A DISSERTATION

ON

A Dynamic Approach Multipath Routing for Mobile Ad Hoc Networks

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE OF

MASTER OF TECHNOLOGY

IN

SOFTWARE TECHNOLOGY

BY

BHUPESH PANDEY

ROLL NO: 2K12/SWT/05

UNDER THE GUIDANCE OF

VINOD KUMAR

Associate Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DELHI TECHNOLOGICAL UNIVERSTIY

2015

DECLARATION

I hereby declare that the thesis entitled "**A Dynamic Approach Multipath Routing for Mobile Ad Hoc Networks"** which is being submitted to the **Delhi Technological University**, in partial fulfillment of the requirements for the award of degree of **Master of Technology in Software Technology** is an authentic work carried out by me. The material contained in this thesis has not been submitted to any university or institution for the award of any degree.

____
Bhupesh Pandey

Department of Computer Science & Engineering,

Delhi Technological University,

Delhi.

CERTIFICATE

Delhi Technological University
(Government of Delhi NCR)
Bhawana Road, New Delhi-42

This is to certify that the thesis entitled **"A Dynamic Approach Multipath Routing for Mobile Ad Hoc Networks"** done by **Bhupesh Pandey** (Roll Number: **2K12/SWT/05**) for the partial fulfillment of the requirements for the award of the degree of **Master of Technology** in **Software Technology** in the **Department of Computer Science & Engineering**, Delhi Technological University, New Delhi is an authentic work carried out by her under my guidance.

Project Guide:
Vinod Kumar
Associate Professor
Department of Computer Science & Engineering
Delhi Technological University, Delhi

ACKNOWLEDGEMENT

I take this opportunity to express my deepest gratitude and appreciation to all those who have helped me directly or indirectly towards the successful completion of this thesis.

Foremost, I would like to express my sincere gratitude to my guide **Vinod Kumar, Associate Professor, Department of Computer Science & Engineering**, **Delhi Technological University, Delhi** whose benevolent guidance, constant support, encouragement and valuable suggestions throughout the course of my work helped me successfully complete this thesis. Without his continuous support and interest, this thesis would not have been the same as presented here.

Besides my guide, I would like to thank the entire teaching and non-teaching staff in the Department of Computer Science and Engineering, DTU for all their help during my course of work.

Bhupesh Pandey
Roll No. 2K12/SWT/05
Master of Technology (Software Technology)
Delhi Technological University
Bawana Road, Delhi - 110042

Table of Contents

1. Introduction---6
1.1 General Concepts---6
1.2 Motivation---7
1.3 Related Work---3
1.4 Problem Statement---9
1.5 Scope of Work---9
1.6 Thesis Organization--9

2. Routing in Mobile Ad Hoc Networks---11
2.1 MANETs Routing Protocols---11
2.1.1 AODV---11
2.1.2 Dynamic Source Routing---14

3. Related Researches on Routing Techniques for MANETs-----------------------------15
3.1. Load Balancing Technique---15
3.1.1 Cached Hops Routing Protocol--16
3. 2 Trust Based Routing---19
3.3 Multipath Routing--22
3.4 QOS Aware Multipath Routing--23
4. Proposed Method---26
5. Implementation---27
5.1 Network Simulator---27
5.1.1 Components of NS2--27
5.1.1.1 NAM---28
5.1.1.2 C++Models--28
5.2 NS2 Setup--28
5.2.1 XGraph---28
5.2.1 AWK--28

6. Results and Analysis---29

6.1 Simulation Traces---29
6.2 Piloted Graphs---31
7. Conclusion ---35

References--36

Chapter 1: Introduction

Mobile Communication is an infrastructural need for any country. Mobile Ad Hoc network plays an important role in executing emergency services during disasters. In MANETs nodes communicates each other without help of any centralized control. MANETs are highly dynamic in nature with limited bandwidth and capacity. Efficient routing and communication approaches are always challenging to design and implement MANETs. Due to frequent link breakages multiple route discoveries are carried out. These route discoveries lead to extra overhead on broadcasting messages and updating routing tables.

Multipath routing methods are already suggested to find multiple optional routes for a pair of nodes with one time root discovery. MANETs now became a popular topic aong researchers to make its more utilization on future communication networks. Many techniques are suggested considering limited resources and required Quality of services. Already existing techniques are mostly based on nodes characteristics e.g. battery energy, signal strength and mobility. We propose to have MANETs processing capacity as an important factor on designing. Each node is a standalone machine with individual processing capacity and power. Based on processing capacity into consideration new routing techniques can be discovered. Researches need to be done considering processing capacity as one important parameter. We proposed a Multipath routing method that uses QOS parameters and processing capacity to choose a route in case of any link failure. Advanced MANETs are designed for multiple query processing and those could be used in informational networks in future. This can be used in network information system application. When a node requests information to another node in MANETs if destination node has that information will directly respond or that will forward request further.

1.1 General Concepts

Mobile Ad hoc networks are infrastructure less wireless networks, in which network nodes can send or receive data to each other without help of any centralized access point or base station. Those temporary networks also called standalone networks. Such networks are formed with mobile nodes those move in or out from each other communication ranges with time.

Nodes within communication range of each others can send or receive packets directly to each other, while nodes not in communication range of each other communicate with the help of other nodes (intermediate nodes). Intermediate nodes receive the data checks address fields and forward to the destination node. There can be multiple intermediate nodes with in a communication between two nodes. Each node is itself responsible for dynamic routing of packets at any instance of time.

Figure: A typical Mobile Ad Hoc Network

1.2 Motivation

MANETs are assumed to be identical in nature, in terms of factors like battery energy, processing capacity, signal strength. But realistic Manets are difficult to have similar characteristics in nature. Nodes have different battery, processing capacity and signal strength. Each node decides itself the route to its packets. They are capable enough to adopt different network topologies on the fly. In case of a heterogeneous environment nodes updates its topology dynamically. Mobile nodes in Mobile Ad hoc networks have limited Battery energy, signal strength, processing capacity. Various routing methods have been proposed for MANETs. All classical approaches and methods assume nodes are identical.

In MANETs nodes have limited energy, bandwidth and processing capacity. Realistic MANETs are highly moving in nature that creates the more link breakages. Hence designs of MANETs are major challenges for Researchers today. Various classical routing methods have been suggested. Most of popular routing methods are based on assumption of identical nodes. In mobile ad hoc network nodes are not identical; they have different battery energy, processing capacity and signal strengths.

MANETs have been a popular research field in case of mobile communication from last two decade. A lot of researches are ongoing for the techniques of their designs and realistic implementation. Due to no overhead of infrastructure network it can be used widely in many fields of communication. From the vehicular network to emergency aid services and from simple data transfer to query processing or computing networks. These applications are very critical and seek high performance MANETs with high success response and less end to end delay. Multipath routing is a better approach for route finding. Using MANET's nodes processing capacity along with battery energy and signal strength for route selection can provide better results especially in case of QoS defined networks or query processing networks.

1.3 Related Work

Manets has main feature of adopting dynamic configuration and routing. As per the communication needs they update their configuration on the fly in terms of network topology, medium or strength. An efficient design for MANETs is a big challenge for researchers and developers because of their limited resources as processing capacity, battery energy, communication range etc.

A smart routing technique is needed to efficient utilization of limited resources and to achieve more throughputs, less packet loss with minimum delay. Existing classical routing techniques for wired networks do not well suitable for wireless networks. Due to frequent link failures and mobility it is more challenge to have an efficient routing algorithm. Many routing algorithms are suggested till date for Mobile Ad hoc network and their count is continuously increasing. Most of the classical routing methods assumed identical nature of MANETs. But realistic MANETs are not necessarily identical, they may have different in nature and capacity in terms of network size, topology, processing capacity, battery energy etc. Based on their characteristics different routing algorithms performs well for different networks.

I have gone through various routing methods including classical approaches. As earlier said popular traditional methods assumed to have MANETs with similar characteristics. Considering various parameters researches comes with routing algorithms based on different parameters. Those algorithms are suggested treating one or more parameters as route decision maker. With the evolution of various network services QOS (Quality of services) based routing methods were also suggested. Those methods are designed as taken particular QOS into consideration.

Various routing methods and approaches are experimented, observed and compared. Comparison results are used to adopt efficient algorithm for particular scenarios.

1.4 Problem Statement

As discussed in the previous sections result is that MANETs are the very important networks in mobile communication. It has special advantage of without need of any infrastructure network over other popular networks. Due to this characteristic it has a very crucial use in case of emergency services and vehicular networks.

Since they are high mobility networks with nodes having limited resources it is always demanding to have appropriate routing algorithms.

Aim of this thesis is it to develop an efficient method for routing of nodes of MANETs. That utilize more and more limited resources like energy, bandwidth and ensures best results as high throughput, low end to end delay.

We are using multipath routing approach for this purpose that reduces multiple root discoveries in case of any path failure.

1.5 Scope of work

Due to high mobility in MANETs more link failures happens. That results in multiple route discoveries with time and multiple route table update. That wastes measurable amount of energy in bandwidth.

For such scenario MANETs needed efficient routing method which improves their performance and lifetime. With the advancement in technology future MANETs can be imagined as standalone query processing network, powerful vehicular network etc.

1.6 Thesis Organization

Chapter 1: It is introductory section for Mobile ad hoc network and its design challenges

Chapter 2: It introduced with the major classical routing protocols

Chapter 3: It has related research on MANETs routing

Chapter 4: It contains summary of proposed method

Chapter 5: Implementation part of routing method and introduction with network simulator tool used to simulate multipath routing

Chapter 6: Compares and concludes the results from existing approaches with suggested approach

Chapter 6: Conclusion of this thesis

Chapter 2: Routing in Mobile Ad Hoc networks

2.1 MANETs Routing Protocols

Routing of mobile nodes in Ad hoc networks is a major challenge. Various routing techniques are suggested by researchers. Which of based on following routing protocols:

Reactive Protocol

As the name suggests this protocol works on demand. In case no prior route is existed. When a node needs to communicate with any other node within the network and no route is known for this communication. Reactive protocol is used to determine the route between them. For each communication discovering routes just then can make delay in communication.

Proactive Protocol

While in case of proactive protocol routes are discovered in advance for any pair of nodes. That is done constantly in some intervals. The advantage of this method is to avoid delay in discovering route when a node needs to start communicate. But disadvantage is more efforts are done for the routes discoveries which will not be used later.

Hybrid Protocol

This works as a hybrid form of both Proactive and Reactive protocols. As per the requirement multiple techniques can be made using combination of both Proactive and Reactive protocols.

2.1.1 AODV

The Ad hoc On Demand Distance Vector (AODV) routing technique is a reactive routing protocol designed for ad hoc mobile networks. AODV can be used for both unicast and multicast transmission. Since it is an on demand routing method, route discovery is done only if a node need to communicate. Once a route is discovered that is maintained later also. Additionally, AODV forms trees which connect multicast group members. In route discovery an AODV has multiple connections between the nodes. A sequence number is used to determine the most recent route. AODV avoids loops and redundant requests that are more suitable for mobile ad hoc networks as compared to proactive protocols like DSDV.

AODV constructs routes utilizing a route demand/route answer inquiry cycle. At the point when a source hub seeks a route to a destination for which it doesn't as of now have a route, it shows a route demand (RREQ) bundle over the system. Hubs getting this bundle overhaul their data for the source hub and set up in reverse pointers to the source hub in the route tables. Notwithstanding the source hub's IP address, current sequence number, and show ID, the RREQ likewise contains the latest sequence number for the destination of which the source hub is mindful. A hub getting the RREQ may send a route answer (RREP) on the off chance that it is either the destination or on the off chance that it has a route to the destination with relating sequence number more noteworthy than or equivalent to that contained in the RREQ. On the off chance that there is the situation, it unicasts a RREP back to the source. Else, it rebroadcasts the RREQ. Hubs stay informed concerning the RREQ's source IP address and show ID. In the event that they get a RREQ which they have effectively prepared, they toss the RREQ and don't forward it.

As the RREP packet back to the source, hubs set up forward routes to the destination. When the source hub gets the RREP, it may start to forward information bundles to the destination. On the off chance that the source later gets a RREP containing a more noteworthy sequence number or contains the same sequence number with a littler hop count, it may overhaul its steering data for that destination and start utilizing the better route.

The length of the route stay dynamic, it will keep on being kept up. A route is viewed as dynamic the length of there are information parcels intermittently venturing out from the source to the destination along that way. When the source quits sending information bundles, the connections will time out and in the long run be erased from the middle of the road hub steering tables. In the event that a connection break happens while the route is dynamic, the hub upstream of the break engenders a route mistake (RERR) message to the source hub to illuminate it of the now inaccessible destination(s). In the wake of accepting the RERR, if the source hub still wishes the route, it can reinitiate route revelation.

Multicast routes are situated up in a comparable way. A hub wishing to join a multicast gathering telecasts a RREQ with the destination IP location set to that of the multicast bunch and with the 'J'(join) banner set to show that it would like to join the gathering. Any hub getting this RREQ that is an individual from the multicast tree that has a sufficiently new sequence number for the multicast gathering may send a RREP. As the RREPs engender back to the source, the hubs sending the message set up pointers in their multicast route tables. As the source hub gets the RREPs, it stays informed concerning the route with the freshest sequence number, and past that the littlest hop check to the following multicast bunch part. After the predetermined disclosure period, the source hub unicast a Multicast Activation (MACT) message to its chosen next hop. This message fills the need of initiating the route. A hub that does not get this message that had set up a multicast route pointer will timeout and erases the pointer. On the off chance that the hub getting the MACT was not as of now a piece of the multicast tree, it will likewise have been staying informed regarding the best route from the RREPs it got. Thus it should likewise unicast a MACT to its next hop, thus on until a hub that was already an individual from the multicast tree is come to.

AODV keeps up routes for whatever length of time that the route is dynamic. This incorporates keeping up a multicast tree for the life of the multicast bunch. Since the system hubs are versatile, it is likely that numerous connection breakages along a route will happen amid the lifetime of that route. The papers recorded beneath depict how connect breakages are taken care of. A paper portrays AODV without multicast yet incorporates point by point recreation results for systems up to 1000 hubs. Other paper portrays AODV's multicast operation and subtle elements reenactments which demonstrate its right operation. The web drafts incorporate portrayals of both unicast and multicast route revelation, and additionally specifying how QOS and subnet collection can be utilized with AODV. At long last, the IEEE Personal Communications paper and the Infocom paper points of interest an inside and out investigation of reproductions contrasting AODV and the Dynamic Source Routing (DSR) convention, and looks at every convention's particular qualities and shortcomings.

AODV uses destination sequence number to find freshness of the path that other on demand protocol does not have. That makes AODV unique among other on demand reactive protocols. It modifies route tables of nodes based on the latest destination sequence number path.

AODV has routing tables for all nodes within the network. Each routing table has following entries: Destination address (that identifies destination node), Next hop, Destination sequence number (freshness of route), hop count (number of intermediate routes) and lifetime (is route existing).

When a node has to send a packet it first checks routing table if an route already exist from source to destination. If it exist, node direct forward packet to the next hop in the table. If not then it starts a route discovery by sending RREQ (Route request) control packets to the neighbor nodes. This is also called flooding.

Packet type	Rsrvd	Hop Count
BID		
Destination Address		
Destination Sequence Number		
Source Address		
Source Sequence Number		
Time		

Figure: Packet structure in AODV request

2.1.2 Dynamic Source Routing

DSR is also a reactive routing protocol like AODV. DSR uses similar route discovery process as in AODV instead it does not have periodic control message e.g. HELLO message, to check existence of the route. It reduces measurable control message overhead as in case of AODV.

Disadvantage of DSR to get wastage of bandwidth in case of non existing route. In case of failure intermediate node starts route discovery as in case of AODV. Once a route is discovered it is stored as cache in source node. On failure route maintenance is done.

Chapter 3: Related researches on routing techniques for MANETs

3.1 Load balancing technique

To maximize energy efficiency is one of the most important target of MANET routing protocol, MANET nodes has fixed amount of energy. Many routing protocols for Mobile ad hoc networks have been suggested in last two decades. These are DSR, AODV, DSDV, TORA as discussed in earlier section and many are suggested with observation and reference of these protocols.

The standard definition of MANETs assumes there is no path or point known for any nodes in Mobile ad hoc networks. In any case path need to be identified based on route discoveries. It is observed most of the effort or overhead is gone in finding location of a node. Our major goal of MANETs routing methods are to increase more network response, more end to end delivery, to increase energy efficiency, increase in network existence, and to reduce end to end delay. The network throughput is in general determined by data packet transfer ratio when the many countable contribution to energy used is determined by routing effort which is the number or size of routing control packets. The normal observation based on experiments and simulations In the network simulator NS 2.35 is that reactive technique, those finding paths dynamically by request with no effort in advance, perform much improved than other in advanced reactive routing method, which try to maintain the routs for all source-destination pairs. In node to node reactive routing method (e.g. used in Ad hoc On-demand Distance Vector routing, every middle hop computes to which location or hop the routed data packet should be forwarded further. Rout requests are created at every node by local announcing in case of route discovery. A normal flooding announcement for path requests creates a countable redundant packet effort that is a main reason of inefficiency of Mobile ad hoc network routing methods. One method is suggested node caching of nodes in case of root discovery [2]. The nodes which are latest involved in data packet flow are stored and which are then further forward route requests.

A workload dependent adaptive load balancing method that is depend on the fact that by leaving rout discovery packets (RREQ) according to the load status of each nodes, nodes can be excluded from route paths. We apply new energy efficiency measure to MANET routing technique. The target of energy known routing method is to get increased the network existing time. Also, we present new energy saving routing techniques which do adaptive load adjustment technique to our earlier suggested node caching improvement to the Mobile ad hoc networks routing method [2].

3.1.1 Cached Hops routing Protocol –

Let’s firstly talk about node storing improvement of AODV. After that we talk about the adaptive load adjusting method implied to MANET routing techniques. At last, a new node caching AODV is presented with adaptive load adjustment that uses both the method discussed load adjustment and caching of nodes. We have observed that the nodes participated in latest data packets forwarding have more trusted detail about next nodes and have improved paths than other Mobile ad hoc network nodes.

In data packet forwarding the stored nodes are hops latest participated, and select those stored hops to forward route discovery requests. Also earlier methods, hop caching also implied the fact that the announcement for route discovery is not exactly announced. It does not need to reach every hop but only one needed target node. Hence this technique leaves route requests transferring from the hops those are not stored as cache at the time of possible missing target. Leaving route discovery packet transferring from the other hop considerably deducted routing extra effort at the instances when destination is missing. Researchers solved these known drawbacks of CDS – overuse of dominating (cached) nodes – by a new load-adjustment method. They performed more observative research of AODV node cached method in NS 2.35 They also calculated routing workload division among Mobile ad hoc network nodes. The observative study with NS 2.35 of transferring workload adjustment for AODV node cached method makes measurable enhancement in extra effort and delivery ratio.

The updated route request uses a constant minimum value of threshold as K. The first route request is transferred done with low threshold K. At the time a node M receives the route request packet, it observe and compare the recent time T with the time T(M) when the latest data packet through M has been transferred. If T −K >T(M), then M does not belong to the current node cache and, therefore, M will not forwarded the route request. Else if T − K ≤ T(M), then M is in the node stored and the route discovery request is transmitted normal way. Obviously, the node in cache stored can’t make sure existence of routes among all pairs of nodes, Hence if the route request with the small threshold K not gets succeeded to determine a path to destination, so a normal route discovery request (which is not conditioned as stored cache) is created at the source.

On the other hand, AODV caching node method has a constraint of taking fixed nodes unfairly to transmit packets. The unjustified transmission workload leads to reduce in networks existence time and MANET partitioning. In order to prevent unfaithfulness of node cache stored the researchers relieve nodes which stay in stored cache for so long [2]. They suggested a load adjustment technique AODV Node Cached (K: m−t) with the following two additional parameters – the minimum threshold value of data packets m transmitted during time t. If number of packets transmitted by a node M during time interval t is more, then AODV node cached (K: m − t) makes the node N from transmitting cache-constrained route requests for the same time period t. this period the break t, the node M now till transmitted packets also general un constrained route requests. But the forwarding load for M decreases since new paths with high possibility will avoid M.

Transmitting load adjustment procedure is not own taking due to the exact parameter values are identified through many practical observations. It implied that the values of factors need to be for every different situation. This method is on the fact that by leaving out rout request packets (RREQ) as per to the workload computations of every nodes, nodes can be excluded from routes. This method makes the use of the length of the packet queue in nodes and the left load that can be defined combination of the queue pending parameters numbers and residence time of packets in the queue. In simulation starting the minimum and maximum lengths of message queue and workload threshold are parameterized with some particular values. When a node found any RREQ data packets, it verifies the length of queue and computes mean value of two small thresholds values. And then, a node computes extra workload. If queue pending workload length is larger than the mean threshold value and pending workload is larger than workload threshold, it leave out RREQ packets.

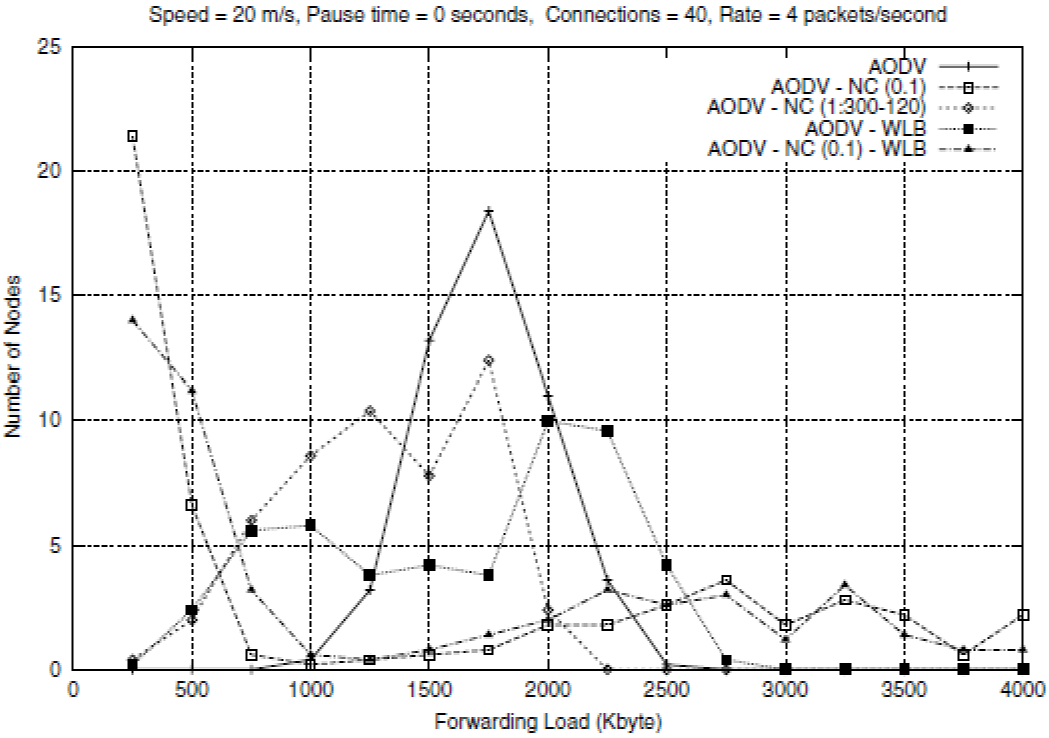


Figure: Load distribution with different number of nodes in network

In other situation the node transmits the RREQ packet to adjacent nodes. Node calculates new minimum required value of message queue length if pending load is larger than workload minimum threshold. Finally we can say that AODV node cached work load balancing method that is a combination of the adaptive workload balancing technique presented with the node cached routing protocol. The forwarding load balancing technique from cannot be mixed together with the adaptive workload balancing technique. The node caching technique is orthogonal to the adaptive workload adjustment technique that allows combine them without making major changes.

We applied workload-based adaptive load adjustment. Although it does not make it fine the original unfair load balancing limitation of AODV node caching method. On the other hand, it relieved nodes in high workload. In regards to routing efficiency, for various speeds AODV node caching technique shows better performance than AODV node caching technique with non-adaptive load balancing as well as AODV work load adjustment with node caching method. In high connection situations, AODV work load adjustment with node caching shows slight improvement in delivery ratio, relative overhead and end-to-end delay than the rest of the protocols.

From the energy efficiency point of view, AODV node cached (0.1) with work load adjustment showed the improved network results and AODV node cached method (1:300-120) showed the more long network existence duration by our new metrics. AODV node cache method (0.1) with work load balance improved throughput most as 20-35% more than general AODV. Moreover AODV node cached method (1:300-120) and AODV node cached workload adjustment method (0.1) makes use of the minimum amount of energy per packet to transmit to the target node and also to move directly to the next hop.

In the routing efficiency, AODV node cached (0.1) showed the best performance in relatively low workload situations. However, in more load scenarios, AODV node cached workload balance technique (0.1) and AODV node cached (1:300-120) showed high performance enhancement than AODV node cached (0.1). In multiple links case, AODV node cached (0.1:300-120) found the smallest routes those are near to the best hops and used the nearest hops to transmit data packets. Also, AODV node cached workload balance (0.1) improved the performance in high workload scenarios. As a result, both non adaptive and adaptive load adjustment method mixed with AODV node cached showed improved performance in energy saving as well as routing.

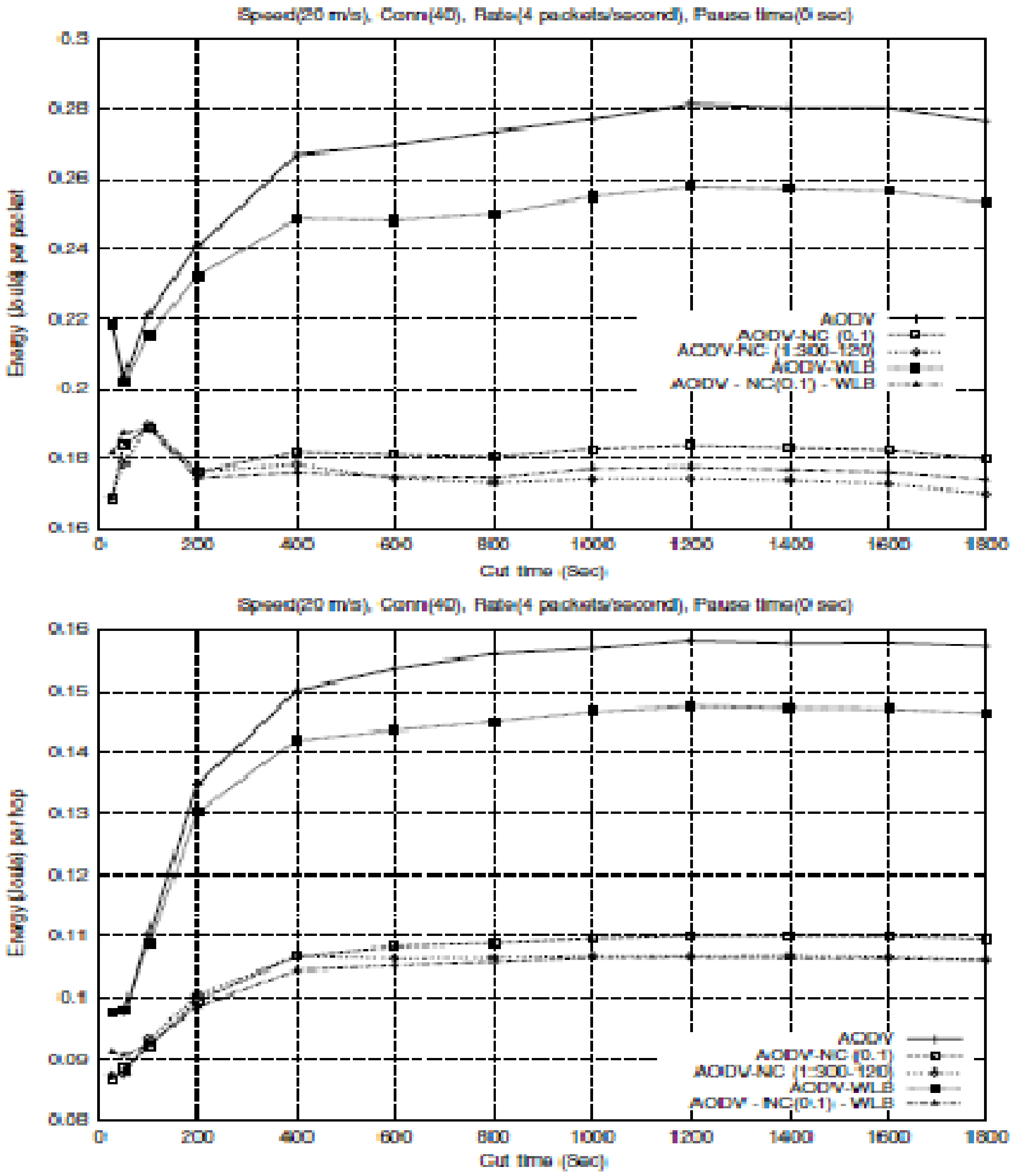


Figure : Energy consumed with different protocols

3.2 Trust based routing

In MANET hubs help one another in information steering. MANET functions admirably if the taking part hubs chip in with one another. It is illogical to expect that, all hubs taking an interest in an open MANET are agreeable and legitimate. For individual hubs it might be advantageous to be non-agreeable and egotistical. However non-collaboration, childishness and pernicious conduct of the taking interest hubs may come about into breakdown of a MANET. Trust based directing calculations expect to distinguish making trouble and non-coordinating hubs in the MANET. These calculations streamline the system execution by using trust commendable hubs in successful way and punishing non-helpful hubs. Here we look at trust based and cryptographic methodologies for executing security in MANET directing. The recent studies talks about outline issues in trust based directing conventions for MANET in subtle elements. The paper introduces a review on trust based steering conventions for MANET. The paper gives headings to future research in trust based steering for MANET [10].

Portable specially appointed system (MANET) is a distributed remote system in which hubs are versatile and foundation is not accessible for the correspondence. MANETs were initially acquainted with utilization in risky circumstances, for example, surge or seismic tremor influenced locales, salvage operations and battle zone. In shut or oversaw specially appointed system, every single portable hub have a place with a same power and are sent to work together with one another for a typical target. The commercialization of MANETs has as of late developed at speedier rate because of their versatility and increment of portable specialized gadgets like advanced mobile phones, PDAs, tablets, and other shrewd radio gadgets. In an open ad hoc system, hubs claimed by distinctive powers form a MANET and share their assets for worldwide network [26]. Henceforth security issues in an open ad hoc system are not quite the same as those in shut one. It's hard to accomplish security in a MANET because of show nature of correspondence, weakness of the connections, sporadic nature of network, unlucky deficiency of brought together administration power and powerfully evolving topology. Customary wired systems have devoted hubs for fundamental system operations like directing, approval, and system administration. These capacities are performed by individual hubs in the MANET. In MANET, the hubs have less handling force, little stockpiling, and constrained battery power. Due to these reasons, regular security calculations intended for customary systems are not viable in MANETs. Hence, new security methodologies are required for MANETs.

Information directing is an essential operation in MANET working. Essential supposition for the outline of directing conventions in MANET is each hub taking an interest in the correspondences is thought to be completely forthright and agreeable. Thus if a hub proclaims that it has a way to the destination, the case is thought to be valid. Correspondingly if a hub sends connection break status message, then that connection won't be utilized. Likewise if a bundle is sent to the neighbor why should gathered forward the parcel further, it is accepted that the neighbor will put best endeavors to forward the parcel. In spite of the fact that these presumptions and trust on alternate hubs empower the outline and execution of directing conventions, it adds weakness to distinctive sorts of Denial of Service (DoS) and bad conduct assault. Bad conduct in directing means difference from general steering and sending prerequisites. The bad conduct may be deliberate or unexpected. Mischief is inadvertent when a hub is damaged. Deliberate misconduct of the hub can either plan to take advantage over different hubs or ruin the system operations. In acting up hubs are ordered as egotistical hubs and ruinous hubs. The childish hubs try to expand their own increases at the loss of different hubs. The dangerous hubs act to corrupt the framework or individual hub execution. The illustrations of bad conduct are - not sending parcels, particular bundle dropping, parcel alteration, message manufacture, false course commercials, fake connection break messages, and timing mischief like including defer in the correspondence. As the extent of acting mischievously hubs expands, it comes about into diminished system throughput and system apportioning. At the point when hubs taking part in the system don't know how to believe one another, they obviously put stock in the great goals of different substances. Be that as it may, this does not work in circumstances where in malevolent hubs are partaking in the system. Setting up trust among dispersed system hubs is considered as viable instrument to manage hub's rowdiness. Research group has utilized trust for security as a part of an extensive variety of uses, which incorporates online administration choice, e-business, approval, access control, shared (P2P) systems, conveyed figuring, and pervasive processing. In most recent couple of years, utilizing trust for choice making as a part of MANET and sensor system directing is developed as a potential territory for examination.

Measurement of Trust Value:

The current trust administration frameworks classify trust as immediate trust and aberrant trust (some of the time called as reputation). The direct trust is gotten from hub's own encounters around a specific hub. Here, Trust or hub straightforwardly watches conduct of Trustee hub. The backhanded trust is registered from suggestions given by different hubs about the Trustee node. The direct trust is more useful if Trust or hub has continuous collaborations with the Trustee hub. In the event that the past collaborations are less or the conduct of hubs changes as often as possible, up and coming direct data may be rare. in such situation circuitous trust is more valuable. Two key components focus aberrant trust. The main is, when and from whom the Trustee can gather proposals. The second is, procedure for ascertaining roundabout trust quality utilizing the suggestions. Numerous current trust administration frameworks propose to discover hub n's trust on another hub as follows

$Tn,m = \alpha 1 \ast nTmS + \alpha 2 \ast nTmO$ (1)

Here nTmS is node n’s direct trust on m which is calculated using direct monitoring of the node m. The nTmO is indirect trust computed from other nodes’ recommendations about node m. The $\alpha 1$ and $\alpha 2$ are weighing factors such that

$\alpha 1 + \alpha 2 = 1$.

The values of $\alpha1$ and $\alpha2$ can be adjusted to give different weights to direct and indirect trust. The value of nTmS is evaluated by monitoring behavior of a neighbor :

nTmS = f(Φ,Ω, σ) (2)

The exact meaning of f can be implementation subordinate. Here Φ speak to movement measurements related with activity volume between hub n and m. The Ω speaks to activity measurements about information trustworthiness. The σ capacity speaks to general nature of system administration e.g. delays, throughput, great put and so on. It can be noticed that, complexity in monitoring criteria expands the likelihood of assault recognition; however it adds an additional weight to the system. Few trust management systems considers movement volume and overlooks respectability of the communication. Such systems are proficient as they don't utilize honesty checks which include additional handling weight. Be that as it may, these systems are not able to perceive unlawful parcel modifications made by intermediate hubs. It is ideal to utilize recommendations given by trustworthy hubs just. It serves to sift through malicious or inaccurate recommendations.

Conclusion based on trust based method [10]

As MANET working totally depends on collaboration and general conduct of taking part hubs, it is unsafe to trust all hubs clearly. Trust administration frameworks being security system cause extra handling and correspondence overhead to the hubs. These frameworks give intends to anticipating future activities of the hubs in light of past perceptions, and make utilization of such expectations for improving system execution. The trust administration frameworks additionally used to recognize and punish making trouble hubs. As of late scientists have investigated utilization of trust administration plans for operations like hub bunching, artful steering, multicast and vitality effective directing. In future, we may witness more novel utilizations of trust in MANET operations. These frameworks are not full confirmation, but rather serve as a danger administration instrument in an open MANET. It definitely serves to upgrade the productivity, unwavering quality and security of the correspondence. In spite of the fact that specialists are taking a shot at trust based directing plans since most recent couple of years; we trust that there is much degree in enhancing effectiveness and precision of trust based systems. The future work in trust based MANET may endeavor to give answers for the difficulties portrayed in the above segment. Taking after are other conceivable zones which can be investigated to gain ground towards hearty and proficient outline of trust based steering.

3.3 Multipath routing

The primary idea of AOMDV is to keep up various ways for every destination. At whatever point any way comes up short, the convention changes to another way without sitting tight for reinitiating the course disclosure as in AODV. The course disclosure is re-started just when very one of the ways to a destination come up short. The AOMDV convention discovers just connection disjoint different ways (no ways have basic connections between them). The AOMDV looks at every copy RREQ rather than dropping it. Since copy RREQ (Route Request Message) may have navigated through an alternate way. The AOMDV incorporates a field called first jump in order to separate between RREQ bundles navigating through connection disjoint ways and RREQ parcels crossing through non connection disjoint ways. At the point when the destination produces RREP bundles, each RREP takes diverse opposite courses to source. An option course is chosen just when the source hub gets a RRER message. Since the course choice technique is static, the AOMDV can't perform well in profoundly dynamic MANETs with continuous connection disappointments where the hub versatility causes the course breakage.

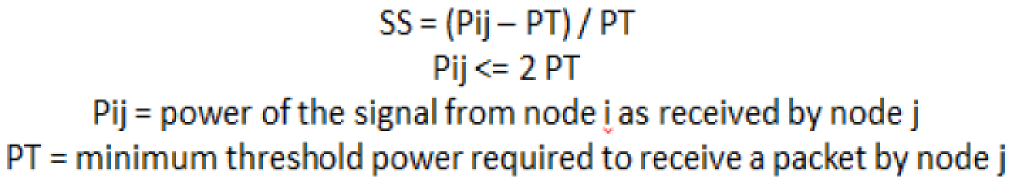
In light of the destinations used to enhance the execution of MANETs, the multipath directing conventions can be classified into five classes, to be specific, delay mindful, solid, least overhead, vitality effective and half breed steering conventions. The dependable multipath conventions experience the ill effects of high overhead in discovering the solid ways and the heap appropriating convention can bring about more defers. Along these lines, out of numerous multipath directing conventions accessible, it is elusive a convention that that is versatile and satisfying the greater part of the prerequisites of a proficient steering convention. The principle challenges in outlining a multipath steering convention are disclosure of numerous ways, determination of the best way from an arrangement of different ways and appropriation of the heap among the accessible various ways.

From the above writing it is watched that the greater part of the late ways to deal with select and keep up option courses in a multipath steering technique do not have a typical metric for course choice which chooses courses in light of diverse quality attributes of the courses. Selecting the sign quality alone as a metric won't give an extensive, predictable course. Despite the fact that the sign quality of a course is high, vicinity of a hub with low remaining battery vitality in the course can break the course. We propose an answer which stays away from stale courses by intermittent support and course exchanging, and gives quality courses to suite the different QoS prerequisites.

3.4 QoS aware Multipath routing

The shot of course disappointments is visit in exceptionally dynamic versatile MANETs. These are essentially because of mobility, limited battery vitality of the hubs and vicinity of lopsided connections. The current multipath steering routines register numerous courses amid the course disclosure prepare and are not looked after appropriately. Thus, the majority of the option ways can be stale which can bring about again course disappointments. So a productive course upkeep instrument is expected to enhance the nature of correspondence by occasionally redesigning the status of option courses. Here, notwithstanding the television of RERR messages for course support, we present a dynamic course upkeep technique which chooses a best course from an arrangement of courses and changes ahead of time to an option course before the breakage of the essential course in light of a metric which is computed from the three QoS attributes of the course [6].

1) Signal Strength: Every node computes its received signal strength from the upstream node as



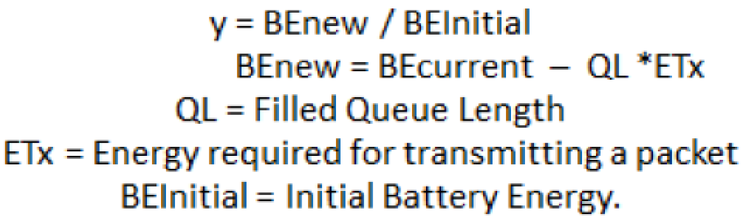
$$SS = (Pij - PT) / PT$$

$$Pij <= 2\ PT$$

Pij = power of the signal from node i as received by node j

PT = minimum threshold power required to receive a packet by node j

2) Remaining Battery Energy (y): Every node calculates its remaining battery energy (BE) which is the measure of complete life interval of the node as



$$y = BEnew / BEInitial$$

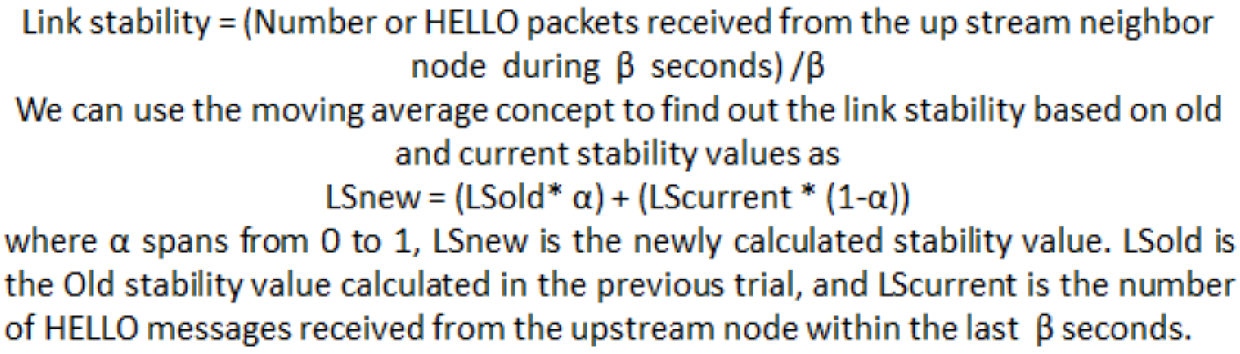
$$BEnew = BEcurrent - QL * ETx$$

QL = Filled Queue Length

ETx = Energy required for transmitting a packet

BEInitial = Initial Battery Energy.

3) Link stability: Link stability provides the information for existence of a particular route between two pair of nodes. In AODV or other classical proactive routing protocols each node broadcast a HELLO packet to all other neighbors. If a node is more stable at a point then others neighbor of that node will receive more HELLO messages.
So number of HELLO messages received from a particular node can be used as a measure for that node stability of that point.



Link stability = (Number or HELLO packets received from the up stream neighbor node during β seconds) /β

We can use the moving average concept to find out the link stability based on old and current stability values as

$$LSnew = (LSold * \alpha) + (LScurrent * (1-\alpha))$$

where α spans from 0 to 1, LSnew is the newly calculated stability value. LSold is the Old stability value calculated in the previous trial, and LScurrent is the number of HELLO messages received from the upstream node within the last β seconds.

Each source node periodically sends a special control message called update message (UPD) towards the destination through all the calculated routes and the message is travelled back through the same path to reach the source. The UPD message has three fields, namely, signal strength (x), remaining battery energy (y) and stability metric (z). Each node along the route updates the UPD message with the following values:

1. Minimum of the relative signal strength with which the node receives a packet from its upstream node and the x value in the UPD message.

2. Minimum of the remaining battery energy of the node and the y value in the UPD message.

3. Minimum of the stability value of the node and the z value in the UPD message. The source node computes the route selection cost (Ci) for each route i from the UPD messages received through that route.

$$Ci = (a\ xi + b\ yi + c\ zi) / (a+b+c)$$

where xi , yi , zi are x, y and z values from the UPD packet received through the path i. Symbols a, b and c are coefficients (weights) and their values can be selected according to the QOS requirements. The source node selects the path which has maximum C value as the primary path and the alternative paths are arranged in the descending order of C values. Whenever the C value of the current primary path becomes lower than the next alternative path, the primary path is switched. So at any point of time our method routes packets through the best available path. Since this method selects the best path based on quality, durability and stability, the overall throughput of the MANETs can be increased considerably. Moreover there are no costly computations in the method. Since the values of a, b and c are tunable according to the different service requirements, this protocol can be effectively used in diverse QOS aware applications like multimedia applications, event surveillance and military/battle field applications.

Using these metric for route selection we gets improvement on throughput and response of MANETs in case of QOS known factors. We must have much better results especially in case of query processing QOS aware networks [6]. Our proposal is to choose processing capacity also into consideration in route selection metric.

Chapter 4: Proposed Method

We propose to consider processing capacity also an important factor for route selection in case of data transmission in MANETs. Many researches are ongoing for MANET applications. Applications VANET (for vehicular communications), Query processing networks or sensor networks fast processing communication units (nodes) can influence their performance drastically. Practically all nodes do not have same processing speed.

Delay is indirectly proportional to processing capacity. A node has much processing capacity can measure a result and act fast compared to the node with low processing capacity.

Extending to multipath routing for QOS known networks we propose to use one more factor depend on processing speed of the node. Let's say for the network we have QoS delay parameter as d and P is the upstream node processing capacity. Every hop i in a particular route will maintain Pi as processing capacity of upstream neighbor node.

$$C = a*Xi + b*Yi + c*Zi + d*Pi$$

Where a, b, c and d are known QOS parameters

Pi = processing capacity of neighbor node of i

Multiple discovered paths are kept in increasing order of this metric C corresponds to that particular path in route queue. Once a link fails queue top route will be selected for transmission.

For realistic networks processing capacity of every node will be known. For the simulation packets delivery for every node with in constant time can be measured as processing capacity. We need to modify existing multipath routing algorithm accordingly.

Chapter 5: Implementation

Implementation is done using Network simulator 2.35

Simulation is done on 50 mobile nodes, with MAC 802.11 using multipath routing as suggested in previous section on 1000x1000 dimensions 150 seconds. TCL code for this simulation is given later.

5.1 Network Simulator

That is a event based powerful tool for network simulation. This tool runs on Linux or Unix platform. Using this tool we can measure and store data for any time of transmission.

Processing capacity is set for all the nodes while configuring nodes. Later source has been modified for route selection.

5.1.1 Components of NS

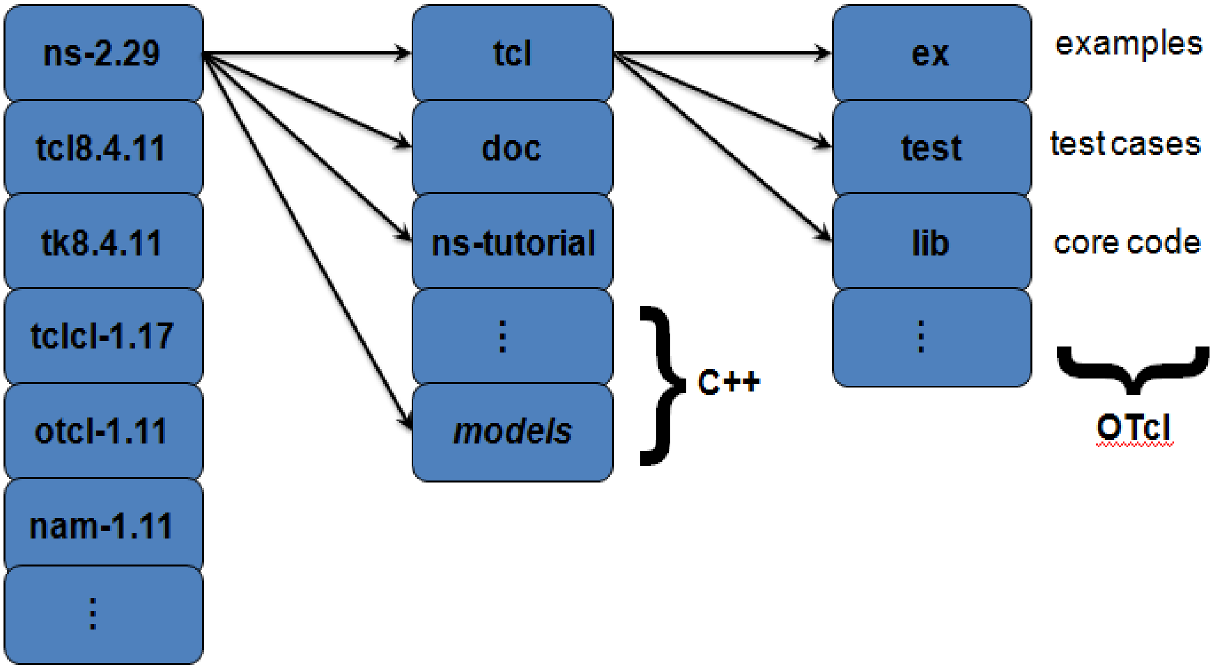


Figure: NS2 Components

5.1.1.1 NAM

It is a core component of Network simulator. It creates the animation of all network events. TCL script controls the NAM (network animation) using commands.

5.1.1.2 C++ models

Multiple modules are implemented in C++ at lower layer in NS2 stack. These models are various routing protocols, topologies and other utility components like priority queue, trace methods.

5.2 NS2 Setup

NS2 is object oriented based tools with multiple components in hierarchy. That is developed in C++ at the bottom. For implementing any simulation one driver script in TCL is executed. OTCL is a kind of TCL including object oriented feature.

Since NS2 is a UNIX tool operating this on windows environment requires Cygwin.

Setting up NS2 on Windows required cygwin installation with gcc, g++, tcl, octl, awk, xgraph packages.

5.2.1 Xgraph

Xgraph is a UNIX utility program that is used to create graph from the traces.

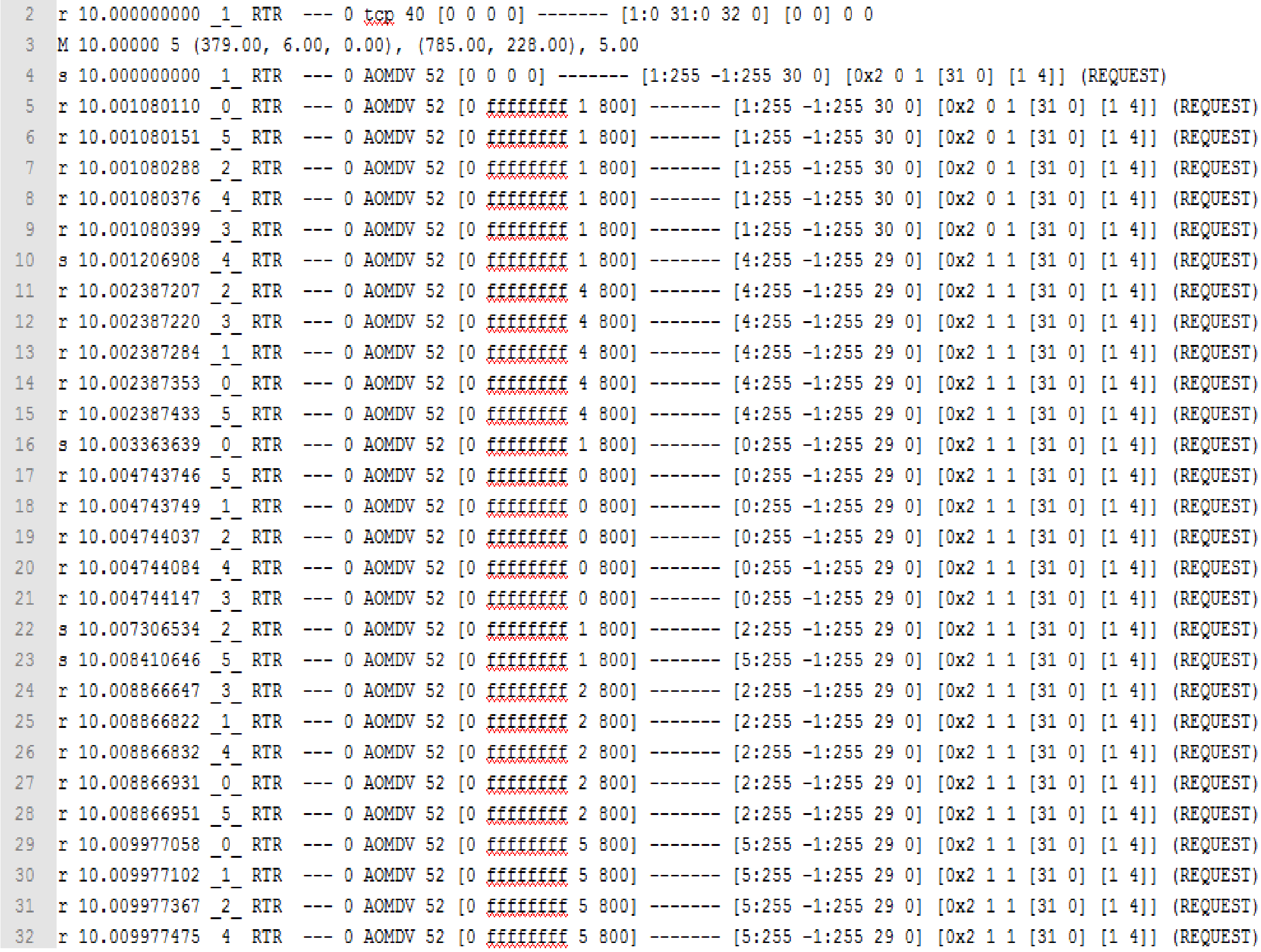
5.2.2 AWK

Awk are script files that create the information from the event stored traces. In awk script we can calculate any further information or parameter derived from the trace file (.tr)

Chapter 6: Results and Analysis

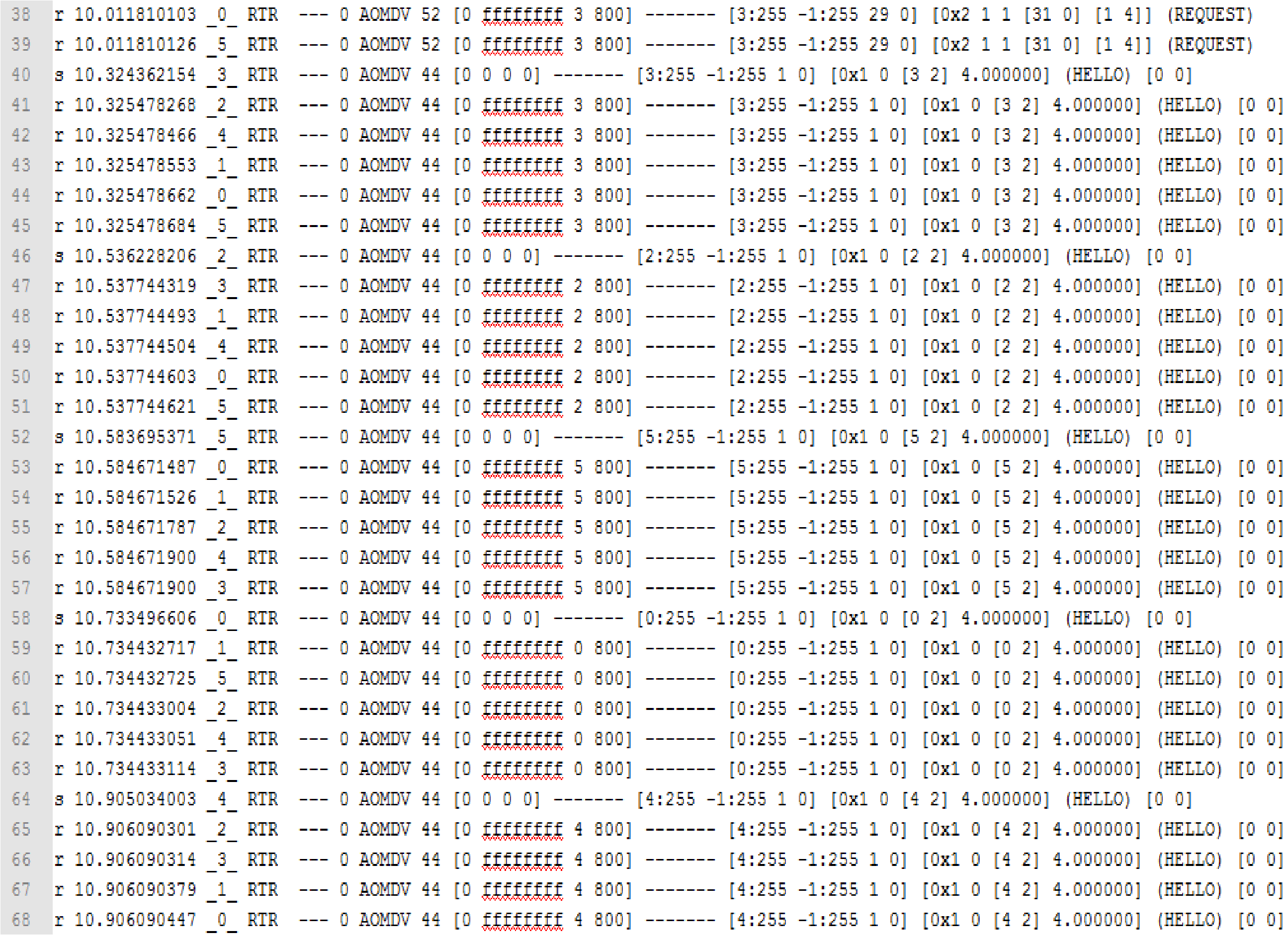
6.1 Traces for simulation

From the simulation and we found the traces are below. We can analyze the traces for the response of presented routing design. From the traces below are stored in trace(.tr) file while simulation happens. We can calculate the various quality and response parameter from these. Graph has been plotted based on these traces. We conclude that multiple path routing improves the performance for Mobile ad hoc network.



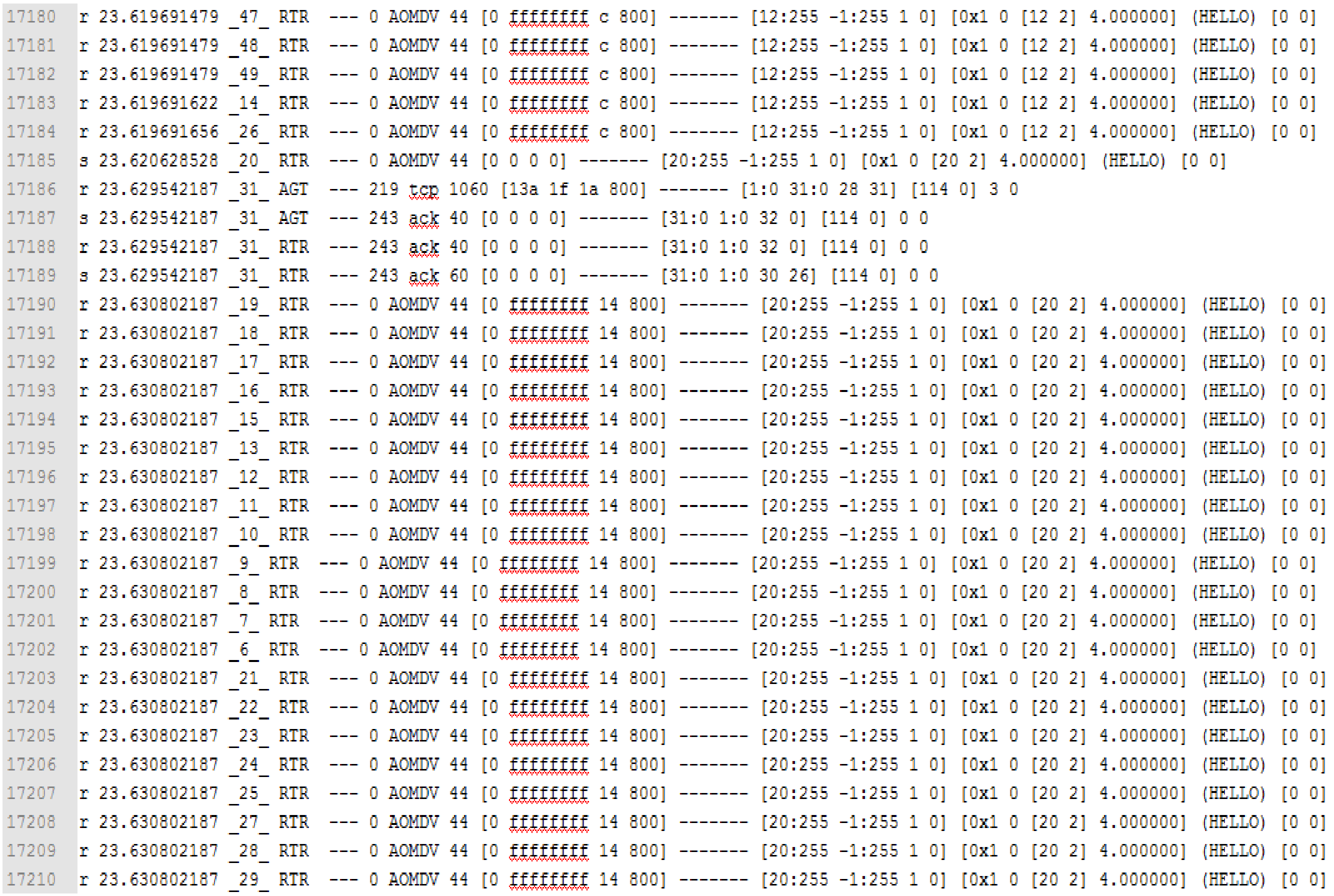
```
r 10.000000000 _1_ RTR  --- 0 tcp 40 [0 0 0 0] ------- [1:0 31:0 32 0] [0 0] 0 0
M 10.00000 5 (379.00, 6.00, 0.00), (785.00, 228.00), 5.00
s 10.000000000 _1_ RTR  --- 0 AOMDV 52 [0 0 0 0] ------- [1:255 -1:255 30 0] [0x2 0 1 [31 0] [1 4]] (REQUEST)
r 10.001080110 _0_ RTR  --- 0 AOMDV 52 [0 ffffffff 1 800] ------- [1:255 -1:255 30 0] [0x2 0 1 [31 0] [1 4]] (REQUEST)
r 10.001080151 _5_ RTR  --- 0 AOMDV 52 [0 ffffffff 1 800] ------- [1:255 -1:255 30 0] [0x2 0 1 [31 0] [1 4]] (REQUEST)
r 10.001080288 _2_ RTR  --- 0 AOMDV 52 [0 ffffffff 1 800] ------- [1:255 -1:255 30 0] [0x2 0 1 [31 0] [1 4]] (REQUEST)
r 10.001080376 _4_ RTR  --- 0 AOMDV 52 [0 ffffffff 1 800] ------- [1:255 -1:255 30 0] [0x2 0 1 [31 0] [1 4]] (REQUEST)
r 10.001080399 _3_ RTR  --- 0 AOMDV 52 [0 ffffffff 1 800] ------- [1:255 -1:255 30 0] [0x2 0 1 [31 0] [1 4]] (REQUEST)
s 10.001206908 _4_ RTR  --- 0 AOMDV 52 [0 ffffffff 1 800] ------- [4:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.002387207 _2_ RTR  --- 0 AOMDV 52 [0 ffffffff 4 800] ------- [4:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.002387220 _3_ RTR  --- 0 AOMDV 52 [0 ffffffff 4 800] ------- [4:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.002387284 _1_ RTR  --- 0 AOMDV 52 [0 ffffffff 4 800] ------- [4:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.002387353 _0_ RTR  --- 0 AOMDV 52 [0 ffffffff 4 800] ------- [4:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.002387433 _5_ RTR  --- 0 AOMDV 52 [0 ffffffff 4 800] ------- [4:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
s 10.003363639 _0_ RTR  --- 0 AOMDV 52 [0 ffffffff 1 800] ------- [0:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.004743746 _5_ RTR  --- 0 AOMDV 52 [0 ffffffff 0 800] ------- [0:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.004743749 _1_ RTR  --- 0 AOMDV 52 [0 ffffffff 0 800] ------- [0:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.004744037 _2_ RTR  --- 0 AOMDV 52 [0 ffffffff 0 800] ------- [0:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.004744084 _4_ RTR  --- 0 AOMDV 52 [0 ffffffff 0 800] ------- [0:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.004744147 _3_ RTR  --- 0 AOMDV 52 [0 ffffffff 0 800] ------- [0:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
s 10.007306534 _2_ RTR  --- 0 AOMDV 52 [0 ffffffff 1 800] ------- [2:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
s 10.008410646 _5_ RTR  --- 0 AOMDV 52 [0 ffffffff 1 800] ------- [5:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.008866647 _3_ RTR  --- 0 AOMDV 52 [0 ffffffff 2 800] ------- [2:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.008866822 _1_ RTR  --- 0 AOMDV 52 [0 ffffffff 2 800] ------- [2:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.008866832 _4_ RTR  --- 0 AOMDV 52 [0 ffffffff 2 800] ------- [2:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.008866931 _0_ RTR  --- 0 AOMDV 52 [0 ffffffff 2 800] ------- [2:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.008866951 _5_ RTR  --- 0 AOMDV 52 [0 ffffffff 2 800] ------- [2:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.009977058 _0_ RTR  --- 0 AOMDV 52 [0 ffffffff 5 800] ------- [5:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.009977102 _1_ RTR  --- 0 AOMDV 52 [0 ffffffff 5 800] ------- [5:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.009977367 _2_ RTR  --- 0 AOMDV 52 [0 ffffffff 5 800] ------- [5:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.009977475  4  RTR  --- 0 AOMDV 52 [0 ffffffff 5 800] ------- [5:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
```

Figure: Traces shown the root request starting from node 1



```
r 10.011810103 _0_ RTR  --- 0 AOMDV 52 [0 ffffffff 3 800] ------- [3:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
r 10.011810126 _5_ RTR  --- 0 AOMDV 52 [0 ffffffff 3 800] ------- [3:255 -1:255 29 0] [0x2 1 1 [31 0] [1 4]] (REQUEST)
s 10.324362154 _3_ RTR  --- 0 AOMDV 44 [0 0 0 0] ------- [3:255 -1:255 1 0] [0x1 0 [3 2] 4.000000] (HELLO) [0 0]
r 10.325478268 _2_ RTR  --- 0 AOMDV 44 [0 ffffffff 3 800] ------- [3:255 -1:255 1 0] [0x1 0 [3 2] 4.000000] (HELLO) [0 0]
r 10.325478466 _4_ RTR  --- 0 AOMDV 44 [0 ffffffff 3 800] ------- [3:255 -1:255 1 0] [0x1 0 [3 2] 4.000000] (HELLO) [0 0]
r 10.325478553 _1_ RTR  --- 0 AOMDV 44 [0 ffffffff 3 800] ------- [3:255 -1:255 1 0] [0x1 0 [3 2] 4.000000] (HELLO) [0 0]
r 10.325478662 _0_ RTR  --- 0 AOMDV 44 [0 ffffffff 3 800] ------- [3:255 -1:255 1 0] [0x1 0 [3 2] 4.000000] (HELLO) [0 0]
r 10.325478684 _5_ RTR  --- 0 AOMDV 44 [0 ffffffff 3 800] ------- [3:255 -1:255 1 0] [0x1 0 [3 2] 4.000000] (HELLO) [0 0]
s 10.536228206 _2_ RTR  --- 0 AOMDV 44 [0 0 0 0] ------- [2:255 -1:255 1 0] [0x1 0 [2 2] 4.000000] (HELLO) [0 0]
r 10.537744319 _3_ RTR  --- 0 AOMDV 44 [0 ffffffff 2 800] ------- [2:255 -1:255 1 0] [0x1 0 [2 2] 4.000000] (HELLO) [0 0]
r 10.537744493 _1_ RTR  --- 0 AOMDV 44 [0 ffffffff 2 800] ------- [2:255 -1:255 1 0] [0x1 0 [2 2] 4.000000] (HELLO) [0 0]
r 10.537744504 _4_ RTR  --- 0 AOMDV 44 [0 ffffffff 2 800] ------- [2:255 -1:255 1 0] [0x1 0 [2 2] 4.000000] (HELLO) [0 0]
r 10.537744603 _0_ RTR  --- 0 AOMDV 44 [0 ffffffff 2 800] ------- [2:255 -1:255 1 0] [0x1 0 [2 2] 4.000000] (HELLO) [0 0]
r 10.537744621 _5_ RTR  --- 0 AOMDV 44 [0 ffffffff 2 800] ------- [2:255 -1:255 1 0] [0x1 0 [2 2] 4.000000] (HELLO) [0 0]
s 10.583695371 _5_ RTR  --- 0 AOMDV 44 [0 0 0 0] ------- [5:255 -1:255 1 0] [0x1 0 [5 2] 4.000000] (HELLO) [0 0]
r 10.584671487 _0_ RTR  --- 0 AOMDV 44 [0 ffffffff 5 800] ------- [5:255 -1:255 1 0] [0x1 0 [5 2] 4.000000] (HELLO) [0 0]
r 10.584671526 _1_ RTR  --- 0 AOMDV 44 [0 ffffffff 5 800] ------- [5:255 -1:255 1 0] [0x1 0 [5 2] 4.000000] (HELLO) [0 0]
r 10.584671787 _2_ RTR  --- 0 AOMDV 44 [0 ffffffff 5 800] ------- [5:255 -1:255 1 0] [0x1 0 [5 2] 4.000000] (HELLO) [0 0]
r 10.584671900 _4_ RTR  --- 0 AOMDV 44 [0 ffffffff 5 800] ------- [5:255 -1:255 1 0] [0x1 0 [5 2] 4.000000] (HELLO) [0 0]
r 10.584671900 _3_ RTR  --- 0 AOMDV 44 [0 ffffffff 5 800] ------- [5:255 -1:255 1 0] [0x1 0 [5 2] 4.000000] (HELLO) [0 0]
s 10.733496606 _0_ RTR  --- 0 AOMDV 44 [0 0 0 0] ------- [0:255 -1:255 1 0] [0x1 0 [0 2] 4.000000] (HELLO) [0 0]
r 10.734432717 _1_ RTR  --- 0 AOMDV 44 [0 ffffffff 0 800] ------- [0:255 -1:255 1 0] [0x1 0 [0 2] 4.000000] (HELLO) [0 0]
r 10.734432725 _5_ RTR  --- 0 AOMDV 44 [0 ffffffff 0 800] ------- [0:255 -1:255 1 0] [0x1 0 [0 2] 4.000000] (HELLO) [0 0]
r 10.734433004 _2_ RTR  --- 0 AOMDV 44 [0 ffffffff 0 800] ------- [0:255 -1:255 1 0] [0x1 0 [0 2] 4.000000] (HELLO) [0 0]
r 10.734433051 _4_ RTR  --- 0 AOMDV 44 [0 ffffffff 0 800] ------- [0:255 -1:255 1 0] [0x1 0 [0 2] 4.000000] (HELLO) [0 0]
r 10.734433114 _3_ RTR  --- 0 AOMDV 44 [0 ffffffff 0 800] ------- [0:255 -1:255 1 0] [0x1 0 [0 2] 4.000000] (HELLO) [0 0]
s 10.905034003 _4_ RTR  --- 0 AOMDV 44 [0 0 0 0] ------- [4:255 -1:255 1 0] [0x1 0 [4 2] 4.000000] (HELLO) [0 0]
r 10.906090301 _2_ RTR  --- 0 AOMDV 44 [0 ffffffff 4 800] ------- [4:255 -1:255 1 0] [0x1 0 [4 2] 4.000000] (HELLO) [0 0]
r 10.906090314 _3_ RTR  --- 0 AOMDV 44 [0 ffffffff 4 800] ------- [4:255 -1:255 1 0] [0x1 0 [4 2] 4.000000] (HELLO) [0 0]
r 10.906090379 _1_ RTR  --- 0 AOMDV 44 [0 ffffffff 4 800] ------- [4:255 -1:255 1 0] [0x1 0 [4 2] 4.000000] (HELLO) [0 0]
r 10.906090447 _0_ RTR  --- 0 AOMDV 44 [0 ffffffff 4 800] ------- [4:255 -1:255 1 0] [0x1 0 [4 2] 4.000000] (HELLO) [0 0]
```

Figure: Traces for HELLO messages from various nodes



```
r 23.619691479 _47_ RTR  --- 0 AOMDV 44 [0 ffffffff c 800] ------- [12:255 -1:255 1 0] [0x1 0 [12 2] 4.000000] (HELLO) [0 0]
r 23.619691479 _48_ RTR  --- 0 AOMDV 44 [0 ffffffff c 800] ------- [12:255 -1:255 1 0] [0x1 0 [12 2] 4.000000] (HELLO) [0 0]
r 23.619691479 _49_ RTR  --- 0 AOMDV 44 [0 ffffffff c 800] ------- [12:255 -1:255 1 0] [0x1 0 [12 2] 4.000000] (HELLO) [0 0]
r 23.619691622 _14_ RTR  --- 0 AOMDV 44 [0 ffffffff c 800] ------- [12:255 -1:255 1 0] [0x1 0 [12 2] 4.000000] (HELLO) [0 0]
r 23.619691656 _26_ RTR  --- 0 AOMDV 44 [0 ffffffff c 800] ------- [12:255 -1:255 1 0] [0x1 0 [12 2] 4.000000] (HELLO) [0 0]
s 23.620628528 _20_ RTR  --- 0 AOMDV 44 [0 0 0 0] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.629542187 _31_ AGT  --- 219 tcp 1060 [13a 1f 1a 800] ------- [1:0 31:0 28 31] [114 0] 3 0
s 23.629542187 _31_ AGT  --- 243 ack 40 [0 0 0 0] ------- [31:0 1:0 32 0] [114 0] 0 0
r 23.629542187 _31_ RTR  --- 243 ack 40 [0 0 0 0] ------- [31:0 1:0 32 0] [114 0] 0 0
s 23.629542187 _31_ RTR  --- 243 ack 60 [0 0 0 0] ------- [31:0 1:0 30 26] [114 0] 0 0
r 23.630802187 _19_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _18_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _17_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _16_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _15_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _13_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _12_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _11_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _10_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _9_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _8_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _7_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _6_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _21_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _22_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _23_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _24_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _25_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _27_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _28_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
r 23.630802187 _29_ RTR  --- 0 AOMDV 44 [0 ffffffff 14 800] ------- [20:255 -1:255 1 0] [0x1 0 [20 2] 4.000000] (HELLO) [0 0]
```

Figure: Traces for Acknowledge message

6.2 Plotted graphs

We have plotted graph for number of Route discoveries, Packet delivery ratio and control overhead in case of proposed method named as QAMR (QOS Aware Multipath Routing). Those have been compared with the AODV measures.

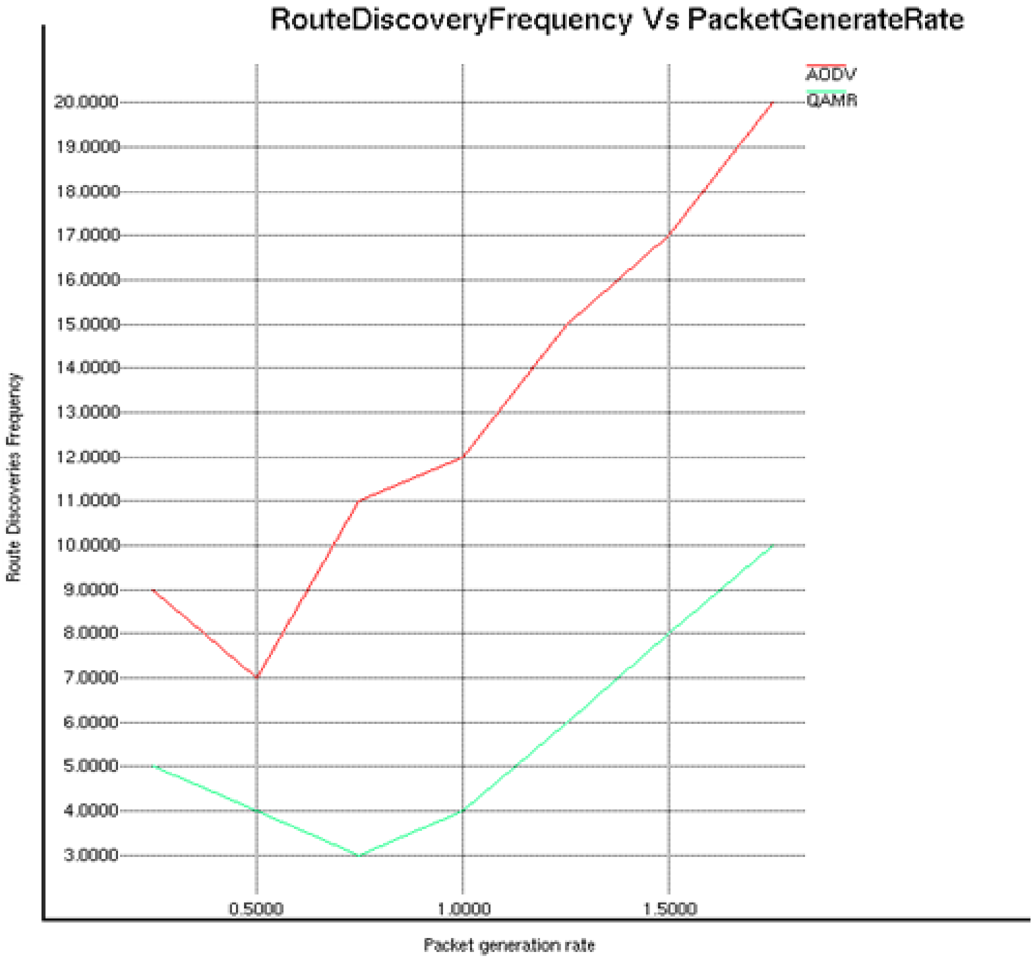


Figure: Number of Route discoveries Vs Packet generation per sec

In QOS aware Multipath Routing, route discoveries are accountably reduced compared to AODV. QAMR performed better than AODV.

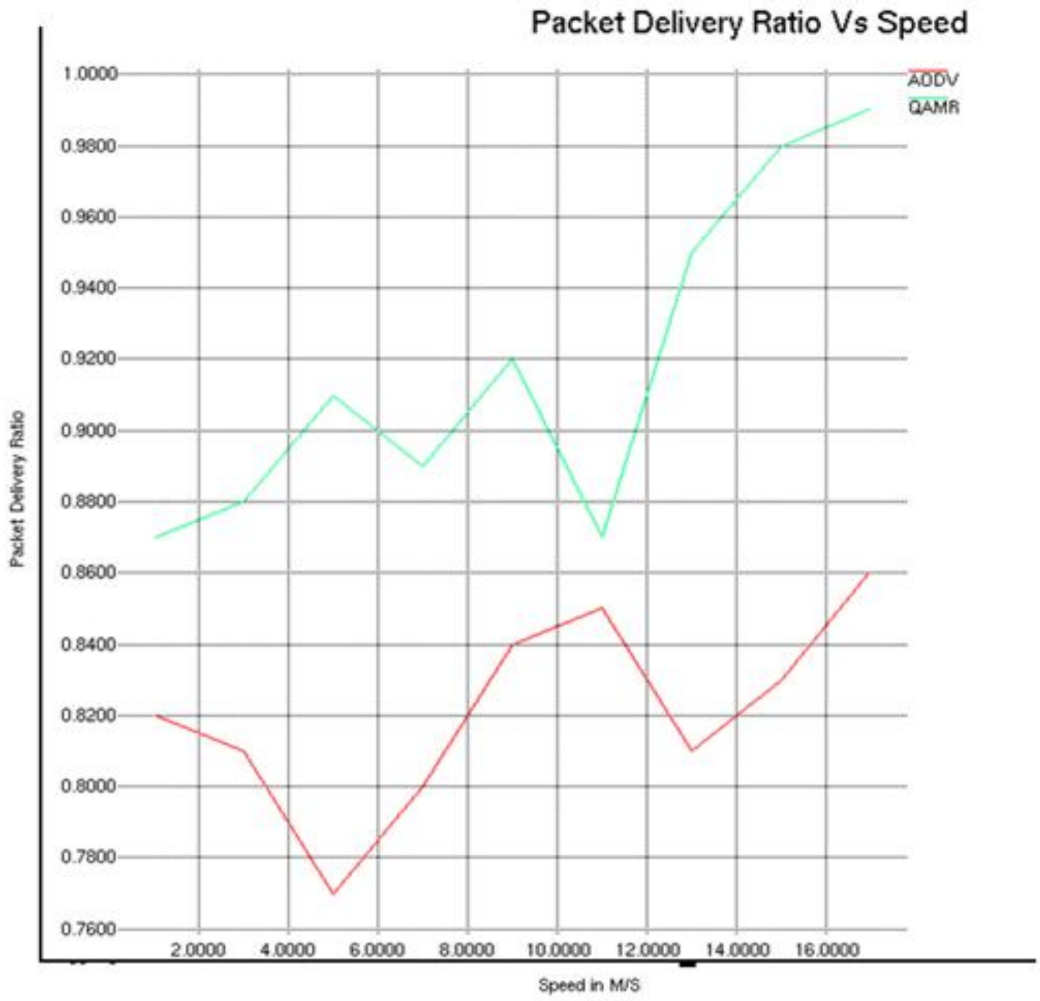


Figure: Packet delivery ratio Vs Speed of mobile nodes

Packet delivery ratio is more in case of QAMR with the mobility in nodes.

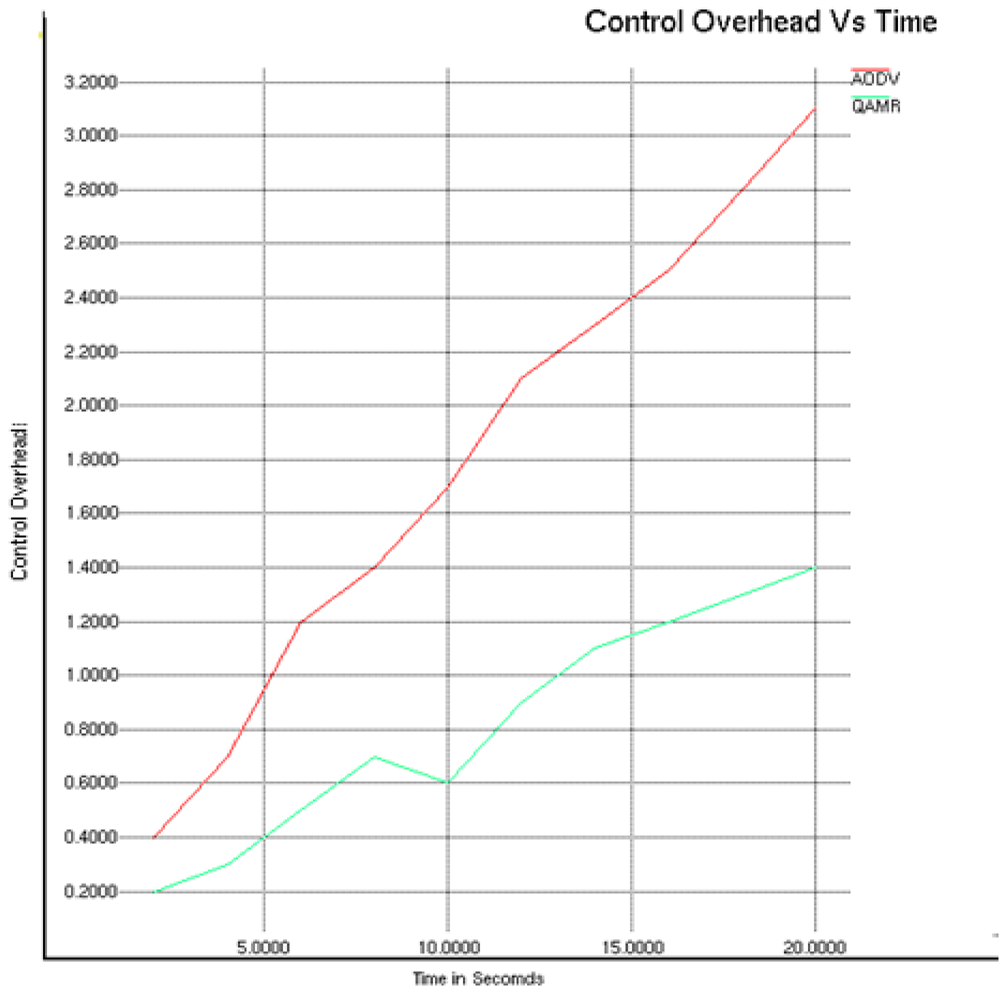


Figure: Control Overhead Vs Time

We can measure control overhead as number of RREQ, RERR and RREP messages. If we compared count of these messages with time for both AODV and QOS multipath proposed routing. It is very less in QAMR compared to AODV.

Chapter 7: Conclusion

Multipath routing is more efficient approach for Mobile Ad hoc networks. It provides improvement over high route failure cases with reduction in multiple route discoveries. For QOS known networks we can use a metric calculating from QOS parameters manipulated with network characteristics. Processing capacity is to be used for path selection metric. In future more research and simulation need to be done considering processing power of nodes as an important factor. More research has to be done to measure processing capacity of mobile nodes on the fly.

In Future more research can be done for relating processing capacity with energy efficiency. Considering there relation more efficient routing protocol can be designed.

We can have better improvements on MANETs widely on emergency services and query based networks. Apart from designing routing procedure for these networks more work further need to be done on their security side.

Any loophole in security either authenticity or information leakage could lead major problems for MANETs. Major future work can also be done in this area.

References

1. Classification of Ad Hoc Routing Protocols: Petteri Kuosmanen
2. Energy Efficiency of Load Balancing in Manet Routing Protocols: Sunsook Jung, Nisar Hundewale, Alex Zelikovsky, Proceedings of Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and First ACIS International Workshop on Self Assembling Wireless Networks(SNPD/SWAN'05), IEEE 2005
3. Performance Evaluation of Energy Consumption for AODV and DSR Routing Protocols in Manets: Mehdi Barati, Kayvan Atefi, Farshad Khosravi, Yashar Azab Daftari, International Conference on Computer and Infromation Science(ICCIS), 2012
4. An Overview of Routing Protocols in Mobile Ad-Hoc Networks: Dr. S. S. Dhenakaran, A. Parvathavarthini
5. Power Aware QoS Multipath Routing Protocol for Disaster Recovery Networks: S. Santhi, Dr. G. Sudha Sadasivam, International Journal of Wireless and Mobile Networks(IJWMN) Vol. 3, No. 6, December 2011
6. Dynamic Multipath Routing for MANETs – A QoS Adaptive Approach: Manu J. Pillai, M. P. Sebastian, S. D. Madhukumar, IEEE 2013
7. A Node Disjoint Multipath Routing Method Based on AODV Protocol for MANETs: C. Lal, V. Laxmi, M. S. Gaur, in Proceedings of 26th IEEE International Conference on Advance Information Networks and Applications, March 2012
8. Node Caching Enhancement of Reactive Ad Hoc Routing Protocol: Sunsook Jung, Nisar Hundewale, Alex Zelikovsky, WCNC'05, 2005

9. A Mutipath on-Demand Routing with Path Selection Entropy for Ad Hoc Networks: Baolin Sun, Chao Gui, Qifei Zhang, Bing Yan, Wei Liu, 9th International Conference for Young Computer Scientists, IEEE 2008
10. Design Issues in Trust Based Routing for MANET: Sandeep A. Thorat, P. J. Kulkarni, 5th ICCCNT – 2014, IEEE – 33044
11. Trust Based Minimum Cost Opportunistic Routing for Ad Hoc Networks: Wang Boa, Huang Chuanhea, Li Layuanb, Yang Wenzhonga, Journal of Systems and Software, Vol. 84, Issue 12, December 2011