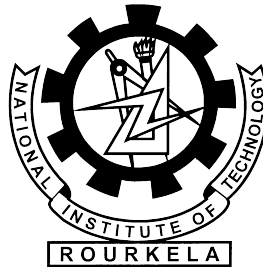


NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA



Design of a Simulator for finding K-best Energy Efficient Paths in Mobile Ad-hoc Networks

by

Purushottam Swami

under the guidance of

Prof. S. Chinara

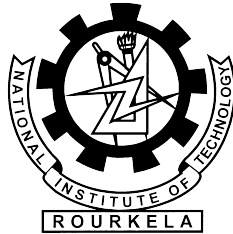
A thesis submitted in partial fulfillment for the
degree of Bachelor of Technology

in the

Department of Computer Science and Engineering

May 2013

NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA



Certificate

This is to certify that the thesis entitled, 'DESIGN OF A SIMULATOR FOR FINDING K-BEST ENERGY EFFICIENT PATHS IN MOBILE AD HOC NETWORKS' by Purushottam Swami in partial fulfilment of the requirements for the award of Bachelor of Technology Degree in Computer Science and Engineering at the National Institute of Technology, Rourkela is an authentic work carried out by him under my supervision and guidance. To the best of my knowledge the matter embodied in the thesis has not been submitted to any other University/ Institute for the award of any Degree or Diploma.

Date:

Prof S. Chinara

Abstract

MANET stands for Mobile ad hoc Network. A MANET is an infrastructure less network where every node acts as a router as well as a normal node. The nodes in the network are self-configuring, ie. they configure themselves to route the packets with the changing topology. The movement of the nodes in the network is frequent so that the topology leads to disruption in the node connectivity and the end to end routing path. Simulator is a software that provides a complete visibility of the network and the contents of the network. In the current work a simulator is designed to show the dynamic movement of nodes by showing their changing neighbors with respect to the node mobility in the network. The path that is established between the source and target node is also visible by tracing path between them. The routing path between the source and destination also changes with changing topology and changing scenario of the network.

There exists several parameters for designing the simulator of a mobile ad hoc network. For the current work we have considered some of the parameters such as node id (unique), node position (x, y), direction of movement, speed, state (moving/stationary), pause time between two consecutive movements, and node battery power. To start with the simulator, we created a number of nodes by assigning their random initial positions in the network. The node direction of movement, pause time between the movements for all nodes are also chosen randomly. Every node continues to move in this direction with an assigned speed till pause time expires. After time-out nodes find a new pause time. After a time-out or reaching at the boundary of the system node gets a new direction to move forward. All the nodes got same transmission range for the network. Node calculates Euclidean distance with remaining nodes and if the distance is less than or equal to neighborhood distance than it assigns them as its neighbor. So a node can be neighbor for more than one node. Source node finds and k-alternate paths. For finding best k-alternate energy efficient paths we used Yen's k-algorithm. It includes Dijkstra's algorithm for best path that is path with sum of minimum battery power of the nodes included in the path. For Dijkstra's weight parameter it uses sum of the battery power of two nodes as edge weight. For Yen's k-algorithm: it first finds a best path using Dijkstra's algorithm and then it removes the edges one by one from the graph(topology) and finds next path that is included in the path that found out first. And then finds next best path using same Dijkstra's algorithm. Appends this new path to the k-alternate paths and restore the edge that was removed. Next it removes the next edge in the graph and continues till no more paths exist or k-paths found.

As battery power of nodes is consumed while transmitting and/or in idle mode. As MANETs are multi-hop, so there are chances of involvement of some intermediate nodes other than source and target node. The routing algorithm decides which intermediate nodes will be selected for transmission. Some conventional algorithms like AODV do

not take care of energy in nodes to determine the routing path. Some nodes may die out soon, as they are used in most of the packet transmission paths and, on the contrary, few nodes may not be used for a long time. This imbalance of energy consumption leads to unreliability of the network.

Keywords: Mobile ad hoc networks, Energy Efficiency, Alternate paths.

Acknowledgements

This project would not have been possible without the help and support of many. I would like to express my gratitude to Prof. Suchismita Chinara for her advice during our project work. As my supervisor, she has constantly encouraged me to keep on focused on achieving goal. Her vast knowledge and expertise in the area of networking was immensely helpful. Her observations and comments helped me to establish to overall direction of the research and to move forward with the study in depth. She has helped us greatly and been a source of knowledge.

We are thankful to all our teachers and friends. Our sincere thanks to everyone who has provided us with inspirational words, a welcome ear, new ideas, constructive criticism, and their invaluable time, we are truly indebted. We must acknowledge the academic resources that we have acquired from NIT Rourkela. We would like to thank the administrative and technical staff members of the department who have been kind enough to advise and help in their respective roles.

Purushottam Swami
(109CS0564)
Department of Computer Science and Engineering
National Institute of Technology
Rourkela

Contents

Certificate	i
Abstract	ii
Acknowledgements	iv
List of Figures	vi
1 Introduction	1
1.1 MANET	1
1.2 Application of MANNET	2
1.3 Challenges of MANET	3
2 Literature Survey	5
2.1 Network Simulator	5
2.1.1 NS-2	5
2.1.2 GloMoSim	6
2.1.3 QualNet	7
2.1.4 NETSIM	8
2.1.5 OPNET	8
3 ALGORITHMS, MODELS AND PRORPOSED SIMULATOR	10
3.1 ALGORITHMS AND MODELS	10
3.1.1 Random Way-Point Mobility Model	10
3.1.2 Neighborhood Detection	11
3.1.3 K-best Energy Efficient Alternate Paths	12
3.1.4 Finding the K best paths in a network	14
3.1.5 Power Consumption	15
3.2 PROPOSED SIMULATOR	16
4 Snapshots	18
4.1 Graphical Interface of Simulator	18
4.2 Node Initialization	20
4.3 Neighborhood detection	21
4.4 Finding Best Path	22

4.5	Status of Nodes	23
4.6	Finding K-best Alternate Paths	24
4.7	Energy Consumption Graph	25
4.8	Complete Simulator	26
5	Conclusion	27
	References	28

List of Figures

3.1	Moving pattern of a mobile node using the RWP model	11
3.2	Node movement and finding their neighbors dynamically	12
3.3	Placement of nodes in a network	13
4.1	Graphical Interface	18
4.2	Node Initialization	20
4.3	Neighborhood detection	21
4.4	Finding Best Path	22
4.5	Status of Nodes	23
4.6	Finding K-best Alternate Paths	24
4.7	Energy Consumption Graph	25
4.8	Complete Simulator	26

Chapter 1

Introduction

Over past few years there has been a revolutionary change in the area of telecommunication brought by different kinds of new technology and devices. Networking devices i.e. Laptops, cell phones etc. are getting lesser with increasing mobility and connectivity to internet cloud. In this fashion dependency on hard wired network is no longer of our unease. With The exponential growth of wireless devices the number of wireless internet users is going to exceed than that of the wired line internet users within few years. IEEE 802.11x standards have brought a revolutionary change in the field of wireless networks. To obtain connectivity within a network or to communicate with devices across the network, only a wireless interface or access point is required.

Network devices can get connected across the network or they can obtain mutual connectivity by organizing themselves into ad-hoc networks. According to IEEE 802.11 An ad-hoc network is a network composed of communication devices within mutual transmission range of each other via wireless medium. In modern communication system where there is a requirement of fast and reliable network, devices may be connected through wireless technologies but in some cases wired structure connects several different wireless networks.

There are several kinds of wireless network technologies like WPAN, WLAN and MANET etc.

1.1 MANET

A mobile ad-hoc network (MANET) is a self-configuring network of mobile devices; i.e. it is infrastructure less and is connected by wireless links. In other words A MANET

is a collection of mobile nodes that communicate over somewhat bandwidth controlled links ”.

It is a randomly deployable network where nodes are mobile with changing topology. In the network we call each device as a node and the virtual connectivity among each node is termed as the link. There is a link among the nodes when they are within the transmission range of each other. The nodes are connected in a decentralize fashion where identifying the topology, delivering the message or implementation of any kind of operation is done by each node independently. Nodes organize themselves to send any message or packet to other nodes by creating a multi-hop networking structure. It may be an unconnected network or can be connected to external networks. Introduction of new technologies like HyperLAN, Bluetooth etc., have facilitated to evolve MANET outside its earlier and constrained domain.

IEEE 802.11 family has two main versions that deals with Wireless networks 1. IEEE 802.11a, that works in 5GHz band, provides data rate up to 54Mbps and uses OFDM interface. 2. IEEE 802.11b , operates in 2.4GHz band with data rate 11Mbps and studied in framework of MANET.

The mobile devices can work under IEEE 802.11 standards. They are the in adhoc mode and the Infrastructure mode. Infrastructure mode contains wireless interfaces that further leads to connectivity through wired backbone. Wireless access point is necessary for this kind of structure. It supports high scalability and centralized security but in ad -hoc wireless network mobile nodes directly communicate with each other without the help of intermediate access point. Ad hoc network is also called as the peer to peer network where each node acts as a router along as well as a communicating device. An ad-hoc network can be further connected to internet with the help of gateway nodes.

1.2 Application of MANNET

Mobile Ad-hoc Networks provide self-deployed mode of communication but it leads to the problem of scalability, mobility and energy efficiency. The set of applications of MANET vary from small scale, static and limited power source network to large scale, highly dynamic and mobile networks. MANET can be deployed in any area where there is requirement of economy, fast and direct communication, such as military service, PDAS, Group communication etc. In these areas the network can be established or removed on demand without affecting the infrastructure.

- Military Services: - This is the basic and common application of MANET. In military services there is requirement of highly mobile network where infrastructure

is not necessary, because to setup a highly wired and complex static network may create problem in battlefield, besides soldiers are highly mobile so in order to broadcast any message we need a dynamic network. But in military scenario there is requirement of security, high latency, reliability and recovery from failures etc. So military networks are designed to establish low probability of detection so nodes in ad-hoc network must use less more power and there should be irregularity in message passing that helps with the problem of detection and interference.

- **Disaster management:-** In Disaster situations use of dynamic and peer to peer communication is highly important. MANET provides network with self-configuration and minimum overhead. MANET can be set up on any area with minimum or no infrastructure. In situations like earthquake, flood or any manual or natural destruction such a network can be implemented as it provides high mobility and requires no central connectivity.
- **Group Communication:-** Any shared application running on current internet based scenario is typically based on client/server communication for ex- Instant messaging, server gaming etc. But if a group wants to communicate with another group it passes to the server and then message is passed to another client through that central network but MANET provides P2P communication without centralized approach. It is known as GCA (Group Communication Application). Now IM services that earlier used to depend on centralized operator has become distributed and can be implemented on a dynamic network. If two group wants to communicate they just have to come close enough to get connected through multi hop paths.
- **Personal Area Network:** - it is a small range of network that connects mobile phones, PDAS laptops etc. It can be established as a network to communicate among connected nodes or it can be used to connect a backbone network.

1.3 Challenges of MANET

So far we have discussed about different benefits and applications of MANET but irrespective of its characteristics of easy deployment, no need of infrastructure there are certain challenges one has to face while implementing Mobile Ad-hoc Network.

- **Dynamic topology:-** Nodes are mobile and can be connected dynamically in an arbitrary manner. Links of the network vary timely and are based on the proximity of one node to another node.

- Scalability: - Scalability can be broadly defined as whether the network is able to provide an adequate level of service even in the presence of a large number of nodes. In MANET when the network size increases, number of packets sent by node also increases that leads to drainage of limited battery power and network life time gets reduced thus scalability is major challenging issue.
- Energy Efficiency: - Energy efficiency is the major challenging issue as it affects the network life time and stability.
- Security: - Mobility implies higher security risks such as point-to-point network architecture or a shared wireless medium accessible to both authentic network users and attackers. Eavesdropping, spoofing and denial-of-service attacks are major weaknesses.
- Autonomous: - No centralized administration entity is available to manage the operation of the different mobile nodes.
- Device discovery:- Identifying relevant newly moved nodes and informing about their existence need dynamic ease automatic optimal route selection.
- Topology Maintenance: - Keep informed of the dynamic links among nodes in MANETs is a major challenge.
- Transmission quality: - It is an inherent problem of wireless communication produced by several error sources that result in deprivation of the received signal.

Chapter 2

Literature Survey

2.1 Network Simulator

To observe the performance of network, performance of data communication protocols, node movements, topology changes. Software simulation is however time efficient and cost effective. Simulation is a method where a program models the performance of a network by calculating the interaction between the different network entities (hosts) and their communication among them. The performance of the network and the various applications and services it supports can also be observed; various applications and services it supports can also be observed; various attributes of the environment can also be modified in a controlled manner to assess how the network would behave under different conditions. Simulator is a software that provides a complete visibility of the network and entities of the network. Today, the popularity of network simulators varies from one simulator to another simulator. Every simulator is based on its own models and topologies to simulate a network. While many network simulators are available in the software world but in this topic there will be comprehensive discussion about performance and functionality provided by the discrete event network simulators like ns-2, QualNet, GloMoSim, Netsim and Opnet.

2.1.1 NS-2

The network simulator ns-2 is another discrete event simulator targeting at networking research. It is the abbreviation of Network simulator version two, which is first been developed by 1989 using as the REAL network simulator. Today, NS-2 is supported by Defense Advanced Research Project Agency (DARPA) and National Science Foundation. In NS-2 there is no graphical option but it has GUI for visualizing the simulation run.

Its modules can be written in Scripting language OTcl or in C++. Besides ns-2 already includes a large library of common modules, which are usually sufficient for simulating common wired or wireless networks. This is open source and provides online document. Advantages:-

- NS-2 can support a significant range of protocols in all layers. For ex., the ad-hoc and WSN specific protocols are delivered by NS-2.
- Open source model of NS-2 saves the cost of simulation, and online documents allow the users easily to modify and improve the codes.
- It can run on any system having GNU C-compiler gcc as it is written in C++. Thus it can run on UNIX platform as well.

Limitations:-

- people who want to use this simulator need to familiar with writing scripting language and modelling technique, the Tool Command Language(Tcl) is somewhat difficult to understand and write.
- NS-2 is sometimes more complex and time-consuming than other simulators to model a desired job.
- NS-2 provides a poor graphical support, no Graphical User Interface (GUI), the users have to directly face to text commands of the electronic devices.
- Due to the continuing changing the code base, the result may not be consistent, and contains bugs.
- NS-2 cannot simulate problems of the bandwidth, power consumption or energy saving in WSN., NS-2 has a scalability problem number of nodes cannot be exceeded to 100.

2.1.2 GloMoSim

GloMoSim stand for Global Mobile Simulator. It is a scalable simulator for wireless and wired network systems, developed by PCL (Parallel Calculation Laboratory) of UCLA (University of California in Los Angeles). It uses Parallel Simulation Environment for Complex System (PARSEC) that is a C based simulation language for sequential and parallel execution of Discreet event simulation model. GloMoSim Supports protocol for purely wireless network. Advantages:-

- Scalability of network model can be increased up to thousands of nodes.
- GloMoSim is useful for a network model where nodes are transmitting high amount of information.

Limitations:-

- The updates of GloMoSim are not regular. The last update was in 2003.
- GloMoSim is being designed using the parallel discrete-event simulation capability provided by PARSEC However, the previous version of GloMoSim supports only one sequential simulation.

2.1.3 QualNet

QualNet is a commercial version of GloMoSim. GloMoSim was designed for Mobile Ad-hoc Networks but QualNet supports wider range of networks and analysis i.e. MANET, QoS, Wired Network, Cellular, Satellite etc. It is designed in C++. QualNet developer toolkit contains following component.

- Animator Graphical component used as animation tool.
- Scenario Designer- Graphical, Finite state machine based custom protocol design tool.
- Analyzer Statistical Graphic tool used for built in and custom statistic collection.
- Packet Tracer packet level tracing and visualization.

QualNet provides a graphical interface where simulation can be observed by the help of its animator component. Advantages:-

- QualNet provides GUI tools for protocol modelling.
- Scalability increases via parallel execution.
- It provides a clear built in measurement for each layer.
- Contains standard API for composition of protocol across each layer.

Limitations:-

- Difficult to install on Linux, as it has been built for older version of Linux (Fedora core 4).
- Slow Java-based User Interface.
- Very expensive.

2.1.4 NETSIM

Netsim provides complete model creation and modification capabilities along with graphical interface and data output. Netsim uses the event graph method and object-oriented Programming for simulation. Netsim is a World Wide Web (www) based simulation package written entirely in Java. Netsim allows expandability for complex modelling and integration with other Java-based programs, such as graphical interface and analysis packages. Any simulation problem modelled in NETSIM can be viewed as a web page like applet in Java. Advantages:-

- It is available for all platforms with no recompilation. It can be accessed remotely.
- It provides security by using password mechanism for individual access, and protect against unauthorized change and delicacy.
- High maintenance as frequent modification is instantly distributed. All modification and implementation are made on-sever.
- Encourage communication and interaction through Java and requires Browser capable of running Java programs.

Limitations:-

- In case of web simulation if there is heavy traffic loading time may exceed and thus simulation may get slower.
- It does not have copy of simulation over system, so dependency on increasing network and interruptions may cause inconsistency.
- It is not efficient for a longer run.

2.1.5 OPNET

OPNET stands for Optimized Network Engineering Tools. The OPNET simulator provides an ACE (Application Characterization Environment) module which can be used

to import packet traces into simulation from different sources including TCP DUMP files., It is a high level event based simulator. It works at packet level. Originally it was built for the simulation of static network but now possibilities of wireless simulation are also very rich. OPNET contain high level user interface which is designed in C and C++ code blocks, and with a huge library of OPNET specific methods. It provides a hierarchical structure where modelling can be done in three domain

1. Network Domain- Network topologies, sub-networks, mobility etc. lies in this domain.
2. Node Domain- it deals with single network nodes i.e. router, mobile devices etc.
3. Process Domain- it contains single module and source code inside a network node.

Advantages:-

- OPNET provides a graphical interface for the topology design, which allows for realistic simulation of the network.
- It has been used widely and it provides real scenario including traffic and all so there is possibility of getting results error free.

Limitations:-

- Simulation in OPNET requires a lot of processing and it may be time consuming for a network containing large number of sender and/or receiver nodes.
- OPNET need parallel processing in order to work efficiently so it requires multiple processors in order to simulate highly scalable scenario.

Chapter 3

ALGORITHMS, MODELS AND PROPOSED SIMULATOR

3.1 ALGORITHMS AND MODELS

3.1.1 Random Way-Point Mobility Model

Random waypoint (RWP) mobility model has been widely used in the mobile ad hoc networks. This model is simple and direct stochastic model. In RWP, a mobile node moves inside a limited continuous system from a current position to a new location by randomly choosing its target positional coordinates, its speed of movement and the amount of time that it will stay when it reaches the target. On reaching the target position, the node becomes stationary for some time choosing according to some random variable and the process repeats itself. Once the pause time expires, the node chooses a new destination, speed and pause time and starts moving. The movement of a node from the starting position to its next target location is defined between the movements of a node from the starting waypoint to its next waypoint is defined as transition length. Figure 3.1 shows an example of moving pattern of a mobile node using the random waypoint mobility model starting at a randomly chosen point or position initially, the speed of the mobile node in the figure is uniformly chosen constant. We note that the movement pattern of a mobile node using the random waypoint mobility model is similar to the random walk mobility model if pause time is zero. In most of the performance of the mobile nodes that use the random waypoint mobility model, the mobile nodes are initially distributed randomly around the simulation area. This initial random distribution of mobile nodes is not typical behavior in which nodes distribute themselves when moving.

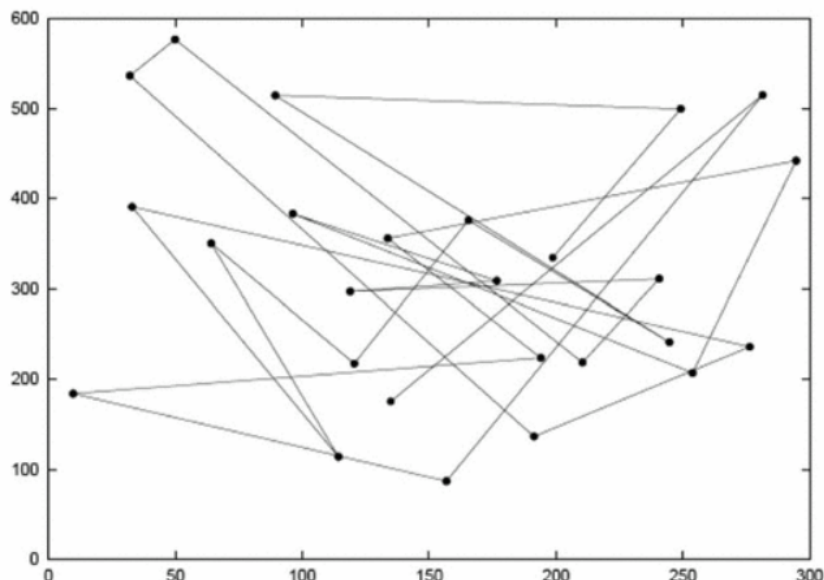


FIGURE 3.1: Moving pattern of a mobile node using the RWP model

When the node reaches at the system boundary it finds a new direction of movement such as it moves again inside the simulation area.

The Random Waypoint model is widely accepted mainly due to its ease of implementation and analysis. However, the basic Random Waypoint model is used in most of the simulations is insufficient to capture the following mobility characteristics:

- **Temporal dependency:** Due to physical constraints of the mobile entity itself, the velocity of mobile node will change continuously and gently instead of abruptly, i.e. the current speed is dependent on the previous speed. However, the speed at two different time slots are independent in the Random Waypoint model.
- **Spatial dependency:** The movement pattern of a mobile node may be inclined by and connected with nodes in its neighborhood. In Random Waypoint, each mobile node moves independently irrespective of its neighbors.
- **Geographic restrictions:** In many cases, the movement of a node may be restricted inside a street or a freeway.

3.1.2 Neighborhood Detection

All the mobile nodes are moving randomly in the simulation area in a random direction irrespective of other nodes location and direction. We set a transmission range for all nodes. Transmission range is the maximum range up to which a mobile device is able

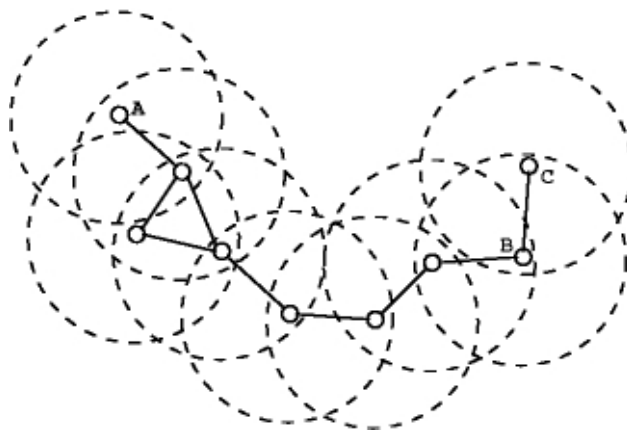


FIGURE 3.2: Node movement and finding their neighbors dynamically

send a packet/communicate directly. But if the target node is not within the transmission range of the source node, we need some intermediate node such that message can be passed to target node using these intermediate nodes. For this, we will find the neighbor nodes for each and every node. In the proposed work, we are taking a common transmission range for each node. We are calculating the Euclidian distance between two nodes say node(i) and node(j). If the distance is less than or equal to the transmission range then, we set them as neighbor of each other and a link will be established between them. If not, then they remain unconnected. As the position of the nodes and changing at every time, so if any node moves away from one nodes transmission range then link will be broken or if any other node comes inside the transmission range a new neighbor will be found. The link formation and broken is very frequent with the movement of nodes. Figure 3.2 shows the neighborhood detection of the nodes in random waypoint mobility.

3.1.3 K-best Energy Efficient Alternate Paths

Traditional algorithms takes into account the least number of hops or the shortest time for choosing the best routing path. For example, as shown in fig. 3.3, if node 1 wants to send some packets to node 5, the traditional algorithms would send through path 1-9-5, as it is the shortest path in terms of distance. Traditional algorithm were designed with an assumption that shortest path are energy saving. However, with this approach we can find that node 9 getting selected in most routing paths with other nodes transmission. This can lead to the chances that node 9 dying out much faster. This may lead to energy inequity between the nodes, eventually leading to network failure.

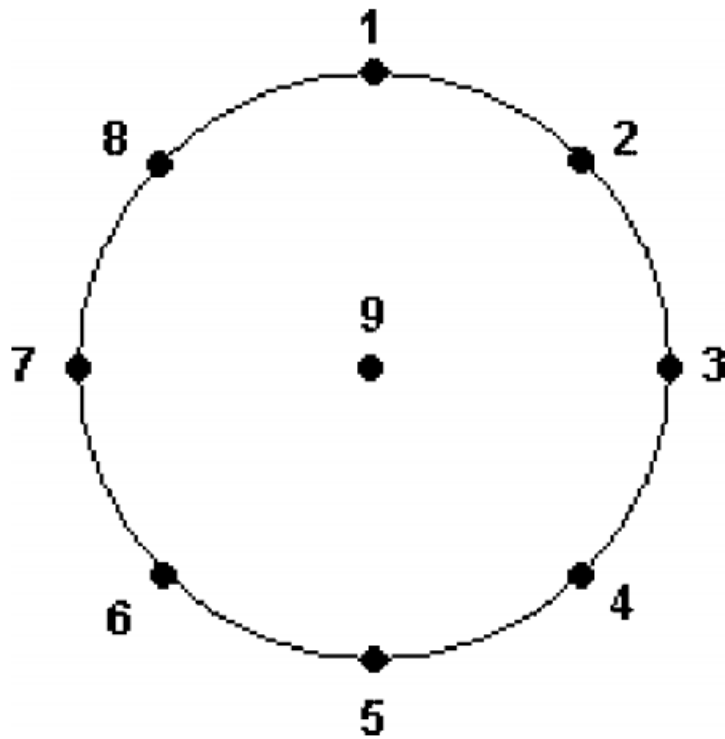


FIGURE 3.3: Placement of nodes in a network

So, a routing algorithm should consider all routing paths and find minimum energy of a node existing in the chosen paths. A routing algorithm should finally choose a path that has the maximum value of the minimum energy in a node in path through all possible paths. Let us assume that there are n total paths that are available with the routing algorithm. The algorithm will find the path on the basis of following:

$$Bestpath : Max\{Min(P1), Min(p2), \dots, Min(Pn)\}$$

Or

$$Min\{1/Min(P1), 1/Min(P2), \dots, 1/Min(Pn)\}$$

Where, $P1, P2, \dots, Pn$ represent the possible complete paths between the source and the target for transmission. Min represents the minimum battery power of a node in a path. Let us suppose that $P1$ is composed of node1, node7 and node4. $Min(P1)$ can be calculated as follows:

$$Min(P1) = Min(Battery(1), Battery(7), Battery(4))$$

Mathematically, let $B(N_i)$ denote the battery power of the i th node. Then cost of path P_1 should be $C(P_1)$, which is proportional to $\min(B(a), B(N_i), B(N_{i+1}), B(b))$, where path j is composed of nodes a, N_i, b .

3.1.4 Finding the K best paths in a network

There are several algorithms available for finding K-shortest-paths in a network.

- Bock, Kantner and Haynes introduce an algorithm that find all possible paths from source to destination, then sorts these paths based on the parameters (generally distance but here are taking battery power.). Then first K paths are taken K shortest (best) paths. This algorithm has two main drawbacks. One is it requires very large number of computations and memory storage. The other is that it requires much effort to solve a problem in which K is small as to solve a problem in which K is large.
- In Pollack introduces an algorithm for finding Kth shortest path in a network. To find the Kth shortest path in a network. To find Kth path this algorithm first finds K-1 shortest paths. Then the cost of edge in each of the 1st, 2nd, ..., (K-1)st shortest paths is set to infinity. The shortest path is calculated in each such case. This algorithm can be considered the most applicable among the algorithms reviewed in this section because it has the lowest computational upper bound for small K. But the computational power increases exponentially with the value of K.
- In Clarke, Krikorian, algorithm to find K shortest path, it first finds the shortest path, then finds the K shortest path from all paths that branch-out from the shortest path. But the efficiency of this algorithm depends upon the topology of the network.
- In Sakarovitch algorithm, it first finds H, $H \geq K$, shortest paths that may contain loops by a procedure similar to Hoffman and Pavleys algorithm. The H paths are scanned for the K shortest paths. But in this case the efficiency also depends on the topology of the network.

The proposed algorithm

1. Determine A using Yens algorithm, i.e obtain first shortest path,

-
2. Iteration k ($k = 2, 3, K$). To determine A^k . In order to find A^k , the shortest paths A^1, A^2, A^{k-1} , must have been previously determined. A^k is then found as follows:
- (a) Check if the subpath consisting of the first i nodes of A^{k-1} in sequence coincide with the subpath consisting of the first i nodes of A^j in sequence of A^j for $j = 1, 2, \dots, k-1$. If so, remove edge $i \rightarrow q$ where q is the $i+1$ st node of A^j ; otherwise, make no changes. Go to step (b).
 - (b) Apply a shortest-path algorithm to find the shortest path from (i) to (N), (here we used Dijkstras algorithm for finding shortest path.) allowing it to pass through those nodes that are not yet included in the path. Note that the subpath from (1) to (i) is R_i^k ; and the subpath from (i) to (N) is S_i^k , the spur of A_i^k . Note also if there are more than one subpaths from (i) to (N) that have the minimum length, take any arbitrary one of them and denote it by S_i^k .
 - (c) Find A_i^k by joining R_i^k and S_i^k . Then add A_i^k to List B.
 - (d) Store only $K-k+1$ shortest A_i^k in List B.
3. Find from List B the path(s) already in List A exceed K , return.

Otherwise, denote this path (or an arbitrary one, if there are more than one such paths) by A^k and move it from List B to List A leaving alone the rest paths in List B. Then go to step 2.

The algorithm requires approximately $N^2 + KN$ addresses to store the adjacency list, list A, list B, and some intermediate data (like spur node).

3.1.5 Power Consumption

The power consumption of nodes for transmission and/or movement is [2] according to the Lucent IEEE 802.11 WaveLAN PC card 2.4 GHz direct sequence spread spectrum linear model consumption. The power consumption data is as:

- $m_{send} = 1.89mW.s/byte$
- $m_{rev} = 0.494mW.s/byte$
- $m_{discard} = 0.2mW.s/byte$
- $Idle = 808mW$.

Total cost of power consumption in a path is given as:

$$Cost = m_{send} \times size + k \times m_{discard} + m_{recv} \times size \quad (3.1)$$

Where,

- k: is total number of intermediate nodes.
- size: is the size of packet being transmitted.

3.2 PROPOSED SIMULATOR

In this section we represent the details of simulator that we designed to evaluate the performance of K-best Energy Efficient Paths and the result that we obtained. The simulator is a Java-based graphical editor simulator for modeling mobile nodes initialization, random movement, neighborhood detection, finding K-best energy-efficient alternate paths. It is platform-independent so as runs on all available platforms (Linux/Windows/Mac). The current simulator was designed from the beginning to support Mobile Ad-hoc Network simulation for a large number of nodes. The objective leads to the following main requirements:

- Scalability,
- Graphical interface to user,
- Supports both static and dynamic movements of nodes,
- Change of topology due to dynamic movements of nodes,
- Provides energy-efficient multiple paths,
- Supports cross-platform environment.

This simulator have only the following parameters such as:

- Node-id: it is unique for every node.
- Node-position: it is the position of node (x, y)
- Mobility Model : Random Way-point mobility
- State: a node can remain in maximum two states either moving or stationary.

- Battery Power: All nodes are initialized with same battery power.

The design of simulator uses Java AWT and SWING components for providing graphical interface. The simulator contains a main frame with two panel. First panel contains a square area that is network area which binds the nodes to move inside itself and prints alternate path if found between source and destination. Second panel contains general information about simulator and control of the network. It displays the mobility model used i.e Random Way-point Mobility model, Simulation area type which is square in this. And then it provides some parametric control to take input from user such as total number of nodes available, source node which want to transmit/send some packets, target node to which source wants to send information. Transmission range is a dropdown list, so that user can select (5, 10, 20, 40, and 50). Pause time is also a dropdown list (1, 2, 3, 4, and 5).

Start Button to start the simulation with initialization of nodes with random position and direction of movement. Pause button do pause the network at an instance. Stop button to stop the simulation completely, after stopping simulation one can set new simulation parameters. Status button displays the real-time information in a table that contains node id, node position (x, y), available battery power and total number of neighbors. Graph button to plot the graph between cost (in Joule) vs. size of packets (in Bytes) using different paths. Cost is the consumption of battery power while sending a packet using a discovered path in the network between source and destination.

Uses inputs the total number of nodes, source node, target node, transmission range, pause time and click on start button. That initializes the nodes using Random Waypoint Mobility model. Dynamically nodes finds their neighbors and sets up links with them or if any node moves away, the link is broken. The source node finds alternate paths, labels them and display at the bottom of the simulation area. A status table is also provided that update dynamically nodes information (node id, position (x, y), available battery power and total number of neighbors). And graph is plotted for the size of packets (bytes) vs. cost (joule) of the path for all calculated alternate paths.

Chapter 4

Snapshots

4.1 Graphical Interface of Simulator

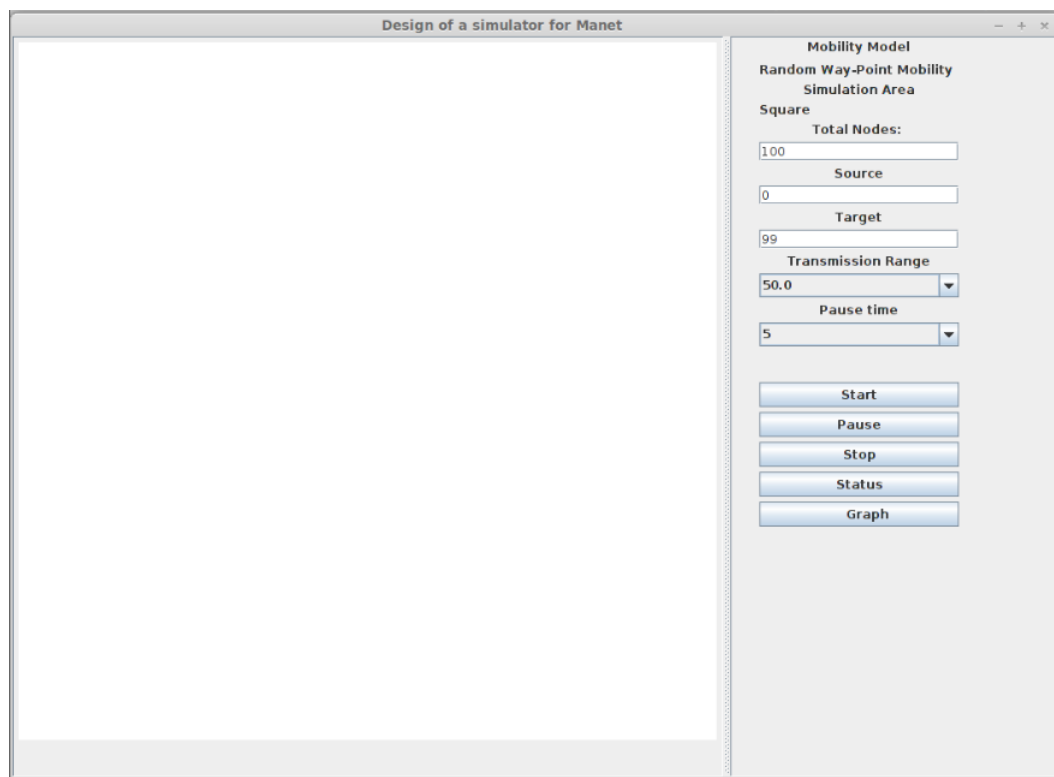


FIGURE 4.1: Graphical Interface

Simulator has two panel. First panel contains a square simulation area for network. And the second panel has control over to simulator.

- Total number of nodes (integer).

- Source node (not greater than total number of nodes). Node wants to send some packets to a target node.
- Target node (not greater than total number of nodes). Node will receive packets sent by source node.
- Transmission range: maximum range upto which a node can directly send packets. If distance between any two nodes is less than or equal to transmission range, then they will be neighbors of each other.
- Pause time: pause time between two consecutive movements of any node.
- Start button: to start the simulation.
- Pause button: to pause the simulation of a network at any desired point of time.
- Stop button: to stop the simulation of a network.
- Status button: display a table that shows different parameters of all nodes (node id, node position (x, y), available battery power, total number of neighbors). Table continues to update its information with respective of node movement.
- Graph button: shows the graph between sizes of packets vs energy consumed in all available alternate paths.

4.2 Node Initialization

Nodes are initialized according to Random WayPoint Mobility Model.

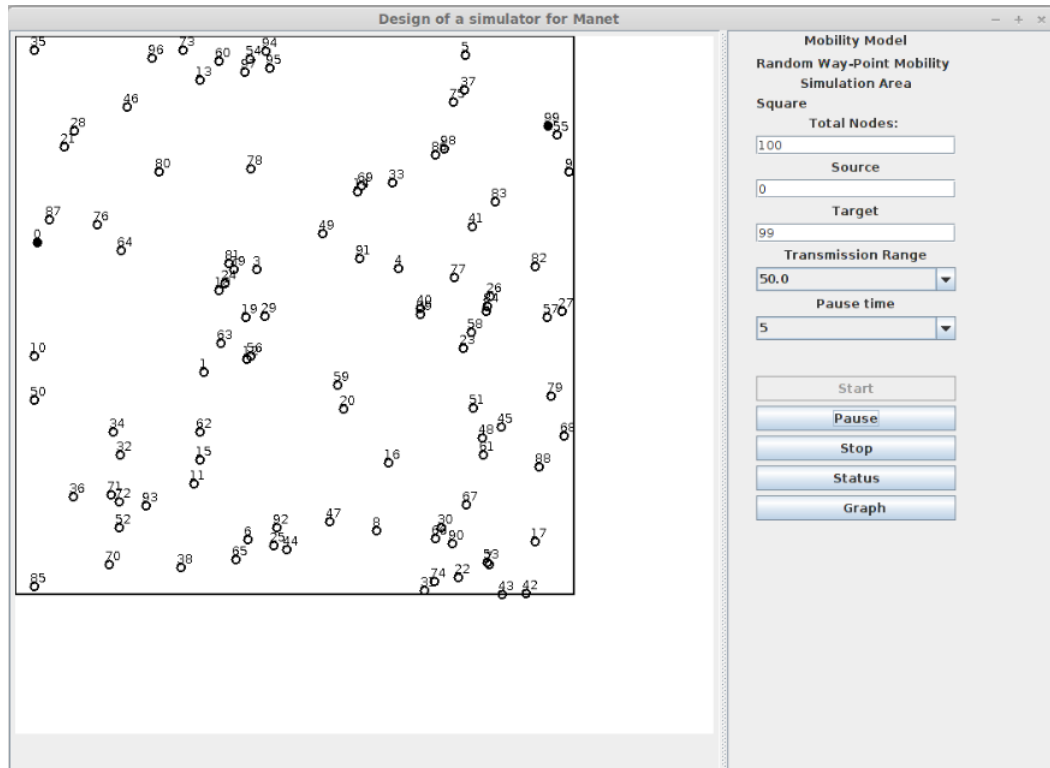


FIGURE 4.2: Node Initialization

4.3 Neighborhood detection

Nodes find their neighbors, if any node lies inside the transmission range then a link is created or if it moves away then link is broken.

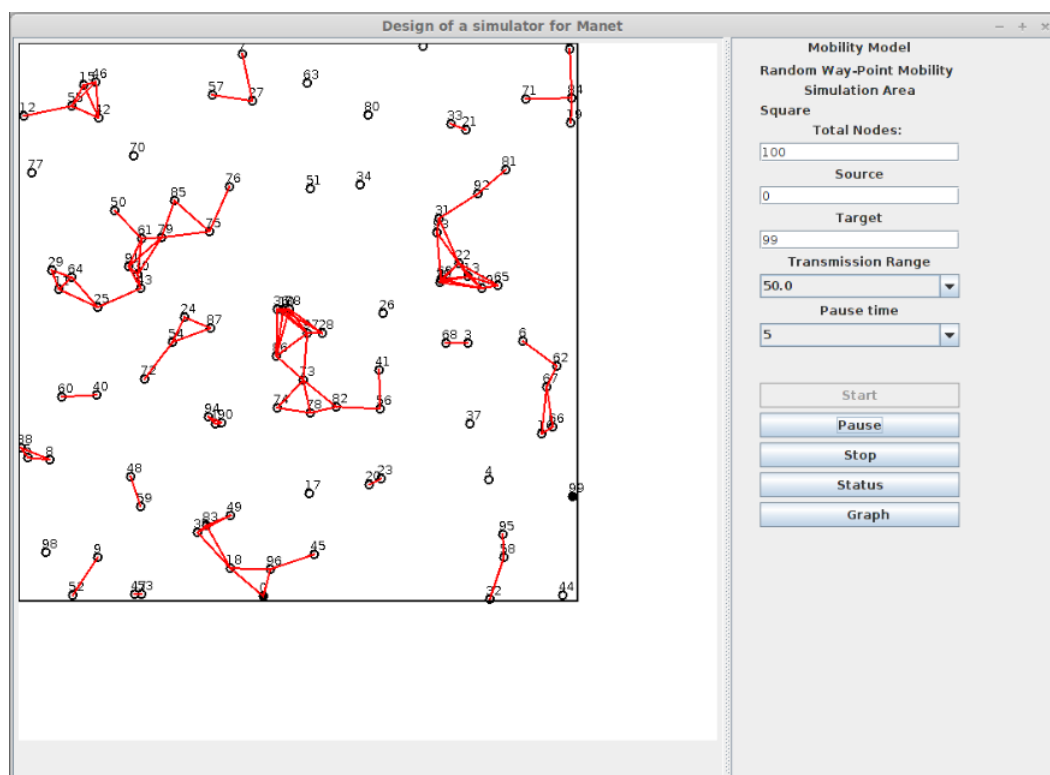


FIGURE 4.3: Neighborhood detection

4.4 Finding Best Path

Finds best path based on energy parameter (path with minimum energy consumption while transmission) between source and target node.

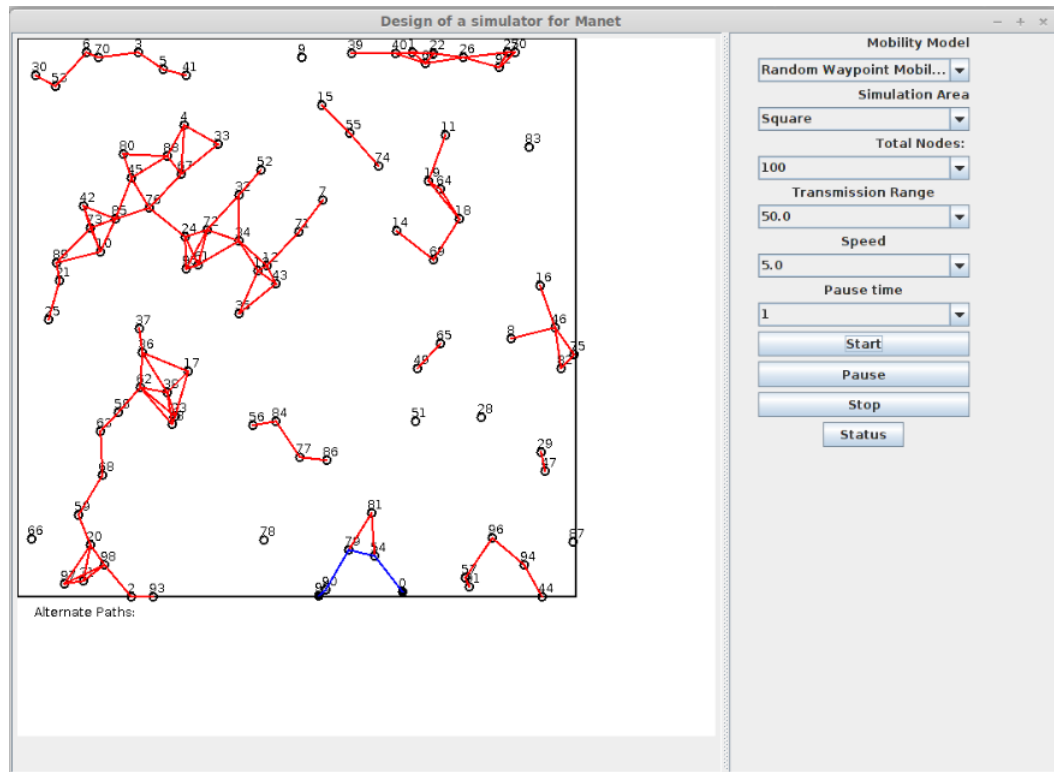
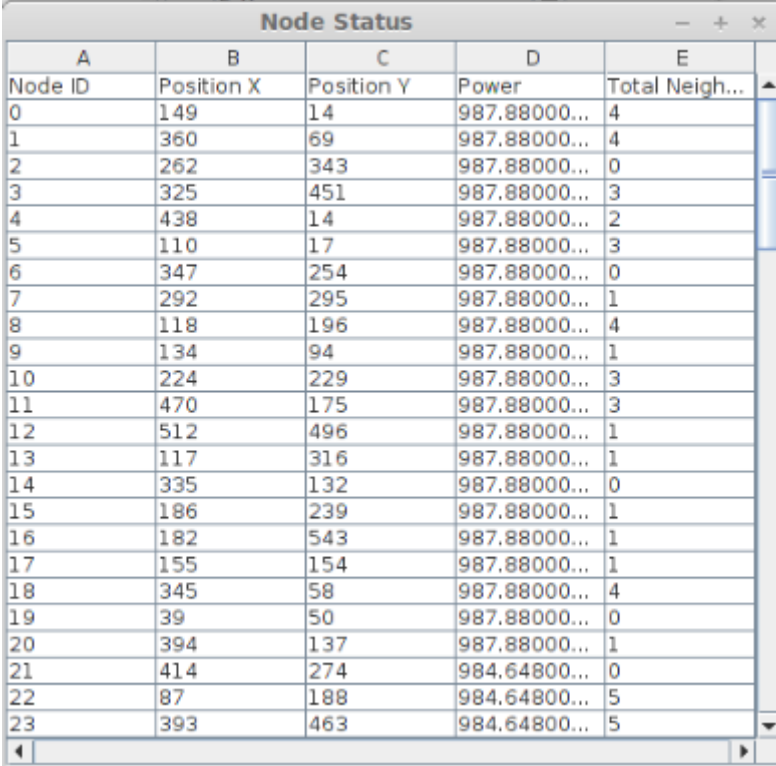


FIGURE 4.4: Finding Best Path

4.5 Status of Nodes

Table is shown that displays real time information of all nodes. It updates its information after every pause timer expires.



A	B	C	D	E
Node ID	Position X	Position Y	Power	Total Neigh...
0	149	14	987.88000...	4
1	360	69	987.88000...	4
2	262	343	987.88000...	0
3	325	451	987.88000...	3
4	438	14	987.88000...	2
5	110	17	987.88000...	3
6	347	254	987.88000...	0
7	292	295	987.88000...	1
8	118	196	987.88000...	4
9	134	94	987.88000...	1
10	224	229	987.88000...	3
11	470	175	987.88000...	3
12	512	496	987.88000...	1
13	117	316	987.88000...	1
14	335	132	987.88000...	0
15	186	239	987.88000...	1
16	182	543	987.88000...	1
17	155	154	987.88000...	1
18	345	58	987.88000...	4
19	39	50	987.88000...	0
20	394	137	987.88000...	1
21	414	274	984.64800...	0
22	87	188	984.64800...	5
23	393	463	984.64800...	5

FIGURE 4.5: Status of Nodes

4.6 Finding K-best Alternate Paths

K-best alternate paths, found using the above proposed algorithm. And displayed at the bottom of simulation area.

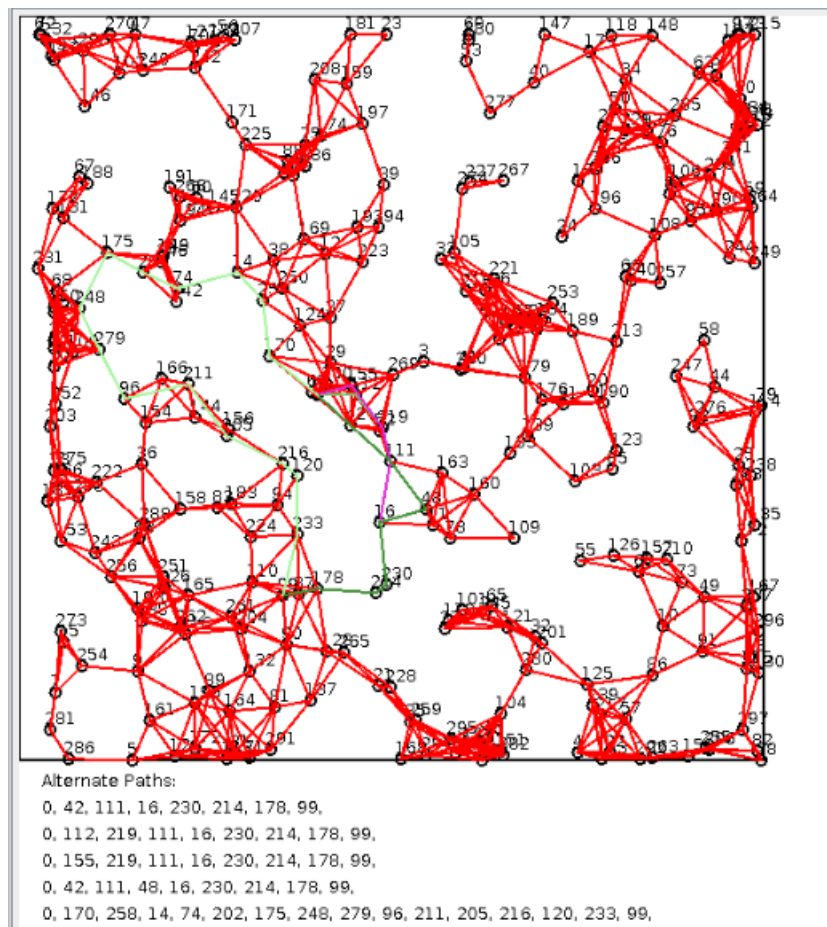


FIGURE 4.6: Finding K-best Alternate Paths

4.7 Energy Consumption Graph

Graph shown for different sizes of packets (in bytes) transmitted vs energy (in jule) consumed in every path.

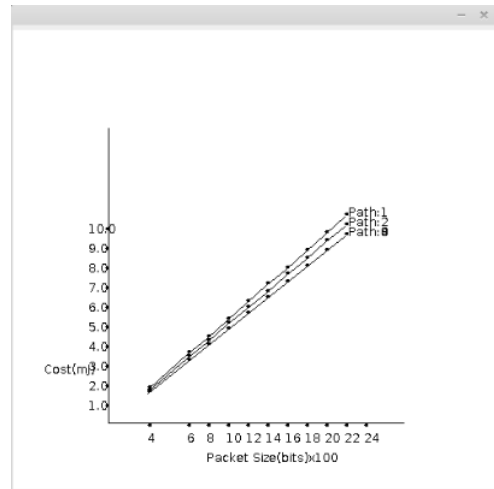


FIGURE 4.7: Energy Consumption Graph

4.8 Complete Simulator

The screenshot displays the 'Design of a simulator for Manet' application. The main window features a network graph with nodes and edges. A control panel on the right includes the following settings:

- Mobility Model:** Random Way-Point Mobility
- Simulation Area:** Square
- Total Nodes:** 200
- Source:** 0
- Target:** 99
- Transmission Range:** 50.0
- Pause Time:** 3

Buttons for Start, Pause, Stop, Status, and Graph are provided. Below the graph, an 'Alternate Paths' section lists several paths. A 'Node Status' table is also visible, showing the following data:

Node ID	Position X	Position Y	Power	Total Neighbors
0	219	77	999.192	2
1	281	87	999.192	2
2	64	274	999.192	5
3	422	406	999.192	7
4	7	197	999.192	5
5	179	504	999.192	4
6	199	387	999.192	4
7	392	404	999.192	7
8	23	196	999.192	1
9	184	52	999.192	9
10	407	282	999.192	1
11	122	478	999.192	13
12	402	429	999.192	8
13	388	196	999.192	19
14	155	445	999.192	1
15	191	507	999.192	5
16	470	502	999.192	5
17	19	354	999.192	4
18	245	513	999.192	3
19	522	391	999.192	5
20	161	43	999.192	7
21	547	438	999.192	6
22	335	14	999.192	2
23	19	524	999.192	13
24	65	18	999.192	4
25	116	517	999.192	5
26	393	344	999.192	4

The bottom right corner shows a graph titled 'Packet Sizes' with a y-axis labeled 'Cost' and a line graph showing data points. The Eclipse IDE interface includes a terminal, a code editor with a Java snippet, and a status bar at the bottom.

FIGURE 4.8: Complete Simulator

Chapter 5

Conclusion

A network simulator is a software program that replicates the working of a computer network. In simulators, the network is typically modelled with devices (nodes) and the performance is analyzed. Users can modify the simulator to fulfil their specific analysis needs. The proposed simulator gives a complete vision of the network scenario with different parameters. A path that consumes minimum energy in the network is selected as the first best path. K-best alternate paths found using Yens algorithm. The simulator is designed to be topology adaptive and scalable. The energy consumption behavior found to be linear with varying size of packets delivered. In future routing and transmission protocols can also be implemented with further modification.

References

1. Sanjay Kumar Dhurendher, Sudip Misra, "EEAODR:An Energy-Efficient Ad-hoc On-demand Routing Protocol for Mobile Ad-hoc Networks" International Journal of Communication Systems, 2009
2. Stefano Basagni, Marco Conti, Silvia Giordano, Ivan Stojmenovic "Mobile Ad Hoc Networking", 2004, 1st edition, 2004
3. G. Lynn, Noubir, R. Rajaram, "Mobility Model for Ad-hoc Network Simulator", IEEE INFOCOM, 2004
4. L. M. Feeney, "An Energy Consumption Model for Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks", Kluwer Academic Publishers, pp.239-249, 2001.
5. C. Bettstetter, G. Resta, P. Santi, "The Node Distribution of the Random Waypoint Mobility Model for Wireless Ad Hoc Networks", IEEE Transactions on mobile computing, vol.2, no.3, pp. 257-269, July-Sep 2003.
6. N. Katoh, T. Ibaraki, H. Mine, "An efficeint algorithm for K shortest simple path", Networks An International Journal, Vol. 12, pp. 411-427, Oct. 2006
7. Jin Y. Yen, "Finding the K Shortest Paths in a network",Management Science, Vol.17, No. 11, pp. 712-716, Theory Series(Jul., 1971)
8. Amit Kumar, Dheerendra Srivastava, "Design of Simulator for Energy Efficient Clustering in Mobile Ad Hoc Networks", National Institute of Technology, Rourkela, 2012.
9. E. Hyytia, P. Lassila, and J. Virtamo, "Spatial Node Distribution of the Random Waypoint Mobility Model with Applications", Networking Laborartory, Helsinki University of Technology, Vol. 5 no. 6 pp. 680-694, IEEE June 2006