Test Bed Simulation for Mobile Ad Hoc Routing Protocol: An On Demand Vector Routing Algorithm Case Study

Jiwa Abdullah[1], Hannes Masandig[1]

[1]Faculty of Electric and Electrical Engineering,
Universiti Tun Hussein Onn Malaysia

Abstract: The developments of routing protocol optimization for highly dynamic topology network such as Mobile Ad Hoc Network (MANET) and Vehicular Ad Hoc Network (VANET) is key challenges in establishing mobile ad hoc communication and intelligent transportation system. This report presents an ad hoc network routing protocol developments using intelligent classification on demand vector routing algorithm test bed; using OMNET++ simulation analysis tools. The simulation runs several nodes of 10, 20, 40 and 80s. The QoS performance measurements metrics obtained using an On-Demand Distance Vector (AODV) routing protocol. The results obtained are to be used for accessing the performance of proposed algorithm.

Keywords: Ad hoc networks, MANET, VANET, Intelligent transportation system, OMNET++ simulation

1. INTRODUCTION

This report presents a test bed simulation designs and implementation of a Mobile Ad-hoc Network routing protocol algorithm for Ad hoc On-Demand Distance Vector [1] (AODV) using OMNET++ [2, 3] simulation environments. The simulation designed for running several different numbers of nodes; 10, 20, 40 and 80s. And evaluate their communication performance based on QoS [4] measurements metrics. The simulation procedure uses C++ and Ned language

2.Mobile Ad hoc network (MANET)

A mobile ad-hoc network [5] (MANET) is a self-configuring non-infrastructure network of mobile devices connected by wireless communication channel. Device in MANET is mobile in any direction, and therefore network topology will change frequently. All devices must forward traffic to other device, therefore devices could also be a router at the same time. The primary challenge in designing MANET network is that each devices has to continuously maintain the information required to properly route traffics. They may be in a networks that communicate among themselves or possibly connected into a larger networks like the Internet. MANETs uses the Open System Interconnection (OSI) model's layer higher than the Data Link Layer for the routing process implementation. Many academic papers propose and evaluate communication protocols, taking into accounts varying degrees of mobility on a bounded space, number of nodes and signal hops. Various different protocols are then evaluated based on Quality of Service measurements such as (a) packet drop rate, (b) the overhead (introduced by the routing protocols), (c) end-to-end packet delays and (d) network throughput.

2.1 End to end transmission Delays

MANET network is a packet switching transmission system; thus experience delays. Transmission delay [6] happened when a receiving devices has to wait for the whole packet to arrived and send it after an error check has been done. Also call store-and-forward delay where some amount of time required pushing all of the packet's bits into the wire. In other words, this is the delay caused by the data-rate of the link. Transmission delay is a function of the packet's length and has nothing to do with the distance between the two nodes. This delay is proportional to the packet's length in bits.

It is given by the following formula:

$$D_T = \frac{N}{R} \tag{11}$$

Where:

D_T is the transmission delay,
N is the number of bits, and
R is the rate of transmission (in bits per second)

Most packet switched networks use store-and-forward transmission at the input of the link. A switch using store-and-forward transmission will receive (save) the entire packet to the buffer and check it for CRC errors or other problems before sending the first bit of the packet into the outbound link. Thus store-and-forward packet switches introduce a store-and-forward delay at the input to each link along the packet's route.

2.2 Packet Delivery Ratio (PDR)

The packet delivery ratio [7] for ad hoc packet transferred calculated as the number of data packets received by the destination nodes divided by the number of data packets transmitted by the source nodes. The formula for PDR is;

$$PDR = \frac{Rcv}{Tx} \quad (2)$$

Where:

Rcv is the number of bits received at the destination, and

Tx is the number bits transmitted from the source

2.3 Routing Overhead

Routing Overhead is measured as the number of control packet transmissions counting every hop) per data packet delivered. This overhead can be represents as;

$$O = \sum_{i=0}^{n} O_i \quad (3)$$

Where:-

O_i is the overhead for i node

3. ROUTING PROTOCOL - AD HOC ON-DEMAND DISTANCE VECTOR (AODV)

Many routing protocols have been designed by researchers. The most well known routing protocol are DSDV and AODV and this report chosen AODV to be the test bed simulation.

Ad hoc routing presents challenges due to high mobility and lack of topology information at each client. We have two categories of ad hoc routing:-

a) Pro-active (Table-driven) Protocols -These algorithms maintain fresh list of destinations and their routes at each node. eg. Destination-Sequenced Distance-Vector (DSDV)
b) Reactive (On-demand) Protocols - this protocol find a route when needed. e.g. Dynamic Source Routing (DSR), Ad Hoc On-Demand Distance Vector Routing (AODV), Temporally Ordered Routing Algorithm (TORA)

3.1 Ad hoc On-Demand Distance Vector (AODV) protocol algorithm

AODV Routing is a routing protocol for mobile ad hoc networks (MANETs) and other wireless ad-hoc networks. It is jointly developed in Nokia Research Center, University of California, Santa Barbara and University of Cincinnati by C. Perkins, E. Belding-Royer and S. Das.

It is a reactive routing protocol, meaning that it establishes a route to a destination only on demand. In contrast, the most common routing protocols of the Internet are proactive, meaning they find routing paths independently of the usage of the paths. AODV is, as the name indicates, a distance-vector routing protocol. AODV avoids the counting-to-infinity problem of other distance-vector protocols by using sequence numbers on route updates, a technique pioneered by DSDV. AODV is capable of both unicast and multicast routing.

3.2 The AODV Algorithm Working Procedure

In AODV, the network is silent until a connection is needed. At that point the network node that needs a connection broadcasts a request for connection. Other AODV nodes forward this message, and record the node that they heard it from, creating an explosion of temporary routes back to the needy node. When a node receives such a message and already has a route to the desired node, it sends a message backwards through a temporary route to the requesting node. The needy node then begins using the route that has the least number of hops through other nodes. Unused entries in the routing tables are recycled after a time. When a link fails, a routing error is passed back to a transmitting node, and the process repeats.

Much of the complexity of the protocol is to lower the number of messages to conserve the capacity of the network. For example, each request for a route has a sequence number. Nodes use this sequence number so that they do not repeat route requests that they have already passed on. Another such feature is that the route requests have a "time to live" number that limits how many times they can be retransmitted. Another such feature is that if a route request fails, another route request may not be sent until twice as much time has passed as the timeout of the previous route request.

The advantage of AODV is that it creates no extra traffic for communication along existing links. Also, distance vector routing is simple, and doesn't require much memory or calculation. However AODV requires more time to establish a connection, and the initial communication to establish a route is heavier than some other approaches.

4. SIMULATION

4.1 Simulation Process Flow

The simulation is according to following process steps;

a) Simple Module Component preparation
b) The sub-module component preparation
c) The network management's module preparation
d) Running simulations

4.2 Simulation components

Various components are to be ready before the simulation has to run. The following are the required components;

a) Network topology or network designs
b) The Message Manager or the Routing Algorithm
c) The network components

4.3 Simulations Implementation

The routing protocol is evaluated based on the three performances metric which is Packet Delivery Fraction, End-to-End Delay and the Routing Overhead. The simulation environments for this scenario are:-

- Various number of node which are 10, 20, and 30 nodes
- Packet size is set to 1400 Bytes
- Area size is set to 1000 x 1000 m^2 flat area
- Node Speed is fixed to 20 m/s
- Random Way Point mobility model is used

4.3.1 Building and running simulations

This section provides insight into working with OMNeT++ in practice: Issues such as model files, compiling and running simulations are discussed.

An OMNeT++ model consists of the following parts:

- NED language topology description(s) (.nedfiles) which describe the module structure with parameters, gates etc. NED files can be written using any text editor or the GNED graphical editor.
- Message definitions (.msgfiles). You can define various message types and add data fields to them. OMNeT++ will translate message definitions into full-fledged C++ classes.
- Simple modules sources. They are C++ files, with.h/.ccsuffix.

The simulation system provides the following components:

- Simulation kernel. This contains the code that manages the simulation and the simulation class library. It is written in C++, compiled and put together to form a library (a file with .a or .lib extension)
- User interfaces. OMNeT++ user interfaces are used in simulation execution, to facilitate debugging, demonstration, or batch execution of simulations. There are several user interfaces, written in C++, compiled and put together into libraries (.aor.libfiles).
- Simulation programs are built from the above components. First, the NED files are compiled into C++ source code, using the NEDC compiler which is part of OMNeT++. Then all C++ sources are compiled and linked with the simulation kernel and a user interface to form a simulation executable.

4.4 Simulation Result

The result of the simulation is shown in Table 1, and the graphical depictions are shown in figure 1 to 4.

Table 1: Simulation Results Average of Quality metrics

Number of nodes	10	20	30	80
Packet Delivery Ratio (PDR)	0.77	0.51	0.48	Na
End-to-end Delay	0.05	0.085	0.055	0.065
Routing Overhead (RO)	55000	61000	78000	Na

End to End Transmission Delay (second)

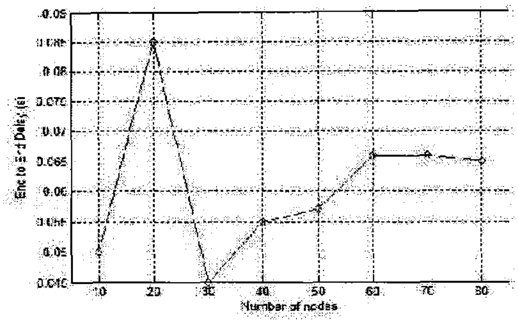

Figure1: End to end Delay (s)

Packet Delivery Ratio (PDR)

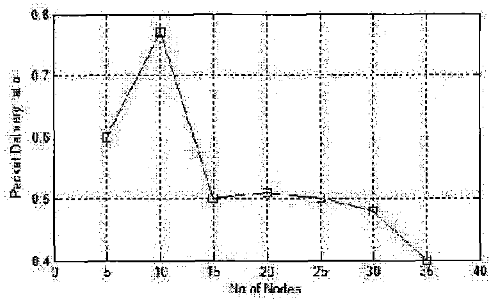

Figure 2: Packet Delivery Ratio

Routing Overhead

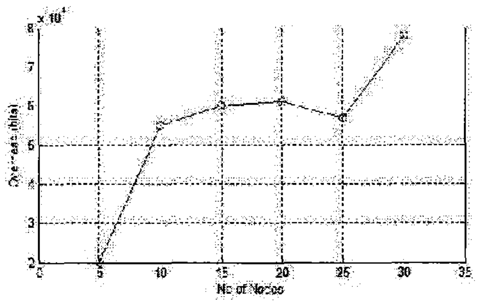

Figure 3: Routing Overhead

Routing Overhead with various node mobility speed

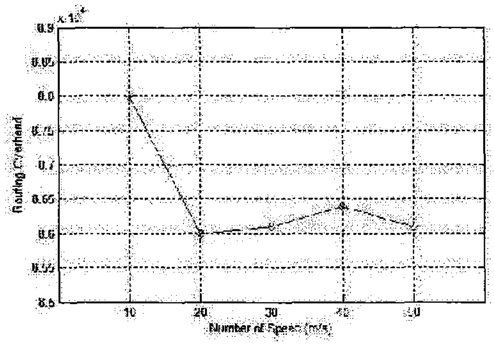

Figure 4: Routing overhead for various speed of node mobility

5. CONCLUSIONS

The optimizations of routing protocol algorithms for ad hoc network are still under study by many researchers. This is true particularly for a mobile ad hoc routing optimization. This test bed study; explore the AODV protocol algorithm performance. The study has been conducted to reveal several quality performance metrics mainly on delivery delay, packet delivery ration and on routing overhead. The results obtained here however are worthless unless a comparative study of various recent routing algorithm schemes has been conducted. This effort however important for the next progress of the study in which more algorithms are to be study in order to propose better routing algorithm techniques.

References

[1] Paul, A.B., et al. *Implementation and performance evaluation of AODV in Wireless Mesh Networks using NS-3*. 2010: IEEE.

[2] Sommer, C., I. Dietrich, and F. Dressler, *Simulation of ad hoc routing protocols using OMNeT++*. Mobile Networks and Applications, 2010. **15**(6): p. 786-801.

[3] Varga, A. and R. Hornig. *An overview of the OMNeT++ simulation environment*. 2008: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[4] Abdullah, J., et al. *GA-Based QoS Route Selection Algorithm for Mobile Ad-Hoc Networks*. 2008: ACM.

[5] Sarkar, N.I. and W.G. Lol. *A study of MANET routing protocols: Joint node density, packet length and mobility*. 2010: IEEE.

[6] Murthy, S., P. Hegde, and A. Sen. *Design of a delay-based routing protocol for multi-rate*

multi-hop mobile ad hoc networks. 2009: IEEE.

[7] Mohamed, H., et al., *Identifying Factors Affecting Data Delivery Performance in Mobile Ad-Hoc Network Routing Protocol Using A Systematic Approach 1*. 2008.