

Simulation-based approach to performance study of routing protocols in MANETs and ad-hoc Networks

Thodeti Srikanth¹ Dr.V.B.Narsimha²

¹Lecturer in Computers Govt.Degree College for Women, Jagtial

²Asst.Professor Dept of Cse University College of Engineering Osmania University, Hyderabad

Abstract:

Mobile ad-hoc network (MANET) is a wireless network without infrastructure. Nodes can communicate each other without central infrastructure; because they are self organized and self configurable with easy deployment. To communicate each other it is required efficient routing protocols in MANET technology. In these we find out an efficient routing protocols for routing, and we had considered different approaches like routing load, end to end packet delivery and performance of protocols. To analyze this we implemented two different routing protocols which are proactive and reactive. To analyze these aspects we used HTTP high and lower load traffic. And we conclude the efficiency of a network can be achieved by choosing the best suitable protocols based on the network requirement.

1 Introduction

Proactive and REACTIVE routing protocols can be implemented by using OPNET IT Guru Academic Edition simulator 9.14. OPNET simulator is best simulator for networking research in graphical mode using windows xp. It provides substantial support for C++ routing and multicast protocols over wired and wireless networks. Using GRAPH we can create graphical results for the project. All this work is done in either in Linux or Windows.

2. OPNET IT Guru Academic Edition simulator

The section describes how to design and implementation of performance comparison of PROACTIVE and REACTIVE routing protocol. The performance is compared with by using average end to end delay and packet loss and packet delivery fraction matrices [1]. The value of simulation in studies of protocols is that it allows near perfect experimental, experiments can be designed at will and then rerun while varying an experimental variable and holding all other variables constant [2][3]. With this simulation it is also possible to test the behavior of networks with more nodes than physical equipment is available for, or networks with equipment that does not even exist yet.

OPNET simulator is one of the graphical mode simulators, which is user friendly. In OPNET no need to do any programming, the simulator which provides drag and drop facility. When we drag and drop and create a scenario simulator will automatically generate program in the background. 1) Detail simulation of protocols requires a systems programming language which effectively manipulates of bytes and packet headers and implementing algorithm that runs large over data sets. For these runtime speed is important. But fixing bug, recompile, rerun having less important. 2) In large network depends on the network the configuration and steps will change. In that scenario iteration time had impotent like rerunning simulation and changing the model. Since the configuration changes once runtime takes less important. OPNET can manage all these in the background. OPNET Modeler [2] [3] was chosen as a simulation environment because it is one of the leading environments for network modeling and simulation. It supports large number of built-in industry standard network protocols, devices, and applications. In addition, its programming library helps researchers to easily modify the network elements and measure their performance in the simulation environment. OPNET also provides rich data analysis features.

3 OPNET architecture

OPNET provides a comprehensive environment to model and do performance evaluation of networks and distributed systems. The OPNET package includes numbers of tools. Those tools fall into three categories corresponding to the three phases of modeling and simulation projects: Specification, Simulation and Data Collection, and Analysis. These phases should necessarily be in sequence and form a simulation cycle as in Figure 1.

OPNET uses the concept of modeling domains to represent its modeling environments, and graphical editors for editing the Network, Node and Process models. Specifically, there are several editors in OPNET: project editor, node editor, process editor, external system editor, link model editor, packet format editor, Interface Control Information editor, and probability density function editor etc.

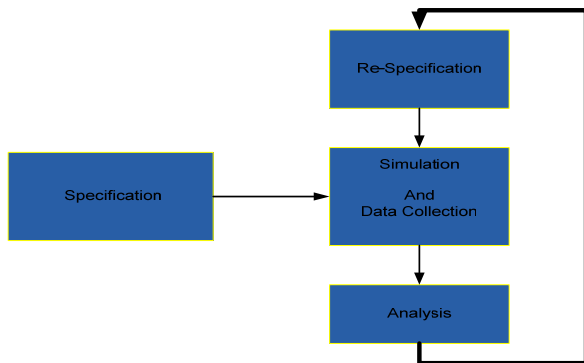


Figure 1: Simulation Cycle in OPNET

Network Domain is used to define the network topology of a communication network. The communicating entities are called nodes. Network domain is created by using the Project editor tool of the OPNET modeler. Node Domain describes nodes' internal architecture in terms of functional elements in the node and data flow between them. Process defines the behavior of processes, including protocols, algorithms and application, specified using infinite state machines and an extended high-level language. External System specifies the interfaces to the models provided by other simulators running concurrently with an OPNET simulation (a co-simulation).

4 OPNET Modeler Wireless Support:

The Wireless module in OPNET provides a flexible and scalable wireless network modeling environment, including a broad range of powerful technologies. The Wireless module integrates OPNET's full protocol stack modeling capability, including MAC, routing, higher layer protocols, and applications, with the ability to model all aspects of wireless transmissions, including the following:

- Radio Frequency propagation i.e. path loss with terrain diffraction, fading, and atmospheric and foliage attenuation
- Interference
- Transmitter/receiver characteristics
- Node mobility, including handover
- Interconnection with wire-line transport networks

The wireless module has rich protocol model suites to optimize the R&D processes, and more effectively design technologies such as MANET, 802.11, Ultra Wide Band, 802.16, Bluetooth, and Transformational Communications systems. Wireless network planners, architects, and operations professionals can analyze end-to-end behavior,

tune network performance, and evaluate growth scenarios for revenue-generating network services.

5. Implementing the Protocols in the OPNET Modeler

In this paper, I have implemented two secure routing protocols DSDV, DSR, AODV, ADR and TORA, in the OPNET Modeler simulation environment, using the Application Programming Interface [2] functions of the OPNET development kit and the embedded C language. The malicious feature of a wireless node is integrated into the routing protocol model, so that each wireless node can be easily switched back and forth between the normal mode and the malicious mode.

We can use the C/C++ language to implement/modify the behavior of a module. For easy development, OPNET provides quite a large library with over 400 predefined functions and procedures [2].

Figure 2 shows steps to add new secure routing protocols reactive and proactive into the OPNET Modeler. Reactive and proactive are respectively which are supported in OPNET, so I did not write any code for this. Which are available I implemented.



Figure 2: Steps to add new secure routing protocols into OPNET

In Figure 2 is further concretized in Figure 3. At the origin nodes that generate the routing packets, the routing packets sent the routing process. These conditions are defined by each specific protocol and added at the processing phase of the routing process.

6. DATA Metrics for evaluation

Quality analysis can be done, how much data is transferred from source node to destination node, and how many packets dropped during process and routing load calculated during the packet transformation from source node to destination node.

6.1 Packet Delivery Fraction (PDF):

This is the output of total number of received data packets divided by total number of sent data packets.

$$\text{PDF} = \frac{\text{Number of Received Data Packets}}{\text{Number of Sent Data Packets}}$$

This metric gives an estimate of how efficient a routing protocol is, since the number of routing packets sent per data packet gives an idea of how well the protocol keeps the routing information updated. The higher the routing metric is, the higher the overhead of routing packets and consequently the lower the efficiency of the protocol.

6.2 Routing Load:

This is the total number of routing packets sent divided by the total number of data packets received. This accounts for the overhead of the routing protocols. The number of total routing packets includes the number of route request packets (RREQ), route reply packets (RREP), route error packets (RERR), acknowledgement packets, hello protocol packets, etc [4] [5].

6.3 Project Scenario setup

In designing the model initially, we have to run OPNET modeler and then to create a blank scenario from the start-up-wizard. In this wizard we have to choose type of propose we are choosing next step to choose the area of the size, soon we will see the work space. Here in the work space we will design our network by using the required network entities for our design. Design can be done either manually or through wizard. All the required entities such as Application configuration, Profile configuration, Mobility configuration, Server, Nodes these entities will be taken from the object palette to our project work space. Which is shown in the figure 3

6.4. Application configuration:

Application configuration is used to specify/choose the required application among the available applications such as FTP, HTTP, Email, Database, Print etc. We can create a name for our choice and give the relevant description in creating new application. In this thesis we have used two web application HTTP performing heavy browsing and light browsing.

6.5. Profile configuration:

Profile configuration will be used to create user profiles these profiles can be specified on different nodes in network designed to generate the application traffic. While configuring profiles applications that are defined in the application configuration are used. In this thesis we have created two profiles HTTP Heavy and HTTP Light based on the applications we chosen in application configuration with this profiles we can restrict the nodes to a assigned profile based on user design requirement.

6.6. Mobility configuration:

Mobility configuration is used to specify mobility model to the nodes in the network and provides parameters that will control the movement of nodes such as speed, start time, stop time etc. In this thesis we have chosen 10 meters/sec and 28 meters/sec speeds respective in based on our simulation scenario requirement, start time at 10 sec, mobile nodes have a trajectory of movement 500 meters max and we have not taken the pause time into consideration. We have chosen random waypoint mobility as this will make sure that mobile nodes are configured with mobility it is widely used mobility model in Ad hoc networks.

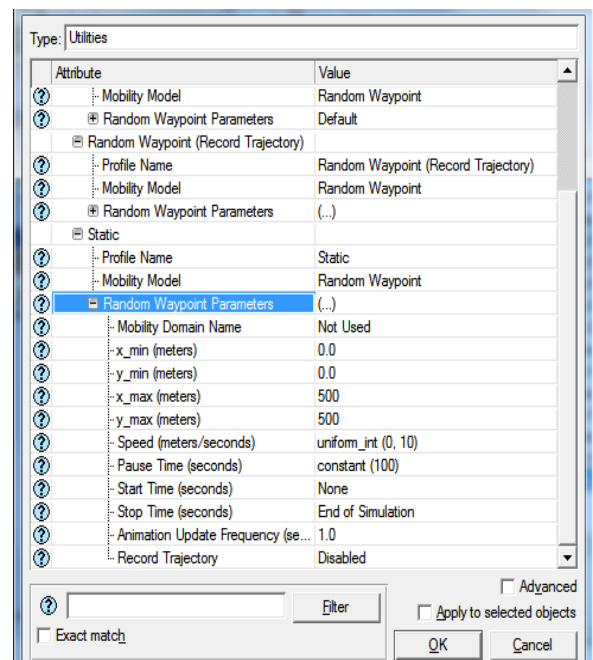


Figure 3 mobility configuration

Server: It is a WLAN server mobile node with applications running over TCP it will support one underlying IEEE 802.11 and its connection at 2 Mbps. In server we can define the supported services based on the user profiles that may support FTP, Email and HTTP etc on the client. In our thesis we have set HTTP as supported services in the server as our clients relying on the profiles HTTP.

Nodes: Nodes are work stations with client server applications running over TCP/IP and UDP/IP this will support underlying WLAN connection at 1 Mbps, 2 Mbps, 5.5 Mbps and 11 Mbps. Here for the nodes we can define the trajectory in designing the model as per the requirement for our model design we have chosen VECTOR trajectory. And we 24 can assign the routing

protocol to be used by the nodes for routing. In our thesis we have used DSR, OLSR and AODV routing protocols. We can assign the start time and end time in our design start time is chosen to start at the start of simulation and end time is chosen to end at the end of the simulation. We can specify the mobility profile defined in the mobility configuration to model the mobility over the nodes. In our design we have specified random way point mobility model. Generally mobile nodes engaged in a network move randomly and takes random destinations, moreover random mobility model is more appropriate for simulation studies.

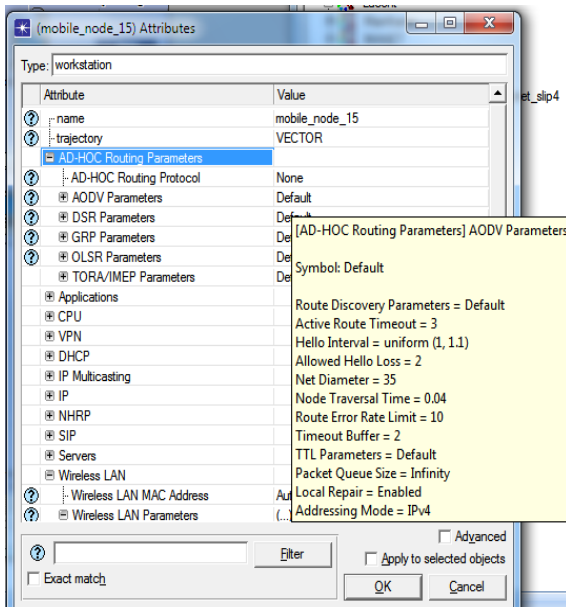


Figure 4.node Attributes

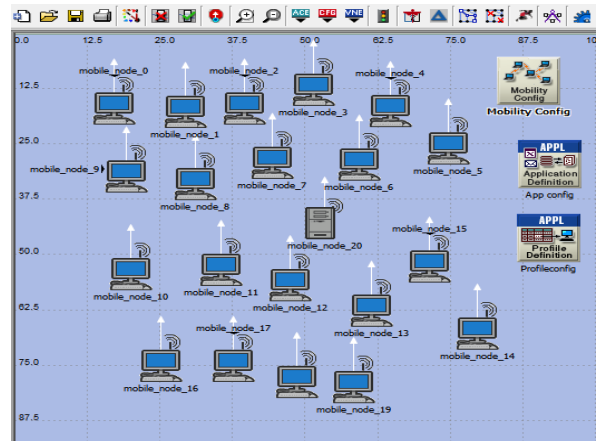


Figure 5: Network setup for the experiment in OPNET simulator

Applying scenario

To specify the scenario that has to be collected during a discrete event simulation we have to choose from the, chose individual DES statistics can be found in workspace pop-up menu. Statistics need to be applied for a designed model is basically two type's

- 1) Global or scenario-wide statistics
- 2) Object statistics.

Global statistics will be collected from the whole network model designed and the object statistics will be collected over nodes. These statistics can be applied to a network model based on the user requirement for his design. In our design for our analysis we have chosen global statistics of Wireless LAN that include delay, throughput and TCP delay in analyzing the performance of the protocols chosen on our modeled scenarios. In the TCP information like delay of packets parsec and routing load will be specified.

7. Conclusion:

By using this methodology we can find out the performance of REACTIVE and PROACTIVE protocols. This paper discusses the simulation-based approach to performance study of routing protocols in MANETs and ad-hoc. It gives an introduction to the OPNET Modeler simulator and provided supports. The chapter also discusses the general approach that is used to conduct the research in the OPNET environment. And also describes a set of scenarios and metrics that are used to analyze the performance of the protocols in malicious as well as baseline environments. In the next chapter, we will discuss the results of the experiments through those defined metrics.

Table 1: Malicious node assignment

Parameters	Values
Number of nodes	10,20
Area size	10 X 10 m
Simulation time	10min
No of topology	4
Node type	MANET
Physical media	502.11
Data rate	11MB
Type of traffic	CBR
Maximum speed	5 M/S
Packet size	512 bytes
Packet rate	5 packets per second

References:

- [1] S. Das, C. Perkins, and E. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In INFOCOM'2000 (1), pages 3–12, 2000.
- [2] OPNET Modeler 11.0. <http://www.opnet.com/products/modeler/home.html>
- [3] Glomosim – Global Mobile Information Systems Simulation Library <http://pcl.cs.ucla.edu/projects/glomosim/>
- [4] Kimaya Sanzgiri, Bridget Dahill, Brian N. Levine, and Elizabeth M. Belding-Royer. “A Secure Routing Protocol for Ad Hoc Networks”. Proceedings of 10th IEEE International Conference on Network Protocols (ICNP'02), Paris, France, November 2002, pp. 78-90. <http://prisms.cs.umass.edu/brian/pubs/aran.icnp02.pdf>
- [5] Yi-an Huang and Wenke Lee. “Attack analysis and Detection for Ad-hoc Routing protocols”. Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID'04), French Riviera, France. September 2004. <http://www.ece.cmu.edu/~adrian/731-sp05/readings/Huang-Lee-Ad-Hoc-Net-IDS.pdf>