already received from the node with the address `D_addr`. If the same packet is received again during the time `D_time`, this "duplicate" packet will be silently discarded without any processing. The `D_time` is the holding time of the entry in the table, upon expiry of which the entry is no longer valid and hence removed. This life time of the entries helps to keep the size of the table limited, by discarding the old entries about the packets which are less likely to be received again, after keeping these entries for a sufficient time.

## 7.2 Neighbor Table

In the neighbor table, each node records the information about its one-hop neighbors, and the status of the link with these neighbors. This table is constructed from the information given by IMEP through Connection-notification and MPR information. The information is recorded in the Neighbor table as a neighbor entry. The neighbor table has a following format to record these entries:

```
--- N_MPR_seq ---
```

```
1.   N_addr   N_status
2.   N_addr   N_status
3.   "        "
```

Each entry in the table consists of `N_addr` and `N_status`, which specifies that the node with address `N_addr` is a one-hop neighbor to this local node and the status of the link between them is `N_status`. `N_status` can be asymmetric, symmetric or MPR. The link status as MPR specifies that the link status with the neighbor node `N_addr` is "symmetric" \*AND\* further more, this `N_addr` is also selected as a multi-point relay by this local node.

Neighbor table also keeps a sequence number value `N_MPR_seq` which specifies that the local node keeping this neighbor table has selected its most recent MPR set with the sequence number `N_MPR_seq`. The nodes in the MPR set are indicated in the neighbor table with their `N_status` equal to MPR. Every time a node selects or updates its multi-point relay set, this `N_MPR_seq` is incremented to a higher value. This multi-point relay set is re-calculated each time when:

- a change in the neighborhood is detected when either a

- symmetric link with a neighbor is failed, or a new neighbor with a symmetric link is added; or
- a change in the two-hop neighbor set (with either symmetric or asymmetric link) is detected.

The selection of MPRs is done by IMEP, and OLSR just uses that information as such.

### 7.3 Topology Table

In the topology table, each node records the information about the topology of the network, and on the basis of this, the routing table is calculated.

A node records the information about each of the multi-point relays of other nodes in the network in its topology table as a topology entry. The topology entries are recorded in the topology table in the following format:

1.	T_dest	T_last	T_seq	T_time
2.	T_dest	T_last	T_seq	T_time
3.	''	''	''	''

Each entry in the table consists of T\_dest, T\_last, T\_seq, and T\_time which specifies that the node T\_dest has selected the node T\_last as a multi-point relay in the multi-point relay set with the sequence number T\_seq. Therefore, the node T\_dest can be reached in the last hop through the node T\_last.

Each topology entry has an associated holding time T\_time, upon expiry of which it is no longer valid and hence removed.

### 7.4 Routing table

On the basis of the information contained in the topology table and the neighbor table, a routing table is calculated. The route entries are recorded in the routing table in the following format:

1.	R_dest	R_next	R_dist
2.	R_dest	R_next	R_dist
3.	''	''	''

Each entry in the table consists of R\_dest, R\_next and R\_dist, which specifies that the node identified by R\_dest is estimated to be R\_dist hops away from the local node, and that the one hop

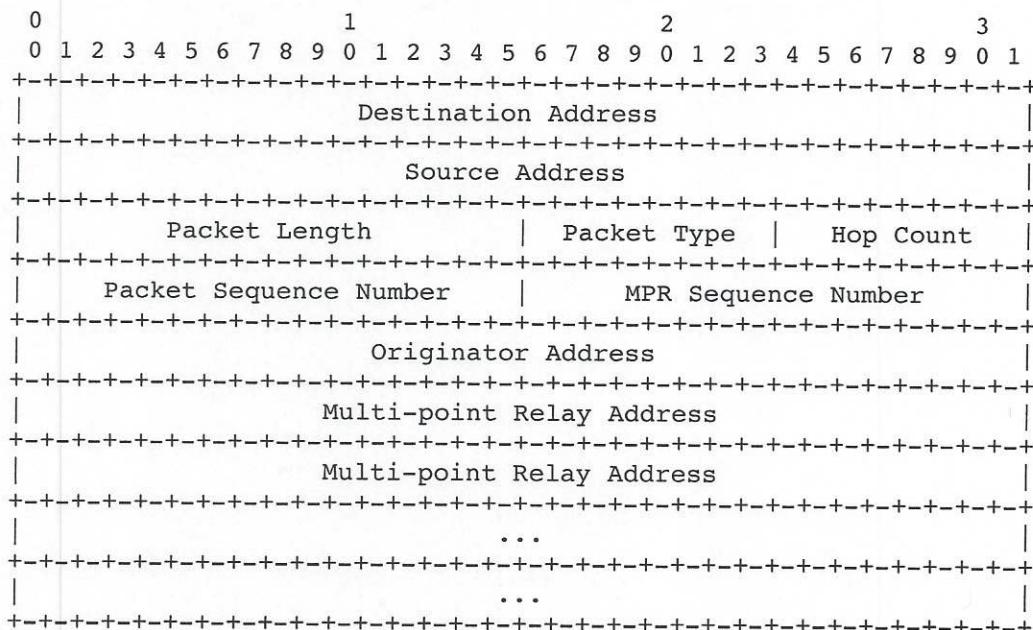
neighbor node with address R\_next in the next hop node in the route to R\_dest.

The routing table is re-calculated each time the neighbor table or the topology table or both are changed.

### 8. Multi-point Relay Information Declaration

A message is sent by each node in the network at regular intervals to declare its multi-point relay set. This message is called TC (Topology Control) packet. The information diffused in the network by these TC packets will help each node to calculate its routing table. The interval between the transmission of two TC packets depends upon whether the MPR set is changed or not, since the last TC packet transmitted. If no change occur in the MPR set, the TC packet will be sent after MAX\_TC\_PKT\_INTERVAL. When a change occur in the MPR set, the next TC packet will be sent after the MIN\_TC\_PKT\_INTERVAL. Then the default value for the interval again becomes MAX\_TC\_PKT\_INTERVAL, until the MPR set is changed again.

The proposed format of a TC packet is



#### 8.1 Description of the fields

## Destination address

For all the TC packets, this 4-bytes address field is always set to the broadcast address of the network, so that when this packet is diffused in the network, every node get this information and hence update its topology table.

## Source address

It is set to the address (Router ID) of the node \*transmitting\* this TC packet. This field should not be confused with the Originator Address of the TC packet. Whenever a node "re-transmit" the TC packet, this field is updated to that transmitter node's address (RID).

## Packet Length

This field contains the length of the whole TC packet in bytes, starting from the Destination Address field.

## Packet Type

The Packet Type field is set to the TC\_PACKET value.

## Hop Count

This field will contain the maximum number of hops a TC packet can attain. Every time when the TC packet is re-transmitted, this field is decremented by 1. When this field reaches zero, the TC packet is no more re-transmitted and is destroyed.

## Sequence Number

While generating the TC packet, the "originator" node will assign a unique identification number to this packet, and will put this number in the Sequence Number field. This sequence number will be different for all the packets originated by that node, which will help to recognize the duplicate reception of the packets.

## Originator Address

This field contains the address of the node which has originally generated this TC packet to declare its multi-point relay set. This field should not be confused with the Source Address field, which is changed each time to the address of the intermediate node which is "re-transmitting" this TC packet, while the Originator Address field in never changed.



### Multi-point Relay Sequence Number

A sequence number is associated with the multi-point relay set and every time a node selects/updates its multi-point relay set, it increments this sequence number. This number is sent in this MPR Sequence Number field of the TC packet to keep track of the most recent information. When a node receives a TC packet, it can decide on the basis of this MPR Sequence Number field, whether the information about the multi-point relays of the originator node is more recent than that it already have, or not.

### Multi-point Relay Address (MPR)

This field contains the address of the node which is selected as a multi-point relay by the Originator node (of the TC packet). All the node addresses of the multi-point relays of the Originator node are put in the TC packet, one after another. If the maximum allowed size of the TC packet (`MAX_TC_PKT_SIZE`) is attained and there are still some multi-point relay addresses which remain in the MPR set, then more TC packets will be generated, with different "Packet Sequence Number" but having the same "MPR Sequence Number", until all addresses in the multi-point relay set are transmitted.

Note: This protocol is designed to work with IMEP protocol with multi-point relay mode "enabled". Nevertheless, if it is not enabled in the IMEP protocol (for what so ever reason), then instead of multi-point relays, a node will declare all its neighbors with a symmetric link, in the TC packet. Nothing else will be required to change in this protocol to calculate its routes.

## 9 Information recording in the tables

### 9.1 Duplicate table

Each time a packet is received by OLSR, the protocol checks if this packet is already received earlier. If the Originator Address of the packet corresponds to `D_addr` of an entry in the duplicate table, and the value of the Sequence Number field of the packet is the same as the corresponding `D_seq` of the duplicate entry, then the packet is silently discarded and no further processing is done.

Otherwise, a new duplicate entry is recorded in the duplicate table with :

- D\_addr is set to the Originator Address of the packet,
- D\_seq is set to the Sequence Number of the packet, and
- D\_time is set to the holding time of the duplicate entries, which is a pre-specified value DUPLICATE\_HOLD\_TIME.

The packet is then processed further.

## 9.2 Neighbor table

This table is updated with the information provided by the IMEP protocol through Link/Connection notification and MRA packets. Hence the entries will contain the one hop neighbors as N\_addr along with the link status N\_status as asymmetric, symmetric or MPR. To keep this information up to date, IMEP is supposed to update this information whenever a change occur in its neighborhood (concerning a symmetric link only) or its MPR set.

## 9.3 Topology table

The entries in the topology table are recorded with the routing information that is exchanged among the network nodes through TC packets. On the receipt of a TC packet, the following procedure is executed to record the information in the topology table:

### Step 1:

If the Originator Address of the received TC packet corresponds to T\_dest of a topology entry in the topology table, then

If T\_seq of that topology entry is higher in value than the MPR Sequence Number field of the received TC packet, no further processing of the TC packet is done, and it is silently discarded.

If T\_seq of that topology entry precedes the value of the sequence number field of the received TC packet, that topology entry is removed.

### Step 2:

For each MPR address in the received TC packet

If the MPR Address corresponds to the T\_last of the topology entry whose T\_dest corresponds to the Originator Address of the TC packet, then the holding time T\_time of that entry is updated to TOPOLOGY\_HOLD\_TIME.

If the MPR address does not corresponds to T\_last of any topology entry whose T\_dest corresponds to the Originator Address of the TC packet, then

a new topology entry is recorded in the topology table with:

- T\_dest is set to the Originator Address of the TC packet,
- T\_last is set to the MPR address,
- T\_seq is set to the value of MPR Sequence Number in the TC packet,
- T\_time is set to the TOPOLOGY\_HOLD\_TIME

#### 9.4 Routing table

Each node maintains a routing table with it which allows it to route the packets for the other destinations in the network. The routing table is based on the information contained in the neighbor table and the topology table. Therefore, if any of these tables is changed, the routing table is re-calculated to update the route information about each destination in the network.

The following procedure is executed to calculate (or re-calculate) the routing table in case of a change in the neighbor table or the topology table or both:

1. All the entries of the routing table are removed.
2. Then new entries are recorded in the table for each destination in the network for which the route is known. All the destinations for which the route is broken or partially known are not entered in the table. This information is recorded in the following manner:

- 2.1 The new entries are recorded in the table starting with the one hop neighbors as the destination nodes.

For each neighbor entry in the neighbor table, whose link status is symmetric (or MPR), a new route entry is recorded in the routing table where R\_dest and R\_next are both set to the address of the neighbor and R\_dist is set to 1.

- 2.2 Then the new route entries for the destination nodes k+1 hops away, where k>0, are recorded in the routing table.

For each topology entry in the topology table, if its T\_dest does not correspond to R\_dest of any route entry \*AND\* its T\_last corresponds to R\_dest of a route entry whose R\_dist is k, then a new route entry is recorded in the routing table where R\_dest is set to T\_dest, R\_next is set to R\_next of the route entry whose R\_dest is T\_last, and R\_dist is set to k+1.

3. After calculating the routing table, the topology table entries which are not used in calculating the routes may be removed, if there is a need to save memory space. Otherwise, these entries may provide redundant routes.

## 10. References

1. Corson et al. Internet MANET Encapsulation Protocol. Internet draft, draft-ietf-manet-imep-spec-01.txt, August 21, 1998

## 11. Authors' Addresses

Philippe Jacquet  
Project HIPERCOM  
INRIA Rocquencourt  
BP 105  
78153 Le Chesnay Cedex, France  
Phone: +33 1 3963 5263  
Email: Philippe.Jacquet@inria.fr

Paul Muhlethaler  
Project HIPERCOM  
INRIA Rocquencourt  
BP 105  
78153 Le Chesnay Cedex, France  
Phone: +33 1 3963 5278  
Email: Paul.Muhlethaler@inria.fr

Amir Qayyum  
Project HIPERCOM  
INRIA Rocquencourt  
BP 105  
78153 Le Chesnay Cedex, France  
Phone: +33 1 3963 5273  
Email: Amir.Qayyum@inria.fr