IPv6 Support for OLSR in ns-3

Ankit Deepak, Shravya K.S., Radhesh Anand M.C. and Mohit P. Tahiliani
Wireless Information Networking Group (WiNG)
NITK Surathkal, Mangalore, India, 575025
adadeepak8@gmail.com, shravya.ks0@gmail.com, radheshanand@gmail.com, tahiliani@nitk.ac.in

ABSTRACT

Optimized Link State Routing Protocol (OLSR) is one of the most comprehensive and efficient routing protocols designed for Mobile Ad hoc Networks (MANETs). OLSR has its base in the link state algorithm, and is optimised to meet the requirements of wireless mobile networks. Currently, ns-3 provides an implementation of OLSR protocol which supports IPv4 addressing only. In this paper, we propose a model of OLSR in ns-3 which supports IPv6 addressing. Further, we validate our implementation by using an example program provided in the existing OLSR model of ns-3, and compare the results obtained from the proposed model with that of the existing model.

CCS Concepts

•Networks → Network simulations; •Computing methodologies → Model development and analysis;

Keywords

OLSR, IPv6, ns-3, Routing, MANETs

1. INTRODUCTION

Extensive research in the field of ad hoc routing has paved way for various classes of routing protocols namely, proactive, reactive and hybrid algorithms. One such routing protocol is OLSR which belongs to the class of proactive routing protocols. OLSR [1] has also been included in the list of experimental RFC and has been implemented in several network simulators. Implementation of OLSR supporting IPv4 addressing is available in ns-3. Apart from providing larger number of addresses, IPv6 has been successful in solving several issues of IPv4, including the functionalities pertaining to the configuration of network and routing. Hence, it is necessary to extend OLSR to work with IPv6 addressing as well. In this paper, we present an implementation and validation of OLSR protocol in ns-3 which supports IPv6 addressing.

WNS3, June 15-16, 2016, Seattle, WA, USA

ISBN 978-1-4503-2138-9.

DOI: 10.1145/1235

The rest of the paper is organised as follows: Section 2 provides a brief description of OLSR protocol. Section 3 discusses the implementation of OLSR in ns-3 which supports IPv6 addressing, and the problems encountered. Section 4 provides the validation of the proposed model by comparing it's results with that of the existing OLSR model.

2. BACKGROUND OF OLSR

OLSR [1] protocol tries to achieve optimisation by using the concept of Multipoint Relays (MPRs) [3], which reduces the number of transmissions required to pass the information about the network. Each node broadcasts a periodic message to it's immediate neighbor, consisting of the information about it's neighbors. Also, each node chooses certain immediate neighbors as MPR which are responsible for forwarding the link-state advertisement of this node. Other immediate neighbors do not forward this information. The MPR announces to the network that it has accessibility to the node which has selected it as an MPR by periodic updates. MPRs aid in establishing a route from source to a destination, and also effectively control the flooding of routing information in the network.

OLSR comprises of the following messages:

Hello messages: Each node periodically broadcasts hello messages to discover and maintain its immediate neighbors. This interval is known as HELLO_INTERVAL. Based on this information, the immediate neighbors of each node having symmetric link with its two-hop neighbors is chosen as MPR.

Topology Control Messages: Topology Control (TC) messages are broadcasted by MPR which provide information about node which selected them as MPRs to the network. Hence, all the nodes obtain information about the network via TC messages. Also, each node in a network can be reached directly or via it's MPR if there exists a path through which that node can be accessed.

HNA messages: Host and Network Association (HNA) messages are generated by each gateway which consist of information regarding the hosts and networks to which it is connected. MPRs flood these messages periodically to all the nodes in the network, for every interval known as HNA_INTERVAL.

Multiple Interface Declaration messages: If a node has more than one interface, then it sends a special type of message called "Multiple Interface Declaration". This message consists of all the interface addresses of the node along with one of the fixed interface address, called main_address which it arbitrarily chooses.

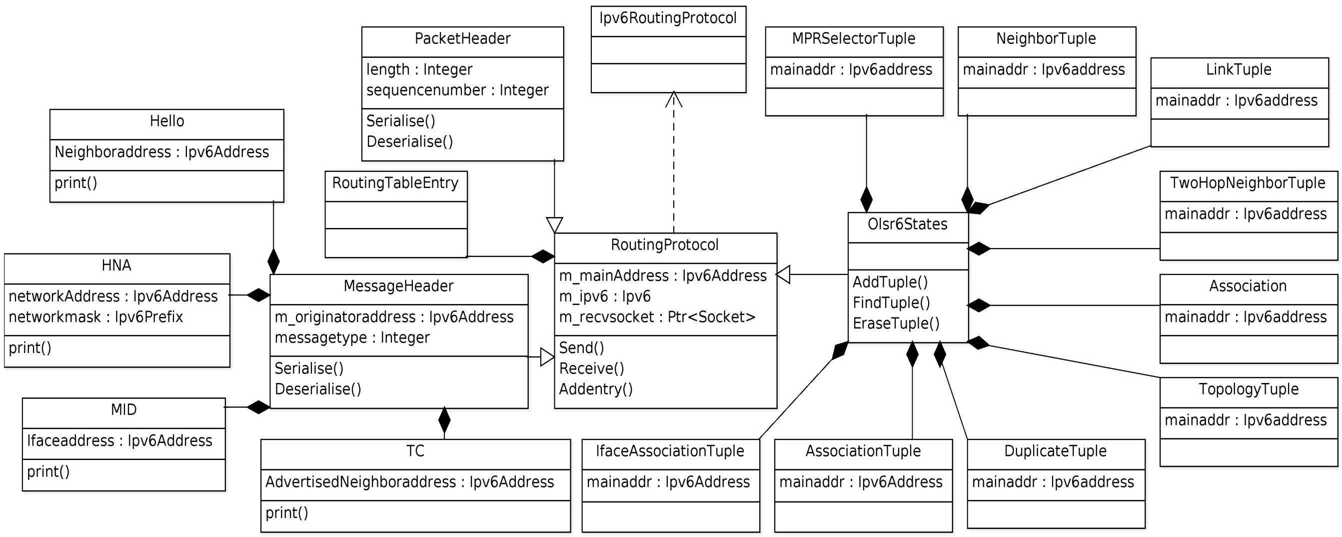

Figure 1: Class diagram for olsr6 module in ns-3

3. IPV6 SUPPORT FOR OLSR IN NS-3

In this section, we provide the implementation details of the newly proposed model for OLSR protocol with IPv6 support, as illustrated in the Figure 1. Most of the implementation is in-line with the existing ns-3 [2] model of OLSR which supports IPv4 addressing.

3.1 Protocol Header

The IPv6 addressing [4] support for OLSR uses headers similar to the one used by IPv4 implementation, except for the address fields increased to 16 bytes.

3.2 Protocol Implementation

Some of the key features of IPv4 addressing were removed while implementing the IPv6 addressing support for OLSR. The major change in IPv6 addressing [4] [5] is the removal of the provision of a broadcast and the presence of at least two addresses (link local and global) on every interface except one, i.e., the loopback interface. In OLSR, broadcast is used to share the link state advertisements with the neighbors of the node.

In the existing implementation of OLSR, a node was identified by its first address, named as m_mainaddress. Any OLSR message originating from this node had this address as its originator address. The further processing of this packet was done based on the source address extracted and destination from the IPv4 header. A similar implementation has been followed in OLSR IPv6 address support.

Additionally, there are multiple sockets used per nodes. Each global address of the node has a socket bound to it; they are used to send multicast messages to link local multicast address [6], FF02::1, is all node link local multicast address, and is the most closest match to to broadcast in IPv6 addressing, among all the different kinds of link local multicast addresses. In order to listen to the multicast address, another socket bound to that address is created in every node. The messages bound to the multicast addresses are routed using the static routing and are sent to every node in the link. The IPv6 header of this message when received at the receiver has the source address set to the global address of the sender to which the sending socket was bound to, and will have its destination address set to the IPv6 Address "::". To further process the OLSR message, we need the receiving interface of the message. We find this

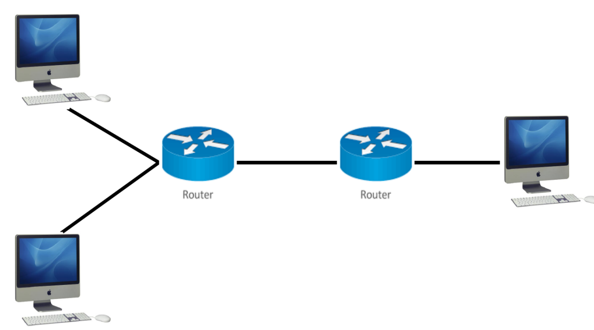

Figure 2: Simulation topology

Table 1: Simulation parameters

Parameter	Value
Node0 - Node2 Link Bandwidth	5 Mbps
Node0 - Node2 Link Delay	2 ms
Node1 - Node2 Link Bandwidth	5 Mbps
Node1 - Node2 Link Delay	2 ms
Node2 - Node3 Link Bandwidth	1.5 Mbps
Node2 - Node3 Link Delay	10 ms
Node3 - Node4 Link Bandwidth	1.5 Mbps
Node3 - Node4 Link Delay	10 ms
UDP DataRate	448,000 bps
UDP Packet Size	210 B
Queue	DropTail
Application start time	1.1 s
Application stop time	10.0 s
Simulation time	30.0 s

```
Node: 0, Time: +30.0s, Local time: +30.0s, Ipv4ListRouting table
  Priority: 10 Protocol: ns3::olsr::RoutingProtocol
Node: 0, Time: +30.0s, Local time: +30.0s, OLSR Routing table
Destination            NextHop          Interface       Distance
10.1.1.2               10.1.1.2                 1              1
10.1.2.1               10.1.1.2                 1              2
10.1.2.2               10.1.1.2                 1              1
10.1.3.1               10.1.1.2                 1              2
10.1.3.2               10.1.1.2                 1              1
10.1.4.1               10.1.1.2                 1              2
10.1.4.2               10.1.1.2                 1              3
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv4StaticRouting
Node: 0, Time: +30.0s, Local time: +30.0s, Ipv4StaticRouting table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
127.0.0.0       0.0.0.0         255.0.0.0       U     0      -      -   0
10.1.1.0        0.0.0.0         255.255.255.0   U     0      -      -   1
```

(a) Node 0

```
Node: 1, Time: +30.0s, Local time: +30.0s, Ipv4ListRouting table
  Priority: 10 Protocol: ns3::olsr::RoutingProtocol
Node: 1, Time: +30.0s, Local time: +30.0s, OLSR Routing table
Destination            NextHop          Interface       Distance
10.1.1.1               10.1.2.2                 1              2
10.1.1.2               10.1.2.2                 1              1
10.1.2.2               10.1.2.2                 1              1
10.1.3.1               10.1.2.2                 1              2
10.1.3.2               10.1.2.2                 1              1
10.1.4.1               10.1.2.2                 1              2
10.1.4.2               10.1.2.2                 1              3
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv4StaticRouting
Node: 1, Time: +30.0s, Local time: +30.0s, Ipv4StaticRouting table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
127.0.0.0       0.0.0.0         255.0.0.0       U     0      -      -   0
10.1.2.0        0.0.0.0         255.255.255.0   U     0      -      -   1
```

(b) Node 1

```
Node: 2, Time: +30.0s, Local time: +30.0s, Ipv4ListRouting table
  Priority: 10 Protocol: ns3::olsr::RoutingProtocol
Node: 2, Time: +30.0s, Local time: +30.0s, OLSR Routing table
Destination            NextHop          Interface       Distance
10.1.1.1               10.1.1.1                 1              1
10.1.2.1               10.1.2.1                 2              1
10.1.3.1               10.1.3.1                 3              1
10.1.4.1               10.1.3.1                 3              1
10.1.4.2               10.1.3.1                 3              2
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv4StaticRouting
Node: 2, Time: +30.0s, Local time: +30.0s, Ipv4StaticRouting table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
127.0.0.0       0.0.0.0         255.0.0.0       U     0      -      -   0
10.1.1.0        0.0.0.0         255.255.255.0   U     0      -      -   1
10.1.2.0        0.0.0.0         255.255.255.0   U     0      -      -   2
10.1.3.0        0.0.0.0         255.255.255.0   U     0      -      -   3
```

(c) Node 2

```
Node: 3, Time: +30.0s, Local time: +30.0s, Ipv4ListRouting table
  Priority: 10 Protocol: ns3::olsr::RoutingProtocol
Node: 3, Time: +30.0s, Local time: +30.0s, OLSR Routing table
Destination            NextHop          Interface       Distance
10.1.1.1               10.1.3.2                 1              2
10.1.1.2               10.1.3.2                 1              1
10.1.2.1               10.1.3.2                 1              2
10.1.2.2               10.1.3.2                 1              1
10.1.3.2               10.1.3.2                 1              1
10.1.4.2               10.1.4.2                 2              1
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv4StaticRouting
Node: 3, Time: +30.0s, Local time: +30.0s, Ipv4StaticRouting table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
127.0.0.0       0.0.0.0         255.0.0.0       U     0      -      -   0
10.1.3.0        0.0.0.0         255.255.255.0   U     0      -      -   1
10.1.4.0        0.0.0.0         255.255.255.0   U     0      -      -   2
```

(d) Node 3

```
Node: 4, Time: +30.0s, Local time: +30.0s, Ipv4ListRouting table
  Priority: 10 Protocol: ns3::olsr::RoutingProtocol
Node: 4, Time: +30.0s, Local time: +30.0s, OLSR Routing table
Destination            NextHop          Interface       Distance
10.1.1.1               10.1.4.1                 1              3
10.1.1.2               10.1.4.1                 1              2
10.1.2.1               10.1.4.1                 1              3
10.1.2.2               10.1.4.1                 1              2
10.1.3.1               10.1.4.1                 1              1
10.1.3.2               10.1.4.1                 1              2
10.1.4.1               10.1.4.1                 1              1
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv4StaticRouting
Node: 4, Time: +30.0s, Local time: +30.0s, Ipv4StaticRouting table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
127.0.0.0       0.0.0.0         255.0.0.0       U     0      -      -   0
10.1.4.0        0.0.0.0         255.255.255.0   U     0      -      -   1
```

(e) Node 4

Figure 3: **IPv4 Routing Table of all nodes**

Table 2: **Address mapping to the interface**

Interface	IPv4 Address	IPv6 Address
Node0 - 2	10.1.1.1	2001:0:1:0:200:ff:fe00:1
Node1 - 2	10.1.2.1	2001:0:2:0:200:ff:fe00:3
Node2 - 0	10.1.1.2	2001:0:1:0:200:ff:fe00:2
Node2 - 1	10.1.2.2	2001:0:2:0:200:ff:fe00:4
Node2 - 3	10.1.3.2	2001:0:3:0:200:ff:fe00:6
Node3 - 2	10.1.3.1	2001:0:3:0:200:ff:fe00:5
Node3 - 4	10.1.4.1	2001:0:4:0:200:ff:fe00:7
Node4 - 3	10.1.4.2	2001:0:4:0:200:ff:fe00:8

receiving interface by looking for an interface which is bound to an address in the same subnet as that of the source address. After we get sender and receiver interface, we process IPv6 OLSR messages in the same way as the IPv4 OLSR messages.

In order to validate our implementation, we chose the topology simulated in the *simple-point-to-point-routing.cc*, provided in the examples directory of existing OLSR model in ns-3. Figure 2 illustrates the topology for the same. The details of the simulation setup are as tabulated in Table 1. The IPv4 and IPv6 address mapping of corresponding interfaces is presented in Table 2.

4. MODEL VALIDATION

Figures 3 and 4 presents the snapshots of the routing table taken at the end of simulation for both, OLSR with IPv4 addressing and OLSR with IPv6 addressing. We can observe that the routing decision of OLSR remains unaffected by the choice of IPv4 addressing or IPv6 addressing.

We further validate our model by porting all *test.cc* and *test-suite.cc* files of existing OLSR model to our proposed OLSR model. All the tests pass successfully, thereby, proving the correctness of our implementation. The source code of our proposed OLSR model can be found here [1]. The above mentioned example program and test suites are also provided in the same repository.

[1]https://github.com/adeepkit01/OLSR6

```
Node: 0, Time: +30.0s, Local time: +30.0s, Ipv6ListRouting table
  Priority: 10 Protocol: ns3::olsr6::RoutingProtocol
Node: 0, Time: +30.0s, Local time: +30.0s, OLSR6 Routing table
Destination                NextHop                Interface        Distance
2001:1::200:ff:fe00:2      2001:1::200:ff:fe00:2          1               1
2001:2::200:ff:fe00:3      2001:1::200:ff:fe00:2          1               2
2001:2::200:ff:fe00:4      2001:1::200:ff:fe00:2          1               1
2001:3::200:ff:fe00:5      2001:1::200:ff:fe00:2          1               2
2001:3::200:ff:fe00:6      2001:1::200:ff:fe00:2          1               1
2001:4::200:ff:fe00:7      2001:1::200:ff:fe00:2          1               2
2001:4::200:ff:fe00:8      2001:1::200:ff:fe00:2          1               3
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv6StaticRouting
Node: 0, Time: +30.0s, Local time: +30.0s, Ipv6StaticRouting table
Destination                Next Hop                         Flag Met Ref Use If
::1/128                    ::                               UH   0   -   -   0
fe80::/64                  ::                               U    0   -   -   1
2001:1::/64                ::                               U    0   -   -   1
```

(a) Node 0

```
Node: 1, Time: +30.0s, Local time: +30.0s, Ipv6ListRouting table
  Priority: 10 Protocol: ns3::olsr6::RoutingProtocol
Node: 1, Time: +30.0s, Local time: +30.0s, OLSR6 Routing table
Destination                NextHop                Interface        Distance
2001:1::200:ff:fe00:1      2001:2::200:ff:fe00:4          1               2
2001:1::200:ff:fe00:2      2001:2::200:ff:fe00:4          1               1
2001:2::200:ff:fe00:4      2001:2::200:ff:fe00:4          1               1
2001:3::200:ff:fe00:5      2001:2::200:ff:fe00:4          1               2
2001:3::200:ff:fe00:6      2001:2::200:ff:fe00:4          1               1
2001:4::200:ff:fe00:7      2001:2::200:ff:fe00:4          1               2
2001:4::200:ff:fe00:8      2001:2::200:ff:fe00:4          1               3
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv6StaticRouting
Node: 1, Time: +30.0s, Local time: +30.0s, Ipv6StaticRouting table
Destination                Next Hop                         Flag Met Ref Use If
::1/128                    ::                               UH   0   -   -   0
fe80::/64                  ::                               U    0   -   -   1
2001:2::/64                ::                               U    0   -   -   1
```

(b) Node 1

```
Node: 2, Time: +30.0s, Local time: +30.0s, Ipv6ListRouting table
  Priority: 10 Protocol: ns3::olsr6::RoutingProtocol
Node: 2, Time: +30.0s, Local time: +30.0s, OLSR6 Routing table
Destination                NextHop                Interface        Distance
2001:1::200:ff:fe00:1      2001:1::200:ff:fe00:1          1               1
2001:2::200:ff:fe00:3      2001:2::200:ff:fe00:3          2               1
2001:3::200:ff:fe00:5      2001:3::200:ff:fe00:5          3               1
2001:4::200:ff:fe00:7      2001:3::200:ff:fe00:5          3               1
2001:4::200:ff:fe00:8      2001:3::200:ff:fe00:5          3               2
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv6StaticRouting
Node: 2, Time: +30.0s, Local time: +30.0s, Ipv6StaticRouting table
Destination                Next Hop                         Flag Met Ref Use If
::1/128                    ::                               UH   0   -   -   0
fe80::/64                  ::                               U    0   -   -   1
2001:1::/64                ::                               U    0   -   -   1
fe80::/64                  ::                               U    0   -   -   2
2001:2::/64                ::                               U    0   -   -   2
fe80::/64                  ::                               U    0   -   -   3
2001:3::/64                ::                               U    0   -   -   3
```

(c) Node 2

```
Node: 3, Time: +30.0s, Local time: +30.0s, Ipv6ListRouting table
  Priority: 10 Protocol: ns3::olsr6::RoutingProtocol
Node: 3, Time: +30.0s, Local time: +30.0s, OLSR6 Routing table
Destination                NextHop                Interface        Distance
2001:1::200:ff:fe00:1      2001:3::200:ff:fe00:6          1               2
2001:1::200:ff:fe00:2      2001:3::200:ff:fe00:6          1               1
2001:2::200:ff:fe00:3      2001:3::200:ff:fe00:6          1               2
2001:2::200:ff:fe00:4      2001:3::200:ff:fe00:6          1               1
2001:3::200:ff:fe00:6      2001:3::200:ff:fe00:6          1               1
2001:4::200:ff:fe00:8      2001:4::200:ff:fe00:8          2               1
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv6StaticRouting
Node: 3, Time: +30.0s, Local time: +30.0s, Ipv6StaticRouting table
Destination                Next Hop                         Flag Met Ref Use If
::1/128                    ::                               UH   0   -   -   0
fe80::/64                  ::                               U    0   -   -   1
2001:3::/64                ::                               U    0   -   -   1
fe80::/64                  ::                               U    0   -   -   2
2001:4::/64                ::                               U    0   -   -   2
```

(d) Node 3

```
Node: 4, Time: +30.0s, Local time: +30.0s, Ipv6ListRouting table
  Priority: 10 Protocol: ns3::olsr6::RoutingProtocol
Node: 4, Time: +30.0s, Local time: +30.0s, OLSR6 Routing table
Destination                NextHop                Interface        Distance
2001:1::200:ff:fe00:1      2001:4::200:ff:fe00:7          1               3
2001:1::200:ff:fe00:2      2001:4::200:ff:fe00:7          1               2
2001:2::200:ff:fe00:3      2001:4::200:ff:fe00:7          1               3
2001:2::200:ff:fe00:4      2001:4::200:ff:fe00:7          1               2
2001:3::200:ff:fe00:5      2001:4::200:ff:fe00:7          1               1
2001:3::200:ff:fe00:6      2001:4::200:ff:fe00:7          1               2
2001:4::200:ff:fe00:7      2001:4::200:ff:fe00:7          1               1
 HNA Routing Table: empty
  Priority: 0 Protocol: ns3::Ipv6StaticRouting
Node: 4, Time: +30.0s, Local time: +30.0s, Ipv6StaticRouting table
Destination                Next Hop                         Flag Met Ref Use If
::1/128                    ::                               UH   0   -   -   0
fe80::/64                  ::                               U    0   -   -   1
2001:4::/64                ::                               U    0   -   -   1
```

(e) Node 4

Figure 4: **IPv6 Routing Table of all nodes**

5. CONCLUSION AND FUTURE WORK

In this paper, we have successfully implemented and validated the proposed model of OLSR protocol supporting IPv6 addressing. The detailed account of the changes made in order to make OLSR compatible with IPv6 addressing has been provided. We look forward to submit our code for review and subsequently merge it in the mainline of ns-3.

6. REFERENCES

[1] P.Minet A. Qayyum A. Laouiti T. Clausen P. Jacquet, P. Muhletaler and L. Viennot. Optimized link state routing protocol, 2003.

[2] Network Simulator 3. https://www.nsnam.org/, 2011. [Online].

[3] A. Laouiti A. Qayyum and L. Viennot. Multipoint relaying technique for flooding broadcast messages in mobile wireless networks, 2002.

[4] R. Hinden and S. Deering. Ip version 6 addressing architecture, 1998.

[5] S. Deering and R. Hinden. Internet protocol, version 6 (ipv6) specification, 1998.

[6] R. Hinden and S. Deering. Ipv6 multicast address assignments, 1998.