

Acc No. (PNS) T-1205
1-1205 ★ MW

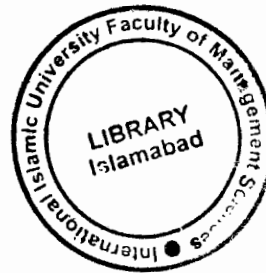
Scalability Of Zone Routing Protocol Extension Of Mobile Ad-hoc Networks

T01205



Presented By

M. Bilal Nazir
73-CS/MS/02



Supervised By

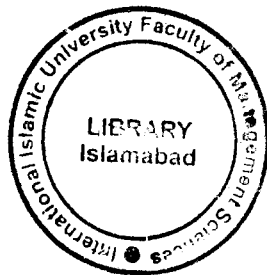
Dr. S. Tauseef-ur-Rehman

Head Department Of Telecommunication
Faculty Of Applied Sciences
International Islamic University Islamabad

Dr. M. Sikander Hayat Khiyal

Head Department Of Computer Science
Faculty Of Applied Sciences
International Islamic University Islamabad

**Department of Computer Science
Faculty of Applied Sciences
International Islamic University Islamabad (2004).**



Dedicated
to
My Parents

Who motivated me to achieve this position in life and
helped me in all fields of life.

International Islamic University, Islamabad
Department of Computer Science

FINAL APPROVAL

24, 06, 2004

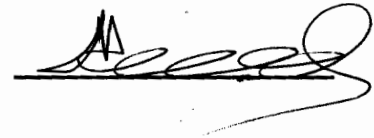
It is certified that we have read the project report submitted by **Mr. Muhammad Bilal Nazir** and it is our judgment that this project is of sufficient standard to warrant its acceptance by International Islamic University, Islamabad for the Degree of Master of Sciences in Computer Science.

COMMITTEE

External Examiner

Prof. Dr. Qasim Rind

Preston University Islamabad



Internal Examiner

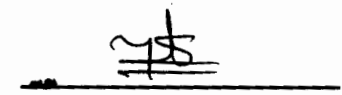
Mr. Muhammad Sher

Assistant Professor

Department Of Computer Science

Faculty of Applied Sciences

International Islamic University, Islamabad



Supervisors


Dr. S.Tauseef-ur-Rehman

Head

Department of Telecommunication,

Faculty of Applied Sciences

International Islamic University, Islamabad.



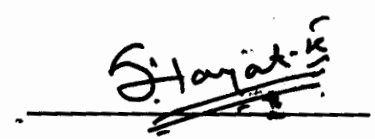
Dr. M. Sikander Hayat Khiyal

Head

Department of Computer Science,

Faculty of Applied Sciences

International Islamic University, Islamabad



Acknowledgments

All praise to Almighty Allah, the most merciful and compassionate, without His help and blessing I was unable to complete the project.

This project could not have come about without the help, encouragement and guidance of the following persons.

To my respected project supervisor and teacher Dr. S. Tauseef-ur-Rehman and Dr M.Sikander Hayat Khiyal. Without their precious guidance and help I could never be able to develop such software.

To my friends Arshad, Maroof, Hanan, Fahad, Rizwan and Jehanzaib their kind and really professional guidance remained with me throughout the way.

And finally to my brother Tahir Nazir and sisters who provided me suitable support both morally and financially.

Muhammad Bilal Nazir

73-CS/MS/02

Declaration

I hereby declare that this software, neither as a whole nor as a part thereof has been copied out from any source. It is further declared that I have developed this software and the accompanied report entirely on the bases of my personal efforts made under the sincere guidance of my teachers. If any part of this system is proved to be copied out I shall stand by the consequences. No portion of the work presented in this report has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

Muhammad Bilal Nazir

73-CS/MS/02

Project In Brief

Project Title: Scalability Of Zone routing Protocol
Extension of Mobile Ad-Hoc Networks

Undertaken By: Mr. Muhammad Bilal Nazir

Supervised By: **Dr S. Tauseef-ur-Rehman**
Head Department Of Telecommunication, IIUI
Dr M. Sikander Hayat Khiyal
Head Department Of Computer Science, IIUI

Date Started: 18th Sep 2003

Date Completed: 18th Jun 2004

Tools: Network Simulator-2 under the environment
of Cygwin, Visual C++ 6.

Operating System: Windows 2000(Professional).

System Used: Pentium IV.

Abstract

This report explains the problems and challenges faced in the routing of mobile ad-hoc networks. Further it explains the Zone Routing Protocol (ZRP), which is a hybrid approach of routing it, combines the advantages of the proactive and reactive approaches. Further, we discuss the problem of routing in ad-hoc networks and the motivation of ZRP. We describe the architecture of ZRP, which consists of three sub-protocols. The ZRP is implemented and on the basis of the output results we describe the advantages of ZRP.

Table of Contents

1. Introduction.....	1
1.1 Overview of ad-hoc networks.....	1
1.1.1 Link layer.....	1
1.1.2 Ad hoc routing.....	2
1.1.3 Network layer.....	2
1.1.4 Transport layer.....	2
1.1.5 Application.....	2
1.1.6 Mobility.....	3
1.1.7 Topology and environment.....	3
1.1.8 Network traffic.....	3
1.2 Routing in adhoc networks.....	4
1.3 Current Routing Protocols.....	5
1.3.1 Link State Routing (LSR).....	5
1.3.2 Distance Vector Routing (DVR).....	6
1.3.3. Destination-sequenced Distance Vector Routing (DSDV).....	6
1.3.4. Ad hoc On Demand Distance Vector Routing (AODV).....	6
1.3.5. Temporally-Ordered Routing Algorithm (TORA).....	7
1.3.6. Dynamic Source Routing (DSR).....	7
1.3.7. Zone Routing Protocol (ZRP).....	7
1.3.8. Basic Spine Routing (BSR).....	8
1.3.9. Hierarchical Spine Routing (HSR).....	8
1.3.10. Wireless Routing Protocol (WRP).....	9
1.3.11. Linked Cluster Architecture (LCA).....	9
1.3.12. Virtual Subnets.....	9
1.4 Scalability in Ad-hoc Networks.....	9
1.5 Issues in scalability.....	11
1.5.1 Traffic Characteristics.....	11
1.5.2 Fault Model.....	13
1.5.3 Utilizing Existing Infrastructure.....	13
1.5.4 Data Replication.....	14
1.5.5 Id Assignment.....	15
1.5.6 Id Change.....	15
1.5.7 Preliminary Ideas.....	16
1.6 Objective.....	16
2. Literature Survey.....	17
2.1 Zone Routing Protocol.....	17
2.2 Architecture.....	17
2.3 Routing.....	20
2.4 Route maintenance.....	21
2.4.1 Example.....	22
2.5 Query-control mechanisms.....	23
2.6 Query detection.....	24
2.7 Early termination.....	25
2.8 Random query-processing delay.....	26
2.9 Caching.....	26

3. Zone Routing Protocol.....	28
3.1 Introduction to ZRP	28
3.2 Intrazone Routing Protocol (IARP)	33
3.3 Interzone Routing Protocol (IERP).....	34
3.4 Bordercast Resolution Protocol (BRP)	35
3.5 Examples.....	37
3.6 BRP	38
3.7 Stationary nodes in a dense network.....	39
3.8 Mobile nodes without a stationary fixpoint	41
3.9 Performance	43
3.10 Performance gain on the example of Bordercasting	44
4.0 Implementation	46
4.1 Ns-2 and Wireless Simulations	46
4.3 Zone Routing Protocol.....	46
4.4 System Requirements.....	47
4.5 Design of the ZRP Module for ns-2.....	48
4.6 Overview of the code	50
4.6.1 Brp.h.....	51
4.6.2 Iarp.h	51
4.6.3 Iarp_efficient.h.....	52
4.6.4 Ierp.h	52
4.6.5 Ndp.h.....	52
4.6.6 Zrp.h.....	52
5.0 Results.....	58
5.1 Traffic measurements.....	58
5.2 Determining the routing zone radius.....	59
Fig 5.1 Zone Traffic Per Node.....	60
5.3 Zone sizing schemes	60
5.4 Comparison of Routing Protocols.....	61
5.4.1 Route Quality	62
5.4.2 Protocol Accuracy.....	62
5.4.3 Protocol Efficiency.	63
5.5 CONCLUSION.....	63
5.5.1 Output Graphs on performance of ZRP	65
6.0 Testing.....	70
6.1 Black Box Testing.....	70
6.2 White Box Testing	71
6.3 Unit Testing	72
Appendix A.....	73
Reference & Bibliography	76

Chapter 1

Introduction

1. Introduction

An ad hoc network is a network where the topology is unplanned and possibly not even fixed. This is a quite loose definition that can be more tightly specified by adding the words wireless, meaning that the nodes use wireless interfaces to communicate, mobile, meaning that the nodes can move around, and multi-hop, meaning that the nodes can communicate with other nodes indirectly by using closer nodes as relays.

Often the shorter-term mobile ad hoc network, or MANET, is used to specify a wireless mobile multi-hop ad hoc network. The term MANET will be used throughout this report with this meaning.

1.1 Overview of ad-hoc networks

Several different parts, or layers, are needed for an ad hoc network to work. There must be a way to physically transport data between nodes and there must be one or more protocols to ensure that the data arrives at the correct node and the correct application.

These layers interact with each other in a well-defined manner. A layer can pass data to another layer, which in turn can add or remove its own layer specific information and pass the result on to the next layer. The other layers usually ignore the details of how each layer works.

However, characteristics like delays and added packet overhead can affect the performance of other layers in more or less subtle ways.

1.1.1 Link layer

To build a network there must be some way to transmit data between the nodes. For MANETs this is usually done with wireless interfaces. These interfaces can logically be divided into two parts: the physical layer, which converts data into physical signals (such as radio waves) and back again, and the link protocol layer, which handles details

like who the source and destination are and when each of the nodes are allowed to send. It is also possible for the link layer to detect transmission errors and do retransmissions of link frames.

1.1.2 Ad hoc routing

In a multi-hop environment there must be some way of finding a route between two nodes. This is done with an ad hoc routing protocol. Often the routing protocol operates below the network layer, but still has knowledge about it.

1.1.3 Network layer

The network layer is responsible for delivering packets based on the nodes network addresses and for knowing which nodes are on the same subnet. Since compatibility with Internet applications is often desired, most MANETs are designed with the TCP/IP stack in mind. A MANET is usually treated as a single IP subnet. This means that on the IP level all nodes are considered to be directly connected to each other. The ad hoc routing is hidden by the ad hoc routing protocol and normally normally can not be seen by higher layer protocols.

1.1.4 Transport layer

While the network layer is mainly concerned with getting data from one node to another, the responsibility for sorting incoming data and passing it on to the correct application falls on the transport layer. The transport layer can also add features like detection of transmission errors, requesting retransmission of missing or bad packets, and sorting incoming data so that it is passed on to the application in the same order as it was sent. Note that the transport layer error detection and retransmission is usually unaware of link layer error detection and retransmission mechanisms.

1.1.5 Application

Applications that use the network usually have no idea about what the network's topology is. They are usually aware of who they are communicating with, but the details of how the data is transmitted are generally invisible to them. This is good, since it allows programmers to write applications without worrying about all the different lower layers that could or do exist. On the other hand, it means that

applications can not easily adapt their behavior based on lower layer conditions. Also, there are some other factors that can affect the application layer user's perceived performance in different ways. Even though they are not directly involved in the data transmission they must be considered when studying MANETs, because of their effects on the data transmission.

1.1.6 Mobility

One of the most prominent useful features of MANETs is that the nodes can move around freely. On the other hand, this also generates a lot of problems. The nodes can move in and out of range of each other and in order to maintain connectivity to specific nodes the ad hoc routing protocol must find new routes through the network. Even if the nodes that are communicating with each other are stationary the movement of other nodes can still affect them. In some cases a node or a group of nodes can move such that there is no possible route between certain other nodes in the network. In this case the network is said to be partitioned and no communication is possible between nodes in the different partitions.

1.1.7 Topology and environment

The physical location of the nodes can also affect their performance. If a lot of nodes are located in a small area, there will be a greater amount of contention for the available transmission capacity in this region, but there is a decrease in the probability of the network partitioning. The surrounding environment, for example metal walls or strong electromagnetic fields, can affect the link layer performance by hindering radio waves or by otherwise interfering with the transmissions.

1.1.8 Network traffic

A network with a high traffic load will naturally have a higher probability of longer delays before a node can transmit and, with link layer protocols without collision avoidance, a higher probability of collisions. The characteristics of the traffic, such as how well it is distributed over the network, or if it has a constant data rate or comes in shorter bursts, can also affect the throughput rate. Collision probability is dependent on MAC protocol details; there are some protocols, which are collision free.

1.2 Routing in adhoc networks

Various design choices for ad hoc networks are:

a) Flat versus hierarchical architectures

b) Proactive versus reactive schemes.

In a hierarchical architecture, the detail of the network topology is concealed by aggregating nodes into clusters and clusters into super clusters and so on. Some nodes, such as cluster heads and gateway nodes, have higher computation and 3-communication burden than other nodes. Hence, the mobility management is complicated. The network reliability may also be affected due to single point of failure of these critical nodes. However, the routing messages may only have to propagate within a cluster; thus, the amount of globally propagated routing messages is small. The multilevel hierarchy also reduces the storage requirement and the communication overhead of a large wireless network. On the contrary, in a flat architecture, all nodes carry the same responsibility. Therefore, the mobility management is simple. However, this flat architecture is not bandwidth efficient because the routing messages have to propagate globally throughout the network. The scalability gets worse when the number of nodes increases.

In a proactive scheme, every node continuously maintains the complete routing information of the network. When a node needs to forward a packet, the route is readily available; thus, there is no delay in searching for a route. However, for a highly dynamic topology, the proactive schemes spend a significant amount of scarce wireless resource in keeping the complete routing information current. On the other hand, in a reactive scheme (or source initiated scheme), nodes only maintain the routes to active destinations. A route search is needed for every new destination. Therefore, the communication overhead is reduced at the expense of delay due to route search. Furthermore, the rapidly changing topology may break an active route and cause subsequent route search. The protocols such as link state routing (Open Shortest Path First) and distance vector routing (Bellman-Ford) are flat and proactive. These routing protocols were never designed to work in a mobile network. They do not converge fast enough to the rapidly changing topology. In addition, they tighten

the scarce wireless resource because a significant amount of communication overhead is required. The performance will also degrade when the number of nodes increases. Other routing protocols use hierarchical architecture or source-initiated scheme to reduce the communication overhead, alleviate the scalability problem and improve the convergence of the protocols.

We propose a "peer-to-peer" hierarchical routing protocol that divides the network into non-overlapping zones. The detail of the network topology is concealed by aggregating nodes into zones. A packet is forwarded by specifying the hierarchical address – zone ID and node ID - of a destination node in the packet header. Each node knows the low level (node level) topology about node connectivity *within its zone* and the high level (zone level) topology about zone connectivity of *the whole network*. Unlike other hierarchical protocols, there are no cluster heads in this protocol. The high level topological information is distributed to all nodes (i.e. in a "peer-to-peer" manner). Therefore, mobility management is simple and single point of failure can be avoided. The protocol is proactive if the destination is within the same zone of the source. Otherwise, it is reactive because a location search is needed to find the zone ID of the destination. Routing is done by specifying the zone ID and the node ID of the destination, instead of specifying the path containing the nodes between the source and the destination as in dynamic source routing and zone routing protocol, in the packet header. Therefore, intermediate link breakage, due to node mobility, will not cause any subsequent location search; the proposed protocol is adaptable to the changing topology.

1.3 Current Routing Protocols

This section describes some of the recently proposed routing protocols for mobile ad hoc networks. They are summarized as follows:

1.3.1 Link State Routing (LSR)

Each node broadcasts a link state packet (LSP) containing a list of its neighbors to all other nodes. With the full topological information, each node can find multiple paths to another node. However, large communication overhead is required to maintain the full topological information. As the network topology changes

frequently, relatively large bandwidth is required for globally propagated routing messages. Due to long propagation delay, short-term routing loops may form. Therefore, LSR is not suitable for mobile ad hoc networks.

1.3.2 Distance Vector Routing (DVR)

Each node sends a distance vector to its immediate neighbors. The vector contains a list of costs to all other nodes in the network. Only one path between each source/destination pair is available for routing. DVR suffers from slow convergence and loop formation. These loops are formed because nodes choose their next-hops in a distributed fashion based on possibly stale information. Furthermore, depending on the location of link change, routing messages may have to be propagated to a large number of nodes.

1.3.3. Destination-sequenced Distance Vector Routing (DSDV)

DSDV prevents loop formation and "counting to infinity" problems by tagging each route table entry with a destination sequence number so that stale information can be deleted. However, DSDV still suffers from slow convergence because a router cannot pass on its routing information until it finishes recomputing its distance vector. In contrast, LSR converges faster because a router can recognize a new LSP and forward it before recalculating route. Similar to DVR, only one path is available between each source/destination pair.

1.3.4. Ad hoc On Demand Distance Vector Routing (AODV)

When a route is needed, a node broadcasts a route request message. The response message is then echoed back once the request message reaches the destination or an intermediate node that contains a fresh route to the destination. For each route, a node also maintains a list of those neighbors actively using the route. A link breakage causes immediate link failure notifications to be sent to the affected neighbors. Similar to DSDV, each route table entry is tagged with a destination sequence number to avoid loop formation. Moreover, nodes are not required to maintain routes that are not active. Thus, wireless resource can be effectively utilized. However, since flooding is used for route search, communication overhead for route search is not scalable for large networks. As route maintenance considers only the link

breakage and ignores the link creation, the route may become non-optimal when network topology changes. Subsequent global route search is needed when the route is broken.

1.3.5. Temporally-Ordered Routing Algorithm (TORA)

A node broadcasts a route query when it needs to set up a route to a destination. Based on the query/reply process, a sequence of directed links leading from the source to the destination is formed. It is a directed acyclic graph (DAG) rooted at the destination. The DAG is self-adapting to the topological changes in the network. The routing messages are generally localized to a small set of nodes near the change. The generation of far-reaching routing message is decoupled from the rate of topological changes. Upon detecting any network partition, all links separated from the DAG are undirected to erase invalid route loops. TORA provides multiple paths to a destination and ensures they are loop-free. This is a destination-oriented protocol in which logically separate version of the protocol is run for each destination in active communication. However, after the DAG creation, new links would not be considered unless the DAG becomes disconnected. Therefore, the route may become non-optimal. In addition, communication overhead for route creation is not scalable because flooding is used.

1.3.6. Dynamic Source Routing (DSR)

A source broadcasts a request in the network to find a path to the destination. Response(s) is echoed back when the request reaches the destination or an intermediate node with a cached route to the destination. Multiple paths may be available for routing. Similar to AODV, the communication overhead for route search is not scalable. Since a complete route is included in every message, message header is large. Moreover, a new route has to be found when all previously known route(s) is broken.

1.3.7. Zone Routing Protocol (ZRP)

ZRP is a hybrid of DVR and DSR. A routing zone is defined for each node that includes nodes at some predefined number of hops from the node. Each node is required to know only the topology of its routing zone; i.e. the routing messages are

only propagated locally. The route discovery requires a relatively small number of query messages as these messages are routed only to "peripheral" nodes, omitting all the nodes within the routing zones. This method is called bordercasting. However, the route is unstable because the peripheral nodes can move out of the routing zones. Subsequent route discoveries may be needed.

1.3.8. Basic Spine Routing (BSR)

A distributed algorithm is used to find an approximation to the minimum connected dominating set (MCDS) of an ad hoc network. Topological information is then gathered from non-MCDS nodes and propagated to all MCDS nodes. Each MCDS node runs an all-pair shortest path algorithm. A non-MCDS node can then request the route information to a destination from an adjacent MCDS node. Since MCDS nodes propagate, store and run shortest path algorithm on the topological information, their communication, storage and computation burdens are heavy. Moreover, mobility management is complicated by the existence of MCDS nodes.

1.3.9. Hierarchical Spine Routing (HSR)

HSR is a two-level routing architecture in order to solve the scalability problem happened in BSR. The network is divided into clusters. Each cluster head stores the topology of a cluster graph, the cluster membership table (it consists of cluster ID for each node) and the list of local boundary nodes (it shows clusters reachable for each boundary node). Within each cluster, BSR is used. Between clusters, LSR is run on the cluster graph topology. Using distinct clustering, the impact on topological change is localized. Moreover, the amount of time for spine construction and maintenance is reduced since clusters essentially run in parallel. The communication and the storage overhead in some nodes can be greatly reduced. However, cluster heads, boundary nodes and spine nodes take on heavy responsibilities. Therefore, the communication, storage and computation burdens are heavy in those nodes. Mobility management becomes complicated and the reliability of the network decreases due to single point of failure.

1.3.10. Wireless Routing Protocol (WRP)

Each node maintains the shortest path spanning trees (SST) of its neighbors and uses that information to generate its own SST. It then forwards the update of its spanning tree to its neighbors. In essence, WRP is one of the DVR protocols. It eliminates "counting to infinity" problem and temporary loops by utilizing information regarding the length and the second-to-last hop of the shortest path to each destination. The routing messages are far reaching because any node, which deletes a failed link from or adds a new link to its SST, must react.

1.3.11. Linked Cluster Architecture (LCA)

A lowest-ID distributed clustering algorithm is used to form an interconnected set of clusters covering the entire network. Cluster heads are used for channel access and power control adjustment. Each node has to broadcast its distance vector and update its routing table according to the distance vectors received from its neighbors. In essence, LCA uses flat DVR protocol for routing.

1.3.12. Virtual Subnets

Each node is a member of a physical subnet and a virtual subnet. Nodes within close proximity form a physical subnet. Nodes of the same virtual subnet form a regional subnet that spans the whole network. The address of a node consists of physical subnet ID and virtual subnet ID. Routing is based on this temporary address. This routing protocol assumes the fully connected network that is not common in mobile ad hoc networks.

1.4 Scalability in Ad-hoc Networks

As wirelessly networked devices become more pervasive, large-scale mobile ad hoc networks will become increasingly important for providing network services. In addition to basic communications, higher-level services must be provided in order to make these networks useful for a large community. When this goal is met, the necessity of deploying dedicated network infrastructure may no longer be required. Ad hoc networks are highly dynamic and decentralized in much the same way as p2p networks, this similarity may make p2p ideas well suited for application in this area.

Ad hoc wireless communication is a powerful technology allowing self organizing connectivity and network services with no preexisting infrastructure. Due to the fact that communication is not tied to any dedicated infrastructure, ad hoc networks are potentially more resilient and pervasive. This flexibility allows networks where there is no place to put wiring or the cost of installing infrastructure is prohibitive. In the case of a disaster where the existing infrastructure is damaged, ad hoc networking would allow communication that would otherwise not be possible. Additionally, ad hoc communication allows the network to grow with the number of people using it, not requiring new infrastructure to be built. Currently, mobile ad hoc networks (MANETs) allow the federation of a few hundred nodes to provide mutual connectivity. The only service provided by the network is basic connectivity; services such as NTP, DNS, and other high level applications are not available or could not be deployed in an efficient manner. The underlying network does not lend itself well to centralized applications since the broadcast medium would be quickly saturated around the server. While this lack of services may be acceptable for the small networks possible today, larger systems will need these applications to be useful. In order for larger ad hoc networks to be practical, scalability problems must be solved and self organizing distributed applications must be developed to replace those traditionally provided by infrastructure servers. There have been several solutions proposed to the problem of scalability, most notably those based on landmarks or coordinate systems. The coordinate system imposed by these networks is strikingly similar to those used in structured overlays. MANETs and structured p2p overlays share the properties of dynamic network membership and a decentralized nature with no administrative control. These similarities may help solve the problem of developing scalable ad hoc applications. Many decentralized applications have been built upon scalable p2p overlays and these same ideas may be useful in developing higher-level services in an ad hoc setting. This idea seems promising, but there are several key challenges that make the environment more difficult than the traditional Internet. The networks are highly dynamic with mobile nodes, and small cross sectional bandwidth introduces challenges not seen in a wired network. The goal of providing higher level service on top of large MANETS requires a number of concerns be addressed.

A number of ad hoc network routing protocols use a routing table to maintain a path from any source to any destination, called proactive routing. However, as the nodes are mobile, a lot of control overhead is generated to maintain these frequently broken routes. *Reactive routing* protocols for ad hoc networks are designed so that routing information is acquired only when needed. The reactive protocol discovers a new route to the destination on-demand, and tends to maintain it as long as the path is still being used for data traffic. This reduces control overhead to zero if the nodes are stationary. The above approach works well for small sized networks but for large networks routing without having any infrastructure is a big challenge. Routes might break as soon as they are discovered, wasting a lot of bandwidth without getting any data across. Scalable routing algorithms based on landmark hierarchies' landmark nodes self-organize themselves into a hierarchy, such that landmarks at a given level in the hierarchy are an approximately equal number of networks hops apart. Each node maintains routes to the nearest landmark node at each level. The address of a node consists of the sequence of ids of the nearest landmarks, from highest to lowest level. During routing, a node extracts from the destination address the highest-level landmark id that differs from its own node address, and forwards the packet towards the landmark with that id. The mapping from node identifiers to their current address is maintained in a distributed fashion. Landmark routing achieves scalability by dramatically reducing the size of per-node routing tables at the expense of somewhat longer routes. Since node addresses are maintained dynamically, landmark routing also lends it-self to mobile hosts. LANMAR is a variation landmark routing that targets mobile ad hoc networks with groups of nodes that are related in function and mobility, combining some of ideas from Landmark routing and from Fisheye State Routing to make MANETs' more scalable. L+ is an adaptation of landmark routing for mobile ad hoc networks.

1.5 Issues in scalability

The different issues regarding scalability of ZRP are as follows

1.5.1 Traffic Characteristics

In the Internet, there are high bandwidth long distance channels that connect physically far away parts of the network. In a MANET, all of the interfaces share the

same channel throughout the network. There is a very limited cross sectional bandwidth because there are no high capacity links. Traffic that is sent across the network will have to traverse the links between many nodes, thus using much of the network's capacity. Cooperative applications designed for the Internet, such as DHTs, must be reexamined before being transferred to a MANET as bandwidth optimization is now important. It would be ideal if all traffic crossed a very few links, but the predominant mode of communication in a large MANET is unknown and must be studied. In addition to studying the general traffic patterns in the network, it would also be useful to observe the traffic patterns of particular applications. Some applications may be more intrinsically suitable for deployment in a wireless network. Studying the traffic patterns of applications may also help determine ways applications could be better adapted to the wireless environment. Some application's data has natural physical locality and is only useful to a group of users who are geographically close to each other. Limiting the scope in which data can be retrieved to a smaller number of nodes than the whole system will naturally limit long distance traffic. Some applications have natural locality such as a community bulletin board system or class newsgroups. There should be a mechanism for inserting data so that limits the scope in which it is viewable, perhaps similar to. This could help conserve bandwidth in the system, but there should be an incentive for an application to do this.

While applications can be designed to minimize the amount of traffic that travels very far in the network, it can not always be avoided. To prevent such traffic from degrading performance of the entire network there should be mechanisms in place to either make such traffic less damaging, or to throttle it. If a metric is developed for how much a packet would cost to send in terms of bandwidth, perhaps a packet can be buffered in order to conserve bandwidth until the networks is idle and then sent. It also might be possible to combine packets headed in the same direction from several sources into one and then split them when their paths diverge so that all the space in one maximum transfer unit is utilized. Once basic network connectivity is established, it would be possible to run a traditional centralized service such as DNS on an ad hoc network. This would not be a scalable solution for several of the reason mentioned above, the broadcast channel around the nodes running these services could easily be saturated, and the distances to the root servers may be significant. To build a viable service on top of an ad hoc network it should balance the load equally

among the nodes and be aware of the wireless environment it is running in, something that traditional services do not do.

1.5.2 Fault Model

Structured overlays are often built in a manner that makes failures unlikely to affect the overall system. This is possible because in the wired internet failures are less likely than in a wireless environment and the network is fairly well connected. In the wireless environment the failures will be more drastic, thus aggravating many of the problems of correlated failure. In an ad hoc environment there may be one node holding two halves of a network together, and if that node fails the two halves of the network will each see half the nodes as having failed.

Additionally there are also many other factors that can cause wireless nodes to fail at a much higher rate. Many devices using wireless interfaces are powered by batteries, and will turn off to conserve batteries. Wireless devices are subject to disconnection due to the peculiarities of signal propagation, a device moving between two rooms may get varying degrees of connectivity. The loss rate of the wireless channel is also much higher than that of a wired ethernet channel resulting in the need to retransmit more packets.

A structured overlay designed on top of an unreliable system such as this must be implemented to handle these types of failures gracefully and not produce much additional overhead. The system must be resilient to failure as extreme as large network partitions that may make many nodes unreachable. For successful scalable network services to be built, these types of faults must not greatly degrade the services being provided.

1.5.3 Utilizing Existing Infrastructure

Having the technology to build large scale MANETs may eliminate the need for dedicated network infrastructure, but it is unknown whether the problems of low bandwidth and reliability can be overcome. One possible way of overcoming this obstacle is to utilize wired infrastructure when possible. In the case of a disaster some small islands of wired connectivity may exist and by attaching these to the ad hoc network, some critical services may be restored. Utilizing existing wired

infrastructure would allow islands of wireless connectivity to be connected forming a much larger network than would otherwise be possible. Inside a wireless island there maybe a wired connections that connects two distant points; it maybe more efficient for a packet to be routed through this connection towards its destination instead of the wireless channel.

The presence of wired infrastructure may also help eliminate several problems of reliability, as the wired channel is more reliable than the wireless. A network that has some underlying wired connectivity is less likely to suffer partitions as it has the redundant connection.

The best way to integrate these two very different environments needs to be studied. Simply using conventional bridging to attach the two types of networks leaves several questions unanswered about how to most efficiently use wired connection. For example it may be better to travel more hops if some of these hops are wired connection and then to take less wireless hops. Also the wired nodes are likely to not be mobile, there may be additional reliability assumptions that can be made about them, and there must be a mechanism to advertise their presence and capabilities. These issues must be addressed before existing infrastructure can be leveraged to provide a better level of service. While utilizing existing infrastructure may make some problems easier, simply wiring everything is not a viable option as it is expensive and should not be relied on as a solution.

1.5.4 Data Replication

In order to guarantee availability of data, it is often replicated over a configurable number nodes nodes in the system. The availability requirements of the data must be weighed against the bandwidth cost to store the data in nodes far away from the inserter. Storing data in a diverse set of nodes will decrease the likelihood that all will fail but it will increase the cost of inserting and retrieving the data. If the nodes close to the inserter are chosen the cost of inserting the data will decrease but in a correlated failure, such as a partition, it will be likely that data is unavailable to the network. This tradeoff must be studied carefully in order to provide highly available distributed data. A combination of caching, storing a small number of replicas far away and some close to the inserting node may yield satisfactory availability. It is

possible that in highly dynamic network such as this that scalable storage and high availability may not be possible. This issue must be studied in more depth before systems requiring availability guarantees can be implemented in this environment.

1.5.5 Id Assignment

In a traditional Internet environment ids are assigned either by generating a random id from a very large id space or assigned by a trusted authority. The possibility of collision is made very small by having the id space be very large like 160 bits. This may not work as well in the wireless environment. A node has a coordinate in the system, as well as a node id. A node's id may be its coordinate for each landmark level and this entire coordinate string uniquely identifies the node. Since the wireless channel is low bandwidth having many bit identifiers, which must be included in each packet, will add overhead to every packet sent. The length of each coordinate must be balanced for efficiency versus the possibility of collision.

Alternatively the node may have a separate node id and coordinate, if this is the case there has to be a distributed lookup mechanism to map node ids to coordinates. This lookup may result in a higher overhead for communication. If ids are generated by location and not by a secure mechanism, the system may be vulnerable to id based attacks. In particular a Sybil attack could be mounted by a hostile party simply moving a large number of mobile nodes into close proximity to a target node. This could allow the attacker to become an intermediate hop in all traffic being sent to the target. An attacker could also not forward control packets that contain information about the topology, causing the target to have an incorrect view of the network.

1.5.6 Id Change

A complicating factor in a mobile environment is that when a node moves its coordinate relative to other nodes will change. This can cause several problems, making this environment much more challenging. First, there is the problem of finding out where the node one wants to communicate is at any given time. This problem can be solved by a distributed look up service similar to the one in L+. This allows a node's locations to be determined by knowing an well known identifier for that node such as an IP address.

Secondly, since nodes can move arbitrarily coordinates that are looked up may become stale. In the event that this happens it may not be detected until no response is received from the destination node. This results in wasted traffic that consumes part of the limited bandwidth. To counteract this, coordinates can be updated in the DHT more often but this itself requires bandwidth. This tradeoff needs to be studied to find the sweet spot where most of the coordinates are right but they are not being updated too frequently. This has important consequences when caching coordinates, which will be very important in reducing traffic from lookups.

1.5.7 Preliminary Ideas

In this paper the issues that must be confronted to use p2p ideas to scale ad hoc networks have been enumerated. While each of the pose a specific problem and must each be individually addressed, there is still reason to be optimistic. A system that takes advantage of the broadcast nature of the wireless environment could use extensive caching to possibly overcome some of the bandwidth concerns. Information about the network could be exposed to the application level so that decisions about how to most effectively use the limited bandwidth resource can be made. In addition to leveraging existing wired infrastructure, these techniques will bring us closer to practical, usable, large-scale ad hoc networks.

1.6 Objective

The main objective of this thesis is to provide a scalable zone routing protocol which is the extension of a mobile ad hoc network. The scalability of ad-hoc networks as discussed earlier depends upon the traffic load, power optimization and other factors. As ZRP is a hybrid approach which is the combination of both reactive and proactive routing approaches. Our goal is to provide a methodology that ZRP can be implemented in a more efficient way where the node mobility and traffic load can be handled for the network. The next purpose is to provide an efficient ZRP where unidirectional links are present, for that we have proposed a query enhancement mechanism that would provide a more scalable ZRP. The simulation-based results on ns-2.26 would represent the scalability of ZRP as compared to the other routing protocols of ad-hoc networks.

Chapter 2

Survey of Literature

2. Literature Survey

2.1 Zone Routing Protocol

As seen, proactive routing uses excess bandwidth to maintain routing information, while reactive routing involves long route request delays. Reactive routing also inefficiently floods the entire network for route determination. The Zone Routing Protocol (ZRP) aims to address the problems by combining the best properties of both approaches. ZRP can be classed as a hybrid reactive/proactive routing protocol. In an ad-hoc network, it can be assumed that the largest part of the traffic is directed to nearby nodes. Therefore, ZRP reduces the proactive scope to a zone centered on each node. In a limited zone, the maintenance of routing information is easier. Further, the amount of routing information that is never used is minimized. Still, nodes farther away can be reached with reactive routing. Since all nodes proactively store local routing information, route requests can be more efficiently performed without querying all the network nodes. Despite the use of zones, ZRP has a flat view over the network. In this way, the organizational overhead related to hierarchical protocols can be avoided. Hierarchical routing protocols depend on the strategic assignment of gateways or landmarks, so that every node can access all levels, especially the top level. Nodes belonging to different subnets must send their communication to a subnet that is common to both nodes. This may congest parts of the network. ZRP can be categorized as a flat protocol because the zones overlap. Hence, optimal routes can be detected and network congestion can be reduced. Further, the behavior of ZRP is adaptive. The behavior depends on the current configuration of the network and the behavior of the users.

2.2 Architecture

The Zone Routing Protocol, as its name implies, is based on the concept of zones. A routing zone is defined for each node separately, and the zones of neighboring nodes overlap. The routing zone has a radius expressed in hops. The zone thus includes the nodes, whose distance from the node in question is at most hops. An example routing zone is shown in Figure 1, where the routing zone of S

includes the nodes A–I, but not K. In the illustrations, the radius is marked as a circle around the node in question. It should however be noted that the zone is defined in hops, not as a physical distance.

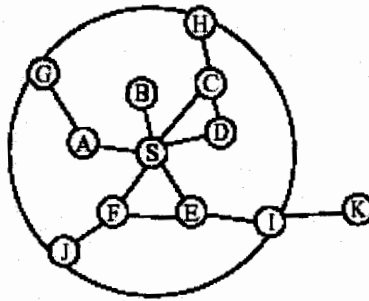


Fig 2.1 Routing Zone With $p=2$

The nodes of a zone are divided into peripheral nodes and interior nodes. Peripheral nodes are nodes whose minimum distance to the central node is exactly equal to the zone radius p . The nodes whose minimum distance is less than p are interior nodes. In Figure 2.1, the nodes A–F are interior nodes, the nodes G–J are peripheral nodes and the node K is outside the routing zone. Note that node H can be reached by two paths, one with length 2 and one with length 3 hops. The node is however within the zone, since the shortest path is less than or equal to the zone radius.

The number of nodes in the routing zone can be regulated by adjusting the transmission power of the nodes. Lowering the power reduces the number of nodes within direct reach and vice versa. The number of neighboring nodes should be sufficient to provide adequate reachability and redundancy. On the other hand, a too large coverage results in many zone members and the update traffic becomes excessive. Further, large transmission coverage adds to the probability of local contention.

ZRP refers to the locally proactive routing component as the Intra-zone Routing Protocol (IARP). The globally reactive routing component is named Inter-zone Routing Protocol (IERP). IERP and IARP are not specific routing protocols. Instead, IARP is a family of limited-depth, proactive link-state routing protocols. IARP maintains routing information for nodes that are within the routing zone of the node correspondingly, IERP is a family of reactive routing protocols that offer enhanced route discovery and route maintenance services based on local connectivity monitored by IARP.

The fact that the topology of the local zone of each node is known can be used to reduce traffic when global route discovery is needed. Instead of broadcasting packets, ZRP uses a concept called *bordercasting*. Bordercasting utilizes the topology information provided by IARP to direct query request to the border of the zone. The Bordercast Resolution Protocol (BRP) provides the bordercast packet delivery service. BRP uses a map of an extended routing zone to construct bordercast trees for the query packets. Alternatively, it uses source routing based on the normal routing zone. By employing query control mechanisms, route requests can be directed away from areas of the network that already have been covered.

In order to detect new neighbor nodes and link failures, the ZRP relies on a Neighbor Discovery Protocol (NDP) provided by the Media Access Control (MAC) layer. NDP transmits "HELLO" beacons at regular intervals. Upon receiving a beacon, the neighbor table is updated. Neighbors, for which no beacon has been received within a specified time, are removed from the table. If the MAC layer does not include a NDP, the functionality must be provided by IARP.

The relationship between the components is illustrated in Figure 2. Route updates are triggered by NDP, which notifies IARP when the neighbor table is updated. IERP uses the routing table of IARP to respond to route queries. IERP forwards queries with BRP. BRP uses the routing table of IARP to guide route queries away from the query source.

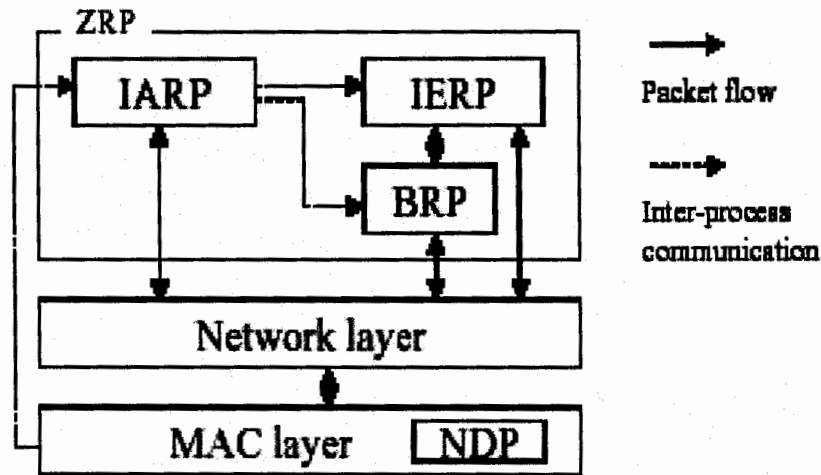


Fig 2.2 ZRP Architecture

2.3 Routing

A node that has a packet to send first checks whether the destination is within its local zone using information provided by IARP. In that case, the packet can be routed proactively. Reactive routing is used if the destination is outside the zone.

The reactive routing process is divided into two phases: the *route request* phase and the *route reply* phase. In the route request, the source sends a route request packet to its peripheral nodes using BRP. If the receiver of a route request packet knows the destination, it responds by sending a route reply back to the source. Otherwise, it continues the process by border casting the packet. In this way, the route request spreads throughout the network. If a node receives several copies of the same route request, these are considered as redundant and are discarded. The reply is sent by any node that can provide a route to the destination. To be able to send the reply back to the source node, routing information must be accumulated when the request is sent through the network. The information is recorded either in the route request packet, or as next-hop addresses in the nodes along the path. In the first case, the nodes forwarding a route request packet append their address and relevant node/link metrics to the packet. When the packet reaches the destination, the sequence of

addresses is reversed and copied to the route reply packet. The sequence is used to forward the reply back to the source. In the second case, the forwarding nodes records routing information as next-hop addresses, which are used when the reply is sent to the source. This approach can save transmission resources, as the request and reply packets are smaller. The source can receive the complete source route to the destination. Alternatively, the nodes along the path to the destination record the next-hop address in their routing table.

In the bordercasting process, the bordercasting node sends a route request packet to each of its peripheral nodes. This type of one-to-many transmission can be implemented as multicast to reduce resource usage. One approach is to let the source compute the multicast tree and attach routing instructions to the packet. This is called Root-Directed Bordercasting (RDB). Another approach is to reconstruct the tree at each node, whereas the routing instructions can be omitted. This requires that every interior node knows the topology seen by the bordercasting node. Thus, the nodes must maintain an extended routing zone with radius 2 hops. Note that in this case the peripheral nodes where the request is sent are still at the distance p . This approach is named Distributed Bordercasting (DB). The zone radius is an important property for the performance of ZRP. If a zone radius of one hop is used, routing is purely reactive and bordercasting degenerates into flood searching. If the radius approaches infinity, routing is reactive. The selection of radius is a trade off between the routing efficiency of proactive routing and the increasing traffic for maintaining the view of the zone.

2.4 Route maintenance

Route maintenance is especially important in ad-hoc networks, where links are broken and established as nodes move relatively to each other with limited radio coverage. In purely reactive routing protocols, routes containing broken links fail and a new route discovery or route repair must be performed. Until the new route is available, packets are dropped or delayed. In ZRP, the knowledge of the local topology can be used for route maintenance. Link failures and sub-optimal route segments within one zone can be bypassed.

Incoming packets can be directed around the broken link through an active multi-hop path. Similarly, the topology can be used to shorten routes, for example, when two nodes have moved within each other's radio coverage. For source-routed packets, a relaying node can determine the closest route to the destination that is also a neighbor. Sometimes, a multi-hop segment can be replaced by a single hop. If next-hop forwarding is used, the nodes can make locally optimal decisions by selecting a shorter path.

2.4.1 Example

Consider the network in Figure 3. The node S has a packet to send to node X. The zone radius is $p=2$. The node uses the routing table provided by IARP to check whether the destination is within its zone. Since it is not found, a route request is issued using IERP. The request is bordercast to the peripheral nodes (gray in the picture). Each of these searches their routing table for the destination.

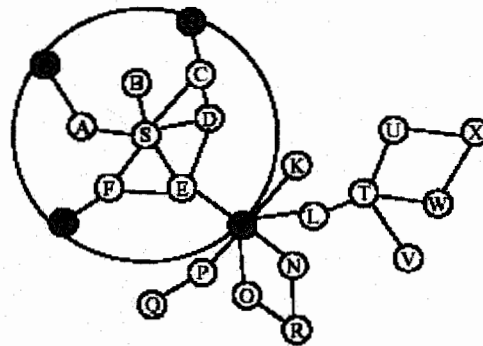


Fig 2.3 The routing zone of node S

Node I does not find the destination in its routing table. Consequently, it broadcasts the request to its peripheral nodes, shown in gray in Figure 4. Due to query control mechanisms, the request is not passed back to nodes D, F and S.

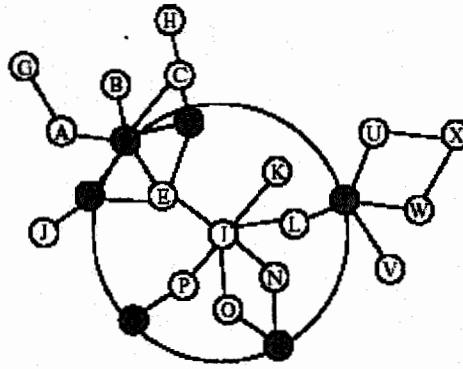


Fig 2.4 The routing zone of node I

Finally, the route request is received by node T, which can find the destination in its routing zone, shown in Figure 2.4. Node T appends the path from itself to node X to the path in the route request. A route reply, containing the reversed path is generated and sent back to the source node. If multiple paths to the destination were available, the source would receive several replies.

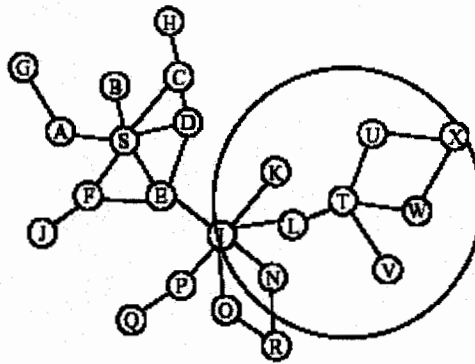


Fig 2.5 Routing Zone of Node T

2.5 Query-control mechanisms

Bordercasting can be more efficient than flooding, since route request packets are only sent to the peripheral nodes, and thus only on the corresponding links. Further efficiency can be gained by utilizing multicast techniques. In that case, only one packet is sent on a link although several peripheral nodes can reside behind this link. However, since the routing zones of neighboring nodes overlap, each node may

forward route requests several times, which results in more traffic than in flooding. When a node bordercasts a query, the complete routing zone is effectively covered. Any further query messages entering the zone are redundant and result in wasted transmission capacity. The excess traffic is a result from queries returning to covered zones instead of covered nodes as in traditional flooding.

To solve this problem, ZRP needs query-control mechanisms, which can direct queries away from covered zones and terminate query packets before they are delivered to peripheral nodes in regions of the network already covered by the query. ZRP uses three types of query-control mechanisms: query detection, early termination and random query-processing delay. Query detection caches the queries relayed by the nodes. With early termination, this information is used to prune bordercasting to nodes already covered by the query.

2.6 Query detection

When a bordercast is issued, only the bordercasting node is aware that the routing zone is covered by the query. When the peripheral nodes continue the query process by bordercasting to their peripheral nodes, the query may be relayed through the same nodes again. To illustrate with an example, the node S in Figure 6 bordercasts a query to its peripheral nodes F–J. As the node J continues by bordercasting to the nodes C, S and E, the query is again relayed by nodes D and E. The query issued by node J to nodes C, S and E is redundant, since these nodes have been covered by the previous query.

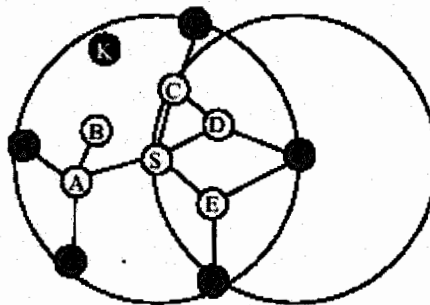


Figure 2.6: Query detection example

To be able to prevent queries from reappearing in covered regions, the nodes must detect local query relaying activity. BRP provides two query detection methods: QD1 and QD2. Firstly, the nodes that relay the query are able to detect the query (QD1). Secondly, in single-channel networks, it is possible to listen to the traffic by other nodes within the radio coverage (QD2). Hence, it is possible to detect queries relayed by other nodes in the zone. QD2 can be implemented by using IP broadcasts to send route queries. Alternatively, unicast can be used if the MAC and IP layers operate in promiscuous mode. In the above example, all nodes except node B relay the query of S. They are thus able to use QD1. Node B does not belong to the bordercast tree, but it is able to overhear the relayed query using QD2. However, node K does not overhear the message, and is therefore unaware that the zone of node S is covered. A query detection table is used to cache the detected queries. For each entry, the cache contains the address of the source node and the query ID. The address-ID pair is sufficient to uniquely identify all queries in the network. The cache may also contain other information depending on the query detection scheme. Especially the address of the node that most recently bordercasted a query is important.

2.7 Early termination

With Early Termination (ET), a node can prevent a route request from entering already covered regions. Early termination combines information obtained through query detection with the knowledge of the local topology to prune branches leading to peripheral nodes inside covered regions. These regions consist of the interior nodes of nodes that already have bordercast the query. A node can also prune a peripheral node if it has already relayed a query to that node. Early termination requires topology information extending outside the routing zone of the node. The information is required to reconstruct the bordercast tree of other nodes within the routing zone. The extended routing zone has a radius of $2p-1$. Alternatively, in the case of root-directed bordercast (RDB), the topology of the standard routing zone and information about cached bordercast trees can be used. In the previous example, node E can use the information in its query detection table to prune the query that the node J sends to its peripheral node F. Node E has an extended routing zone with radius $2p-1=3$, shown as a dashed circle in Figure 7.

T-1205

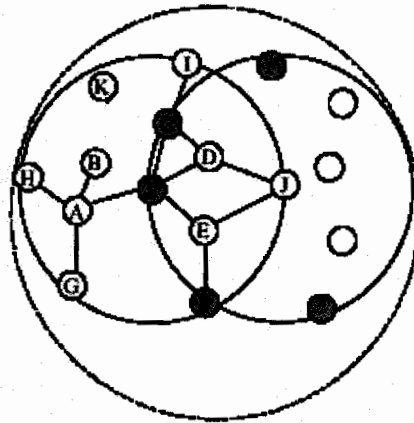


Fig 2.8 The Extended route of zone E

2.8 Random query-processing delay

When a node issues a node request, it takes some time for the query to be relayed along the bordercast tree and to be detected through the query detection mechanisms. During this time, another node may propagate the same request. This can be a problem when several nearby nodes receive and re-broadcast a request at roughly the same time. To reduce the probability of receiving the same request from several nodes, a Random Query-Processing Delay (RQPD) can be employed. Each bordercasting node waits a random time before the construction of the bordercast tree and the early termination. During this time, the waiting node can detect queries from other bordercasting nodes and prune the bordercast tree. To avoid additional route discovery delay, the delay can be combined with the pre-transmission jitter used by many route discovery protocols. Assume that in Figure 7 the nodes C and S both receive a query. Node C schedules a bordercast to its peripheral node E, and node S to its peripheral node F. Without RQPD, both nodes would issue the broadcast simultaneously, and thereafter detect the message of the neighbor node. With RQPD, the node C may detect the query sent by node S during the delay, and prune the branch leading to E.

2.9 Caching

This thesis further proposes caching as a technique for reducing control traffic. The nodes cache active routes, and by using this cache, the frequency of route

discovery procedures can be reduced. Changes in network topology, such as broken links, are compensated by local path repair procedures. A new path then substitutes the path between the ends of the broken link and a path update message is sent to the endpoints of the path. Since the repair reduces the efficiency of the routes, the endpoints may initiate a new route discovery procedure after a number of repairs.

Chapter 3

System Analysis and Design

3. Zone Routing Protocol

ZRP is a hybrid approach and consists of

1. IARP
2. IERP
3. BRP
4. Query Control Mechanism

3.1 Introduction to ZRP

As explained above, both a purely pro-active or purely reactive approach to implement a routing protocol for a MANET have their disadvantages. The *Zone Routing Protocol*, or ZRP, as described in this document combines the advantages of both into a *hybrid* scheme, taking advantage of pro-active discovery within a node's local neighborhood, and using a reactive protocol for communication between these neighborhoods.

In a MANET, it can safely be assumed that the most communication takes place between nodes close to each other. Changes in the topology are most important in the vicinity of a node - the addition or the removal of a node on the other side of the network has only limited impact on the local neighborhoods.

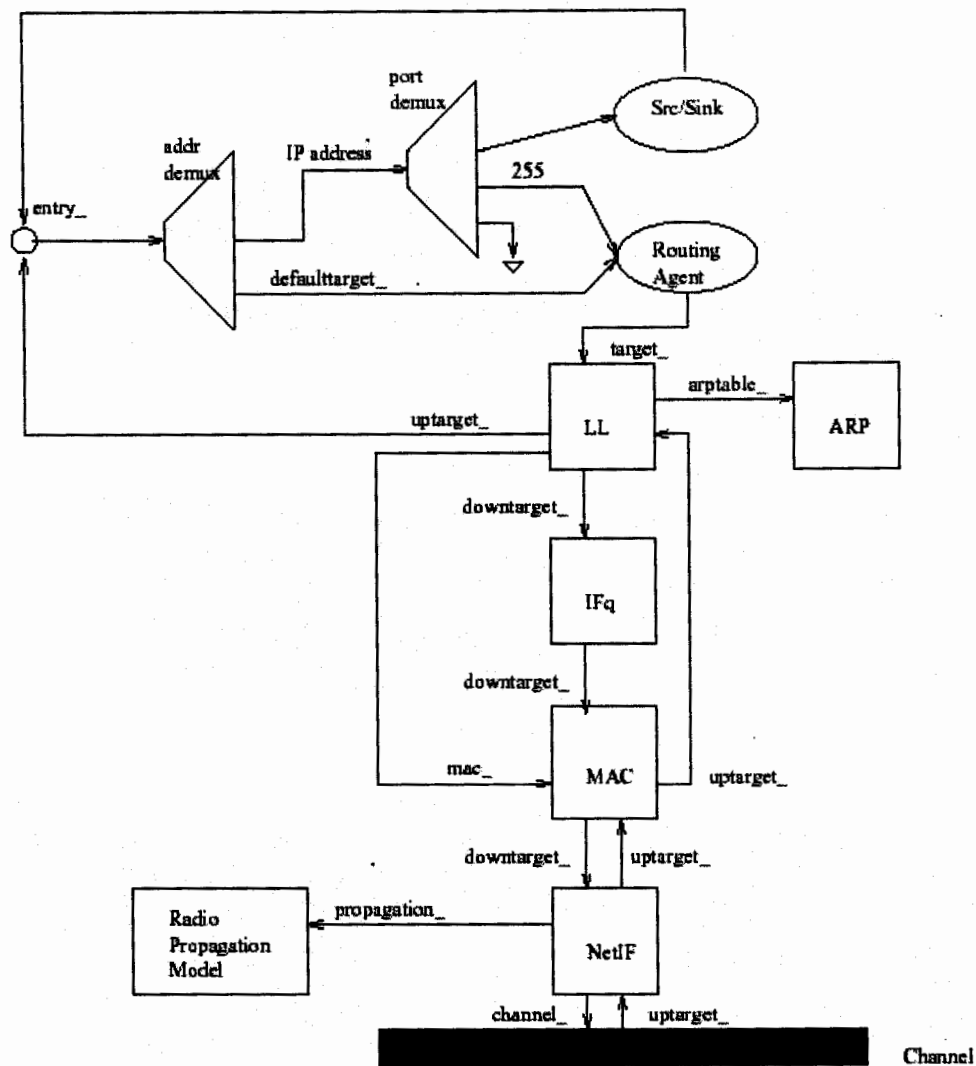


Fig 3.1 Architecture Of ZRP

As mentioned earlier, the ZRP is not so much a distinct *protocol* as it provides a framework for other protocols. The separation of a nodes local neighborhood from the global topology of the entire network allows for applying different approaches - and thus taking advantage of each technique's features for a given situation. These local neighborhoods are called *zones* (hence the name); each node may be within multiple overlapping zones, and each zone may be of a different size. The "size" of a zone is not determined by geographical measurement, as one might expect, but is given by a radius of length ρ , where ρ is the number of hops to the perimeter of the zone.

By dividing the network into overlapping, variable-size zones, ZRP avoids a hierarchical map of the network and the overhead involved in maintaining this map. Instead, the network may be regarded as flat, and route optimization is possible if overlapping zones are detected.

While the idea of zones often seems to imply similarities with cellular phone services, it is important to point out that each node has its own zone, and does not rely on fixed nodes (which would be impossible in MANETs). Figure 1 shows an example routing zone with $\rho=2$.

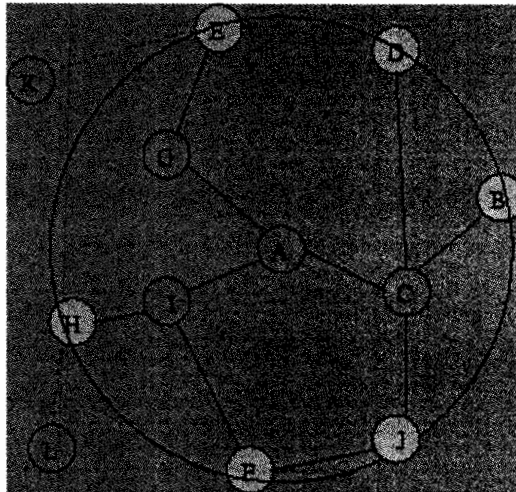


Figure 3.2: Routing Zone of node A with $\rho=2$.

Note that in this example node A has multiple routes to node F, including one that has a hop count of $c > \rho$. Since it also has a route with $c \leq \rho$, F still belongs to A's zone. Node G is out of A's zone,

The nodes on the perimeter of the zone (i.e. with a hopcount $hc = \rho$) are referred to as peripheral nodes (marked gray), nodes with $hc < \rho$ are interior nodes.

Obviously a node needs to first know about its neighbors before it can construct a routing zone and determine its peripheral nodes. In order to learn about its direct

neighbors, a node may use the media access control (MAC) protocols directly. Alternatively, it may require a *Neighbor Discovery Protocol* (NDP). Again, we see that ZRP, as a framework, does not strictly specify the protocol used but allows for local independent implementations.

Such a Neighbor Discovery Protocol typically relies on the transmission of "hello" beacons by each node. If a node receives a response to such a message, it may note that it has a direct point-to-point connection with this neighbor. The NDP is free to select nodes on various criteria, such as signal strength or frequency/delay of beacons etc. Once the local routing information has been collected, the node periodically broadcasts discovery messages in order to keep its map of neighbors up to date. In doing so, it is assumed that these "link-layer (neighbor) unicasts are delivered reliably and in-sequence."

If the MAC layer of the nodes does not allow for such a NDP, the Intrazone Routing Protocol must provide the possibility of direct neighbor discovery. This protocol is responsible for determining the routes to the peripheral nodes and is commonly a proactive protocol. The Intrazone Routing Protocol, or IARP, Communication between the different zones is guarded by the Interzone Routing Protocol, or IERP, and provides routing capabilities among peripheral nodes only. That is, if a node encounters a packet with a destination outside its own zone - i.e. it does not have a valid route for this packet - it forwards it to its peripheral nodes, which maintain routing information for the neighboring zones, so that they can make a decision of where to forward the packet to. Through the use of a bordercast algorithm rather than flooding all peripheral nodes, these queries become more efficient.

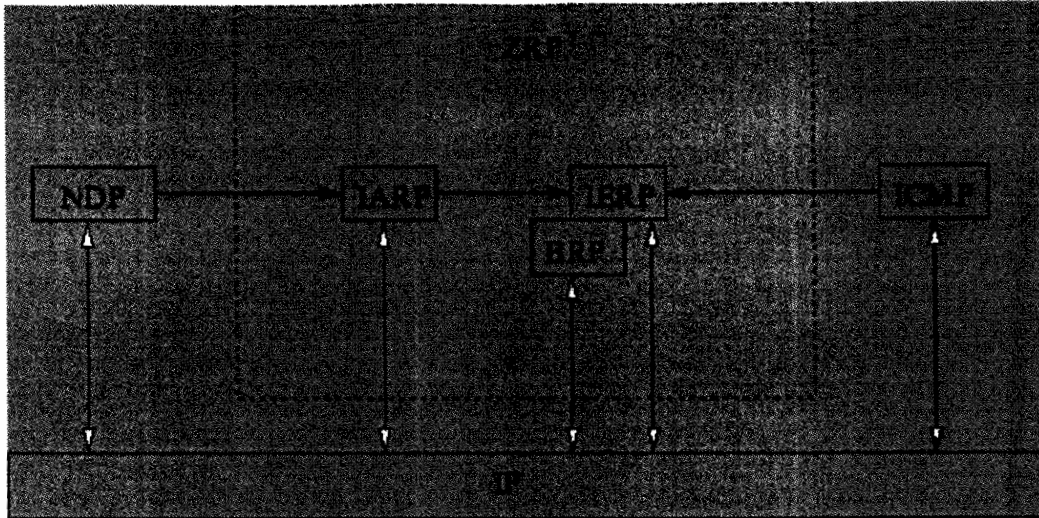


Figure 3.3: ZRP components

As we can see, the Zone Routing Protocol consists of several components, which only together provide the full routing benefit to ZRP. Each component works independently of the other and they may use different technologies in order to maximize efficiency in their particular area. For example, a reactive protocol such as AODV might be used as the IARP, while the IERP is most commonly a pro-active protocol such as OLSR.

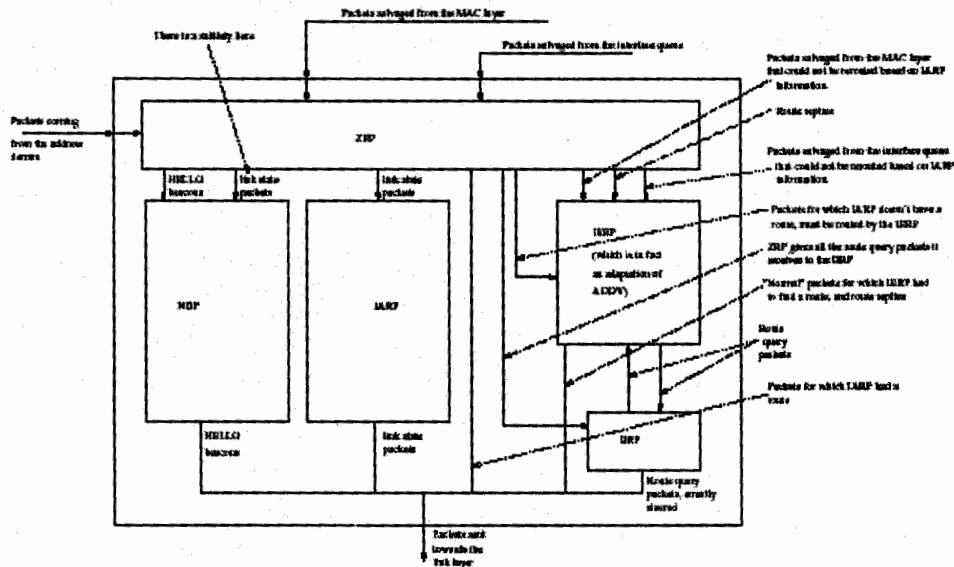


Fig 3.4 Block Diagram Of ZRP

Even though the hybrid nature of the ZRP seems to indicate that it is a hierarchical protocol, it is important to point out that the ZRP is in fact a *flat* protocol. In a hierarchical network architecture, two different protocols are maintained for communication among (a) each individual cluster's nodes and (b) the different clusters. The main difference here is that in the ZRP there is a one-to-one correspondence between nodes and routing zones, causing overlapping zones maintained by each individual nodes

3.2 Intrazone Routing Protocol (IARP)

Since ZRP assumes that local neighbor discovery is implemented on the link-layer and is provided by the NDP, the first protocol to be part of ZRP is the *Intrazone Routing Protocol*, or IARP. This protocol is used by a node to communicate with the

interior nodes of it's zone and as such is limited by the zones radius ρ (the number of hops from the node to it's peripheral nodes). Since the local neighborhood of a node may rapidly be changing, and since changes in the local topology are likely to have a bigger impact on a nodes routing behavior than a change on the other end of the network, the IARP is a pro-active, table-driven protocol.

The node continuously needs to update the routing information in order to determine the peripheral nodes as well as maintain a map of which nodes can be reached locally. The IARP allows for local route optimization through the removal of redundant routes and the shortening of routes if a route with fewer hops has been detected, as well as bypassing link-failures through multiple (local) hops, thus leveraging global propagation.

As mentioned earlier, it is possible that a node A can broadcast messages to a node B, but that node B, due to limitations in it's signal-strength (caused by interference, for example) or low transmission power, can not reach node A. Therefore, it is important for the IARP to provide support for unidirectional links among the local nodes. Due to it's pro-active nature, local route discovery is very efficient and routes to local destinations are immediately available. In order to not

over utilize the available bandwidth resources, the IARP - as the name suggests - is restricted to routing within the zone, which is why it is referred to as a "limited scope pro-active routing protocol.

Global route discovery, communication with nodes in a different zone, is done by guiding the route queries to the peripheral nodes instead of flooding all local nodes. In order to adopt a traditional pro-active link state protocol for use as the IARP in the ZRP, the scope of the protocol needs to be limited to the size of the zone ρ . This may be implemented by adding a Time To Live (TTL) to the route discovery requests, initialized to $\rho - 1$, and decremented by each node until it reaches 0 (when it is discarded).

In Figure uninterrupted lines indicate the areas where the IARP is used to provide routing between the nodes.

3.3 Interzone Routing Protocol (IERP)

As the global reactive routing component of the ZRP, the Interzone Routing Protocol, or IERP, takes advantage of the known local topology of a node's zone and, using a reactive approach enables communication with nodes in other zones.

Route queries within the IERP are issued on demand, that is only when a request for a route is made. The delay caused by the route discovery (in contrast to IARP, where the route is immediately available) is minimized through the use of bordercasting, an approach in which the node does not submit the query to all local nodes, but only to its peripheral nodes. Furthermore, a node does *not* send a query back to the nodes the request came from, even if they are peripheral nodes. In order to convert an existing reactive routing protocol for use as the IERP in the ZRP, it is necessary to disable pro-active updates for local routes, since this functionality is provided by the IARP. Furthermore, the IERP needs to be able to take advantage of the local routing information provided by the IARP, as well as change the way route discovery is handled: Instead of flooding a route request to all nodes, it should instead use the Bordercast Resolution Protocol (BRP) to only initiate route requests with peripheral nodes.

In Figure, dotted lines indicate the areas where the IERP is used to provide routing between the zones. The Interzone Routing Protocol including an example implementation (Reactive Source Routing) is described in more detail in.

3.4 Bordercast Resolution Protocol (BRP)

The Bordercast Resolution Protocol, or BRP, is used in the ZRP to direct the route requests initiated by the global reactive IERP to the peripheral nodes, thus removing redundant queries and maximizing efficiency. In doing so, it utilizes the map provided by the local pro-active IARP to construct a bordercast tree. Unlike IARP and IERP, it is not so much a routing protocol, as it is packet delivery service.

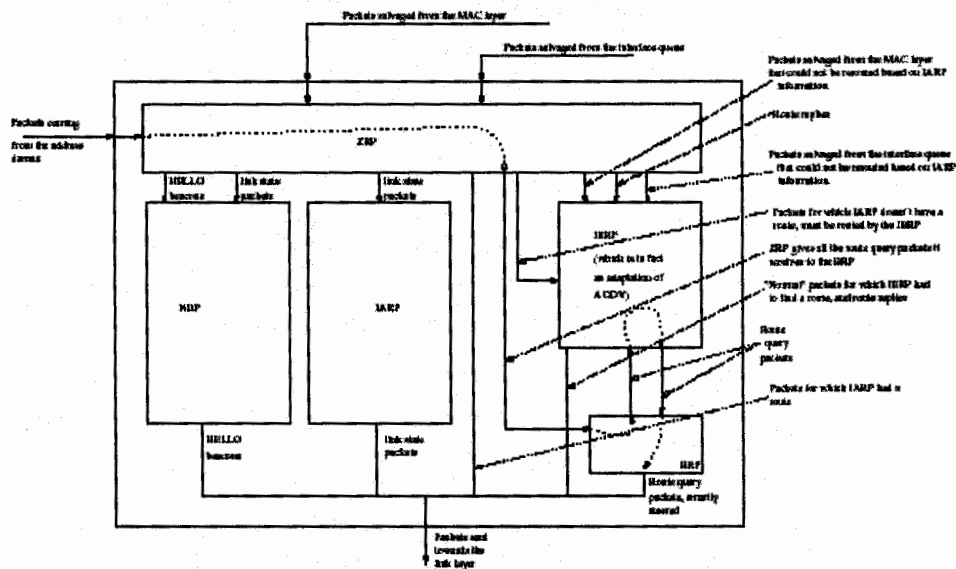


Fig 3.5 Bordercast Routing Protocol

The BRP keeps track of which nodes a query has been delivered to, so that it can prune the bordercast tree of nodes that have already received (and relayed) the query. When a node receives a query packet for a node that does not lie within its local routing zone, it constructs a bordercast tree so that it can forward the packet to its neighbors. These nodes, upon receiving the packet, reconstruct the bordercast tree so that they can determine whether or not it belongs to the tree of the sending node. If it does not, it continues to process the request and determines if the destination lies

within its routing zone and taking the appropriate action, upon which the nodes within this zone are marked as covered.

In order to detect when a routing zone they belong to has been queried, two levels of *Query Detection* are provided by BPR. As they relay the queries to the peripheral nodes, the nodes detect the query and notes which zones have been covered. This is referred to as the first level of Query Detection, or QD1.

Secondly, in networks that use a single broadcast channel, a node can determine this information by listening to the traffic broadcast among other nodes. This approach is referred to as QD2. Figure 3 shows node A bordercasting a query to the peripheral nodes D and F. Nodes B and C, as they relay the query, note that node A's zone has been queried (QD1). In single-channel networks, node E can listen to the traffic and come to the same conclusion using QD2.

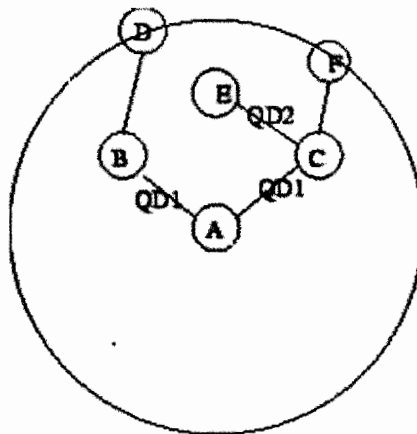


Figure 3.6: QD1 and QD2.

Simply detecting that a given node has already been covered, however, is not enough. The protocol needs to drop packets that would be sent to already covered nodes. This is done using *Early Termination* or ET, which obviously relies on Query Detection, as well as *Loopback Termination* or TL (in which routes that loop back into the querying nodes zone are eliminated, since these nodes can be reached locally using IARP).

In order to further eliminate unnecessary broadcasting, the BRP may implement *Selective Bordercasting*. In this approach, a node needs to know network topology information for an extended zone of size $2^p - 1$. Given this knowledge, a node can further eliminate peripheral nodes from its list of bordercast recipients, if the outer peripheral nodes overlap. Figure 4 shows an example of how node A is able to remove node C from its bordercasting spanning tree, since nodes G and H can be reached through nodes B and D respectively.

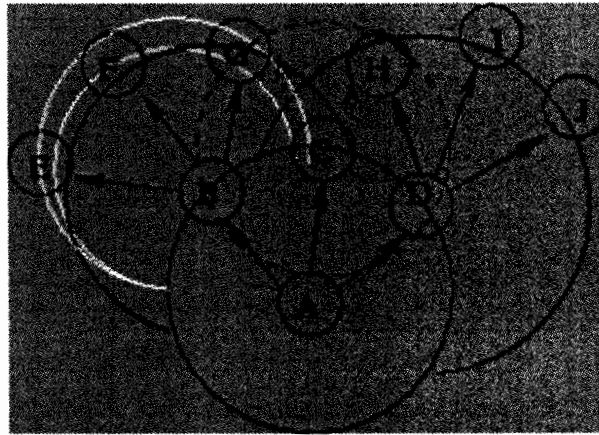


Figure 3.7: Selective Broadcasting.

In the context of ZRP, the BRP can be seen as the glue which ties together the IARP and the IERP in order to take full advantage of the pro-active and reactive components where they are best used.

The Bordercast Resolution Protocol, including the implementation is described in more detail in.

3.5 Examples

In order to better understand how the different components of the ZRP work together and how routing is done using this approach, we will consider a few examples. Several scenarios are possible, among them stationary nodes in a dense network, mobile nodes moving in different directions with and without stable and/or stationary

fixpoints and stationary or moving networks with instable or frequently changing nodes. In these examples, a node is considered *stationary*, if it does not move relative to the other nodes (even though it may be moving into the same direction). A *stable* node is one that broadcasts with a constant signal and does not undergo power fluctuations.

3.6 BRP

Before we go into examples of how routes between nodes from different zones are determined, let us consider an example of how BRP performs Query Detection.

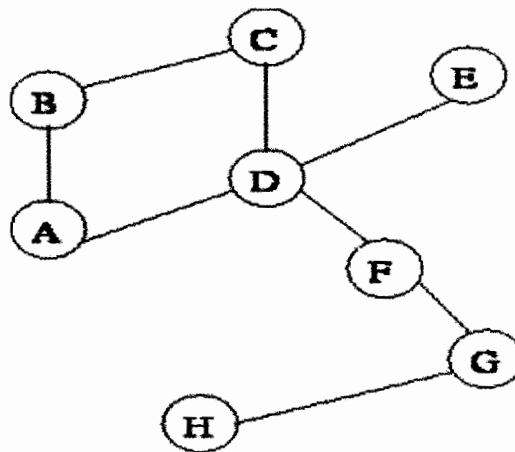


Figure 3.8: A BRP Example.

In Figure, node A needs to send a packet to node H. Since node H is not within its zone, it constructs a bordercast tree spanning its peripheral nodes C, E and F, and sends the query to its local neighbors B and D. Both B and D note that H is not within their zone they in turn construct a bordercast tree. Note that they only include peripheral nodes of theirs that have not been covered previously.

That is, node B will not include node D, even though it is a peripheral node for B's zone. Similarly, node D's tree will not include node B. Node D then forwards the request to node F, which determines that node H lies within its routing zone, and replies with the correct route.

3.7 Stationary nodes in a dense network

Let us extend the network from Figure 3.1 to span a few more nodes. An example situation of such a stationary network, where the number of nodes and their position does not change frequently might be a number of people attending a conference, communicating with each other. This example lends itself to show the basics of ZRP, so we will investigate it in more detail.

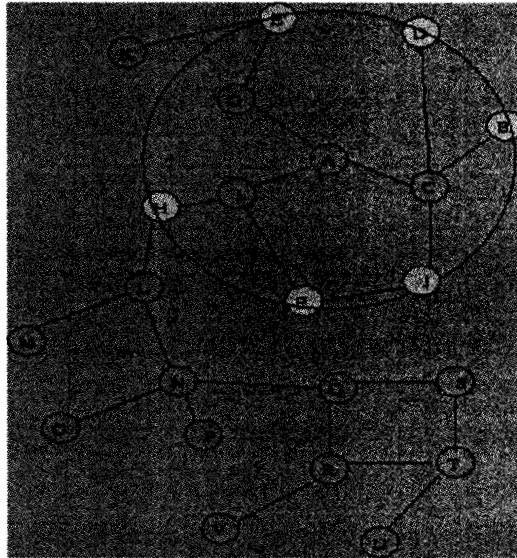


Figure 3.9: A stationary network with $\rho = 2$.

Figure shows the network graph; the radius of each node's zone is still $\rho = 2$. Again, as before, the peripheral nodes for node A are marked gray. In this example, node A has to send a packet to node U.

Node A uses the IARP to determine if node U is within its zone. Since IARP is proactive, the information that U is in this routing zone is readily available, and node A initiates a route request using IERP. IERP now utilizes BRP to bordercast the request: It is not flooded to each of the nodes in A's zone, but only to B, D, E, F, H and J, the peripheral nodes, which in turn search their routing tables for the destination.

Node H does not find U in its routing table and thus bordercasts the request to its peripheral nodes. Figure 3.7 shows node H's routing zone.

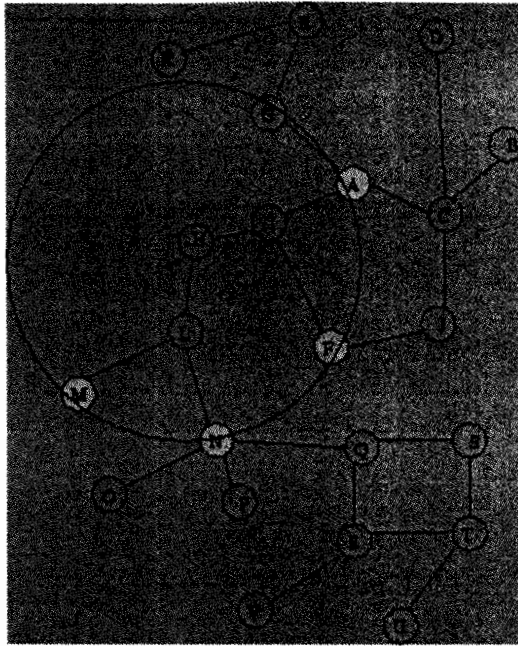


Figure 3.10: Node H's zone

Through the use of BPR, the bordercast tree of node H does not contain nodes F or A: these branches have been pruned, since these nodes have already been covered. Node H broadcasts the route query to nodes M and N. Some of the peripheral nodes of node N's zone are also peripheral nodes of node U's zone - an example of how zones can (and frequently do) overlap. In Figure, the peripheral nodes for both node U and node N are shaded.

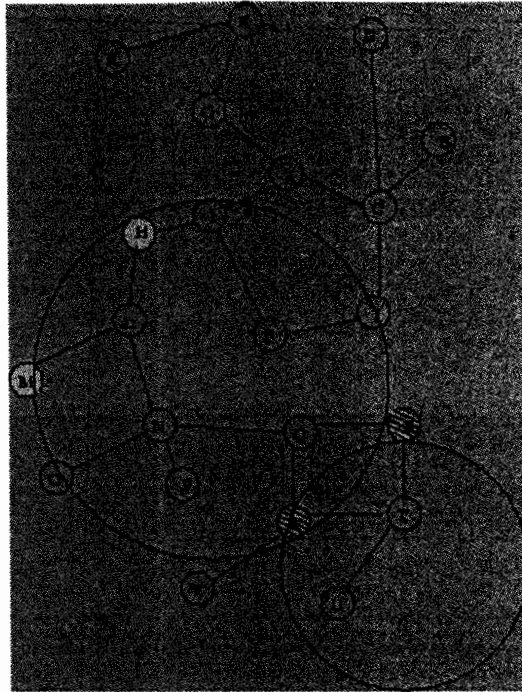


Figure 3.11: Overlapping zones

Now node N bordercasts the query to nodes R and S (nodes M and H are already marked as covered), which in turn reply with the correct route, as they both know U to be within their local zone.

3.8 Mobile nodes without a stationary fixpoint

In this example, we will see how ZRP deals with link-failures and link-optimization, since moving nodes will constantly have to update their network map, as nodes that used to be within their zone move out of transmission range. For simplicity, let us consider a much smaller initial network. In Figure 9, the arrows indicate the direction in which the nodes are traveling. As we observe node A, we mark its peripheral nodes gray.

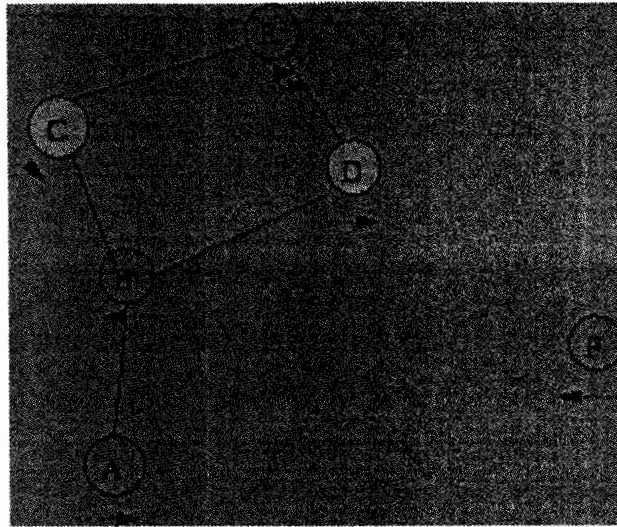


Figure 3.12: Mobile nodes (1).

As nodes D and B move further away from each other, they lose their connection, just as nodes C and E. However, node E moves closer and is able to establish a point-to-point connection with nodes A and D. Similarly, node F moves closer and is able to establish connections with nodes E and A. These changes are reflected in Figure 10.

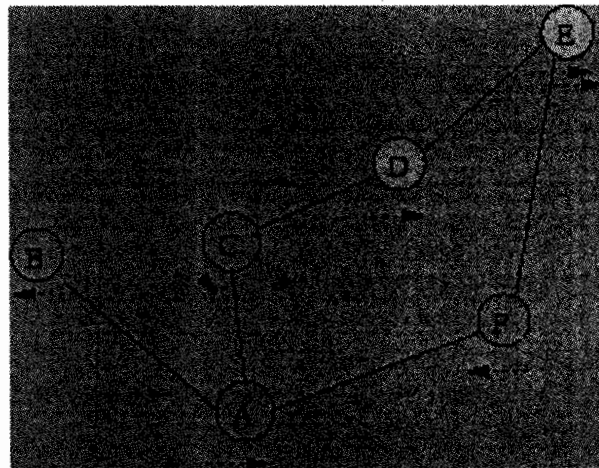


Figure 3.11: Mobile nodes (2).

In this example, it is interesting to point out that node D remains a peripheral node for A's zone, even though the local route from A to D changes. This change shows the need for a pro-active IERP: the changes in the networks topology like will have taken place within a short period of time. Nodes connected to E only need to add an

additional peripheral node (F), as E maintains its connection to node D. However, were a node connected to E request a route to C, the IERP would need to initiate a new route request, as C no longer is within the querying node's zone and the route needs to go through D.

3.9 Performance

In order to maximize performance of the ZRP, we need to minimize the amount of control traffic that is sent. Thus, we wish to maintain an overview of the networks topology that is as accurate as possible (at any given time - thus minimizing delays caused by route discovery requests), while at the same time requiring sending as little packages as possible.

Given the hybrid nature of the ZRP, this goal can be reduced to finding the correct -

i.e. optimal - size of the routing zone radius ρ for the given network - which may vary from case to case, depending on the circumstances.

For example, in a stationary network as considered in Section 4.2, it would be

possible to increase ρ to a larger number, without too much of a penalty: in this situation, the position or the number of available nodes changes infrequently, so that, given a larger routing zone radius, the nodes could take advantage of the comparably static and immediately available, since pro-actively maintained, routes.

The example in Section on the other hand would not benefit from large zones: the cost of maintaining the ever-changing local routes is too high, particularly since most of

the routes are so short-lived that they are never used. Instead, a zone radius of $\rho \leq 3$ would be beneficial, to ensure that the zones overlap enough to allow for route-re-

dundancy. Note, however, that reducing $\rho \rightarrow 0$ effectively turns the ZRP into a completely reactive protocol, obsoleting the advantages gained from its hybrid nature: all routing is done on-demand (using IERP), as no node is able to contact another node using IARP.

The results of showed that the IARP traffic grows with the number of nodes in a given zone, while increased mobility of a the nodes increases IERP traffic: as nodes move, the routes between zones break and need to be "re-discovered". Increasing the number N of nodes in the global network has only limited effect on the amount of pro-active traffic, since pro-active IARP updates are local to a zone. In general, it can be stated that "larger zones provide more efficient queries, which compensates for the IARP maintenance cost".

3.10 Performance gain on the example of Bordercasting

In this section, we will show how BRP can minimize the number of broadcasts significantly. First let us consider a network as shown in Figure 3.11.

The first 8 nodes connected to the node in the middle should be seen as that node's peripheral nodes (that is, interior nodes are not shown in this example), the 8 nodes on the outside as the peripheral nodes of the extended zone. As we can see, the total number of queries if no BRP would be used would sum up to be 40!

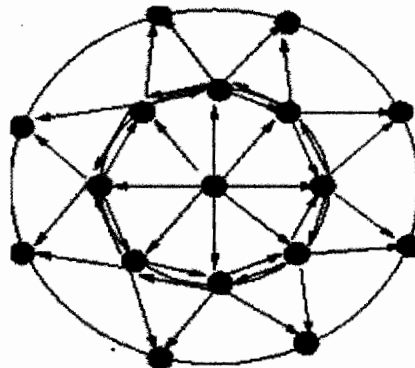


Figure 3.12: Query Flooding

Now we introduce selective bordercasting. A large number of broadcasts can be avoided, since many of the zones overlap and peripheral nodes connect to the same nodes in the extended zone. As Figure below shows, the number has been brought down to 16, a decrease of over 50%!

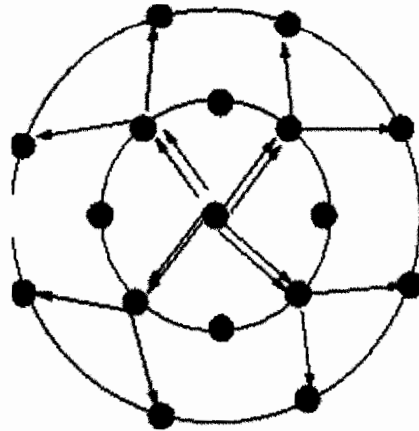


Figure 3.13: Selective Bordercasting

Chapter 4

Implementation

4.0 Implementation

The ZRP was implemented as an simulation using Network Simulator (ns-2.26) which provides following benefits simulating wireless simulations

4.1 Ns-2 and Wireless Simulations

Ns-2 provides a framework for simulation of wired and wireless networks, including some facility for emulation. The ns-2 simulator is written in C++ with a Tcl shell front-end that uses oTcl (object-oriented Tcl) libraries. Scenarios are run by feeding an oTcl script to the ns-2 executable. The output can be read directly or post-processed by an interactive graphics viewer called NAM. NAM does not allow changing parameters on the fly, it is for post-viewing of a simulation dump (a .nam file).

As of this writing the graphics viewer NAM is not advertised to work with wireless simulations, but there are apparently work-arounds¹. Generally ns-2 has a different architecture for wireless and wired node simulation. Though this report will only examine purely wireless simulations, the code can be adapted for mixed wired/wireless environments. A community of users and designers has grown around the ns-2 software. The website has pointers to the latest code and documentation, and additional modules submitted by the developer community. There are several mailing lists for users and developers, and these are conveniently archived on the site. The design section will delineate the specifications for the ZRP protocols, and describe how design decisions were made. The implementation section walks through the functionality of the code. The conclusion should tell what this module looks like, what it can be used for, and the scope of its utility.

4.3 Zone Routing Protocol

ZRP is a distributed routing protocol for Ad-Hoc networks that combines and balances the goals of both pro-active and reactive route discovery schemes, and also specifies a query-control mechanism.

The ZRP as implemented by this project refers to the specification given in the IETF Internet Drafts². The Draft specifies three protocols: IARP, BRP and IERP.

The IARP (Intrazone Routing Protocol) pro-actively and periodically distributes route information among the members of a zone, defined as the nodes that are up to and including radius R hops away. This parameter R singularly defines the behavior of the overall ZRP protocol.

In this project the IARP relies on information from the NDP (Neighbor Discovery Protocol) to track neighbors and link states with neighbors. NDP uses a beacon and acknowledgment packets to obtain link states. New neighbors or expired entries in the neighbor table trigger an immediate update to the entire zone, and periodic link updates keep the entire zone apprised of each other's states.

The prime directive of IARP is to provide a readily available route when a packet needs to be routed to a destination within the zone. But when a packet arrives for which no local route exists, the IERP is in charge of finding the so-called outer route, or external route.

The IERP (Interzone Routing Protocol) sends out requests that are forwarded, accumulating the route along the way, until a forwarding node, or relay, detects that the destination is within its zone. Then a reply is created with the full route and sent back to the source.

The flood of packets from this route discovery process can overwhelm a network, which is where the BRP (The Bordercasting Routing Protocol) steps in. BRP sends, or bordercasts, IERP requests only to peripheral nodes, also called bordercasters, which in turn forward to their bordercasters. BRP further steers outgoing queries by using QD (Query Detection) and Early Termination. Queries destined for a previously queried bordercaster, or a previously queried routing zone, are discarded. Queries are detected as they traverse a network, and this information is used to (ET) terminate redundant trajectories.

4.4 System Requirements

This project was developed under windows 2000 running cygwin on it providing the Unix environment for the prompt. The ns-2 version was "ns-allinone-2.26", which is a single tar ball with all the requisite packages that easily installs with

one command. A link to the ZRP2002 package is available on the contributed module page of the ns-2.

4.5 Design of the ZRP Module for ns-2

The ZRP module was designed within the context of the mobile node architecture of the Monarch (Mobile Network Architecture) Project, a joint project between Rice University and CMU. Their software is considered an extension of the ns-2 code-base. The mobile node differs from wired nodes in that the connectivity and other aspects of network communication depend greatly on the geographical location of the nodes. The model accounts for key aspects of mobile communication, including position and trajectory, propagation and attenuation, media access, power, and a wireless network interface. The ns-2 mobile node is fundamentally a different kind of object than a wired node. Ns-2 also provides a hybrid node for simulations requiring interfacing between wired and wireless networks. This project does not cover these kinds of simulations, but the module does not necessarily preclude them either. The modular design of the project should allow easy adaptation for hybrid nodes.

The choice of node architecture within between a DSR-like architecture or generic node was straight-forward, as the generic node was less complicated to use. The DSR node on the surface seemed useful, because it forces all packets received by the node to go to the routing agent. The ZRP too would be processing all packets, including data packets, at every hop. The workaround was to encapsulate the upper layer header inside the ZRP header in data packets. The ZRP agent at the destination node could then unwrap the original packet and submit it to the upper layer. Another implementation of ZRP as an upper layer agent was possible. But this configuration was ruled out as it poses problems in addressing, and is not as elegant as keeping the routing agent where it was designed to be. The mobile node is shown in Diagram at previous chapter. Ns-2 packets mostly contain the headers of all the protocols ns-2 simulates. To save space for massively large simulations, the header space can be pruned to include only those needed by the simulation. Route accumulation which is part of BRP and link updates required using the data portion of the packet. Using the built-in data packet routines was problematic, as there appeared to be a size limitation on how much memory could be allocated, and were finally abandoned. The workaround was

to use pointers embedded in the ZRP header pointing to where the dynamically allocated data was stored. A test was required at every step of the implementation because so many aspects of the ZRP depend on other parts of the protocol. The ns-2 trace file did not provide the detailed feedback required therefore the testing was accomplished using formatted printf's that went to standard output. The current state of each node was printed for every event that occurred for each protocol, including the neighbor table, local zone route table, link state table, and the list of peripheral nodes. Each event is described in detail. An event is printed with the node id first, then the time, and then aspects of the event delimited by the "|" character. The node tables are printed at the end and are delimited by "!". Here is an example:

```
_ 9_ [12.519476] | Node 9 sent a beacon (seq no. 4). | Node 9 started an
ack-timer to expire at 14.519476 sec. | radius is 3 | Neighbor Table: 8 |
LinkTable: 17=21 8=9 5=6 18=17 4=5 18=19 7=6 7=21 7=8 4=3 16=17 16=15 |
Routes [8 9 ] [7 8 9 ] [21 7 8 9 ] [6 7 8 9 ] Periph[6 21 ]
```

The design does NOT perform: route repair, route shortening, or the second type of query detection (QD2, i.e., hidden terminal). Accumulated routes are not distributed, but are fully included in each IERP Request and Reply. Link metrics can easily be added to the link state structure as required. The project assumed 802.11 as the MAC protocol. The dual nature of ns-2 nodes as part-Tcl object and part-C++ object provided many development challenges, but the scripting language Tcl greatly facilitates simulation runs once the C++ aspect of the code is done. The ns-2 code includes a way of binding variables between the two worlds, via the command() function. For ZRP the most important parameter is the zone radius, but other parameters have been included, including beacon period and the ability to "suspend" nodes temporarily.

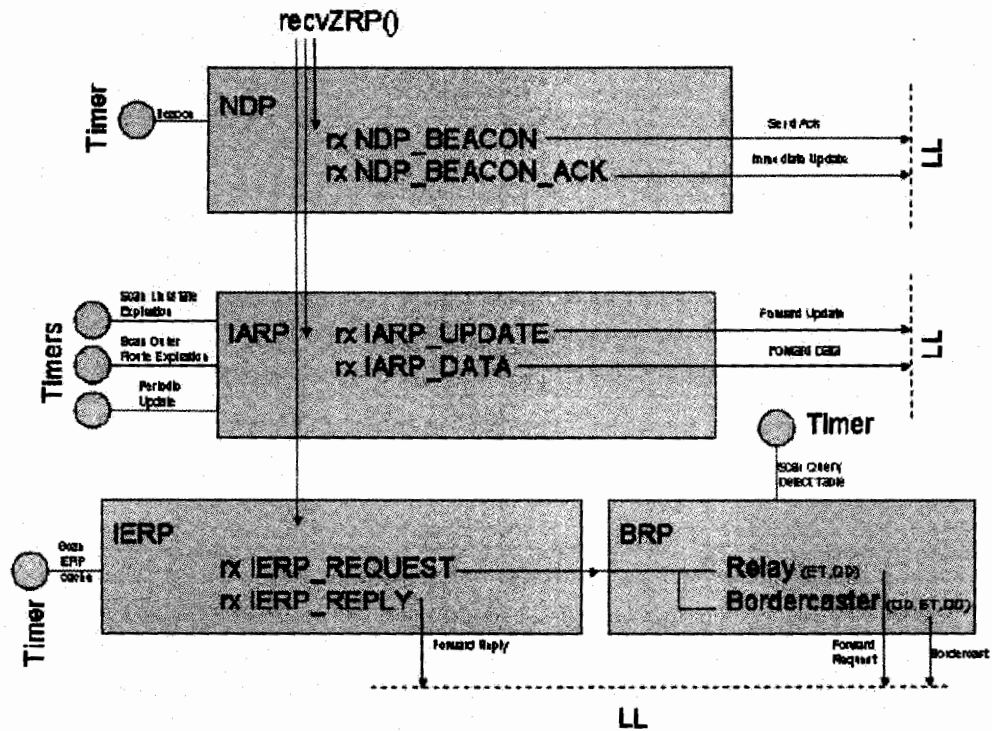


Fig 4.1 Packet Flow Diagram for ZRP

4.6 Overview of the code

The code consists of C++ code which was converted into objects files that need to be used in the tool command language. That can be simulated in the Ns-2.

The combination of C++ files consists of headers and .CC files as NS-2 supports .CC files not the .C or .CPP files. The main headers files are listed below with some of its description

Some of header files (.h) are listed below

- Brp.h
- Iarp.h
- Iarp_efficient.h
- Ierp.h
- Ndp.h
- Zrp.h
- Zrp_packet.h

Some of .cc files are listed below

- Brp.cc
- Iarp.cc
- Iarp_efficient.cc
- Ierp.cc
- Ndp.cc
- Zrp.cc
- Zrp_packet.cc

4.6.1 Brp.h

This header file is responsible for implementing bradcasting routing protocol, in which packets are sent toa all the neighbours

4.6.2 Iarp.h

This header file is responsible for implementing intra zone routing protocol, in which packets are sent to all the nodes with the specified zone.

4.6.3 Iarp_efficient.h

This header file is responsible for implementing intra zone routing protocol more efficiently, in which packets are sent to all the nodes with the specified zone.

4.6.4 Ierp.h

This header file is responsible for implementing inter zone routing protocol more efficiently, in which packets are sent to all the nodes outside the specified zone.

4.6.5 Ndp.h

This header file is responsible for implementing Neighborhood Routing Protocol more efficiently, in which packets are sent to all the nodes with the specified zone.

4.6.6 Zrp.h

This header file is responsible for implementing Zone Routing Protocol, where the zone radius and the nodes within the zone is defined.

4.7 Tcl Code

The tcl implementation of ZRP is given as follows.

```
set val(chan)      Channel/WirelessChannel  ;#Channel Type

set val(prop)      Propagation/TwoRayGround  ;# radio-propagation model

set val(netif)     Phy/WirelessPhy          ;# network interface type

set val(mac)       Mac/802_11               ;# MAC type

set val(ifq)       Queue/DropTail/PriQueue  ;# interface queue type

set val(ll)        LL                       ;# link layer type

set val(ant)       Antenna/OmniAntenna     ;# antenna model

set val(ifqlen)    50                       ;# max packet in ifq
```

```
set val(nn)      2                ;# number of mobilenodes

set val(rp)      DSDV             ;# routing protocol

#set val(rp)     DSR              ;# routing protocol

set val(x)       500

set val(y)       500

# Initialize Global Variables

set ns_          [new Simulator]

set tracefd      [open zone_rout_proto.tr w]

$ns_ trace-all $tracefd

set namtrace     [open wireless_mitf.nam w]

$ns_ namtrace-all-wireless $namtrace $val(x) $val(y)

# set up topography object

set topo        [new Topography]

$topo load_flatgrid $val(x) $val(y)

# Create God

create-god $val(nn)

# New API to config node:

# 1. Create channel (or multiple-channels);

# 2. Specify channel in node-config (instead of channelType);

# 3. Create nodes for simulations.
```

```
# Create channel #1 and #2

set chan_1_ [new $val(chan)]

set chan_2_ [new $val(chan)]

# Create node(0) "attached" to channel #1

# configure node, please note the change below.

$ns_ node-config -adhocRouting $val(rp) \

    -llType $val(ll) \

    -macType $val(mac) \

    -ifqType $val(ifq) \

    -ifqLen $val(ifqlen) \

    -antType $val(ant) \

    -propType $val(prop) \

    -phyType $val(netif) \

    -topoInstance $topo \

    -agentTrace ON \

    -routerTrace ON \

    -macTrace ON \

    -movementTrace OFF \

    -channel $chan_1_
```

```
set node_(0) [$ns_ node]

# node_(1) can also be created with the same configuration, or with a different

# channel specified.

# Uncomment below two lines will create node_(1) with a different channel.

# $ns_ node-config \

# -channel $chan_2_

set node_(1) [$ns_ node]

$node_(0) random-motion 0

$node_(1) random-motion 0

for {set i 0} {$i < $val(nn)} {incr i} {

    $ns_ initial_node_pos $node_($i) 20

}

# Provide initial (X,Y, for now Z=0) co-ordinates for mobilenodes

$node_(0) set X_ 5.0

$node_(0) set Y_ 2.0

$node_(0) set Z_ 0.0

$node_(1) set X_ 8.0

$node_(1) set Y_ 5.0

$node_(1) set Z_ 0.0
```

```
# Now produce some simple node movements

# Node_(1) starts to move towards node_(0)

$ns_ at 3.0 "$node_(1) setdest 50.0 40.0 25.0"

$ns_ at 3.0 "$node_(0) setdest 48.0 38.0 5.0"

# Node_(1) then starts to move away from node_(0)

$ns_ at 20.0 "$node_(1) setdest 490.0 480.0 30.0"

# Setup traffic flow between nodes

# TCP connections between node_(0) and node_(1)

set tcp [new Agent/TCP]

$tcp set class_ 2

set sink [new Agent/TCPSink]

$ns_ attach-agent $node_(0) $tcp

$ns_ attach-agent $node_(1) $sink

$ns_ connect $tcp $sink

set ftp [new Application/FTP]

$ftp attach-agent $tcp

$ns_ at 3.0 "$ftp start"

# Tell nodes when the simulation ends

for {set i 0} {$i < $val(nn)} {incr i} {

    $ns_ at 30.0 "$node_($i) reset";
```



```
}  
  
$ns_ at 30.0 "stop"  
  
$ns_ at 30.01 "puts \"NS EXITING...\" ; $ns_ halt"  
  
proc stop {} {  
  
    global ns_ tracefd  
  
    $ns_ flush-trace  
  
    close $tracefd  
  
}  
  
puts "Starting Simulation..."  
  
$ns_ run
```

Chapter 5

Conclusion

5.0 Results

Based on the implementation we extract following things that the performance results by the number of nodes, the zone radius, the mobility and many more . The graph obtained from the NS-2 results when compared with the other routing protocols provides its greater performance under some typical conditions.

5.1 Traffic measurements

ZRP control traffic under different query control mechanisms was measured in. The results show that the IARP traffic grows with the number of nodes in the zone, which is proportional to the “area” of the zone, ρ^2 . Therefore, the cost of maintaining an extended routing zone (in DB) is high compared to the use of only a normal routing zone (in RDB). Both RDB and DB showed a similar number of packets in IERP route discovery. However, RDB has a higher bit load, since the packets must contain the bordercast tree map. According to the effects of the query control mechanisms were significant in multiple-channel networks. In multiple-channel networks, a routing zone of radius $\rho = 2$ reduces query traffic with 50% compared to flooding ($\rho = 1$), whereas the same improvement in single-channel networks were only 15%. If RQPD is employed, the traffic is further reduced by 10%. The improvement rate slows down with increasing radius. Since the amount of control traffic depends on both node mobility and route query rate, the call-to-mobility ratio (CMR) is useful to characterize the relative traffic amounts. For large values of CMR, where mobility is relatively low, the traffic amount can be reduced with a larger radius. The cost of maintaining proactive information is low relatively to the route discovery traffic. The opposite behavior is seen for low CMR values.

In ZRP was tested in a small network with a few nodes and low traffic amounts. IARP overhead increased rapidly with increasing zone radius. Increasing velocity did not affect the IARP traffic, but the IERP overhead increased due to route repairs. Link stability increased in larger zones, since BRP utilizes local topology information to route around failed links.

5.2 Determining the routing zone radius

With the correct zone size, it is possible to reduce the control traffic to a minimum. Each network configuration has an optimal zone radius value. To determine the optimal value, it is necessary to understand how different factors influence on the traffic amount. According to simulations performed in the main factors are the zone radius ρ , network size N , node density (average number of neighbors per node) and average node velocity v (affecting route stability). Of these, only the zone radius is a configurable parameter. Because of proactive route maintenance, the amount of control traffic from IARP increases with increasing zone radius. Since IARP route updates are a local event, the network size does not affect the amount of proactive traffic. The amount of IERP traffic received by a node is independent of N as well. Instead, an increase in the network size increases the number of route queries. Thus, the amount of reactive route query traffic increases with increasing network size. Therefore, larger zone sizes are favored in large networks. Larger zones provide more efficient queries, which compensates for the higher IARP maintenance costs. The amount of control traffic largely depends on the relationship between node velocity and route usage. Higher velocity causes a linear increase in IARP routing updates and IERP route failures. If the route usage rate is considerably higher than the route failure rate, route discoveries are driven by route failures, and the traffic amount increases linearly with the node velocity. In contrast, if route usage is smaller than the route failure rate, the route query rate is independent of route stability and node velocity. In this case, the load on IARP increases with the node velocity, and a small routing zone is preferential.

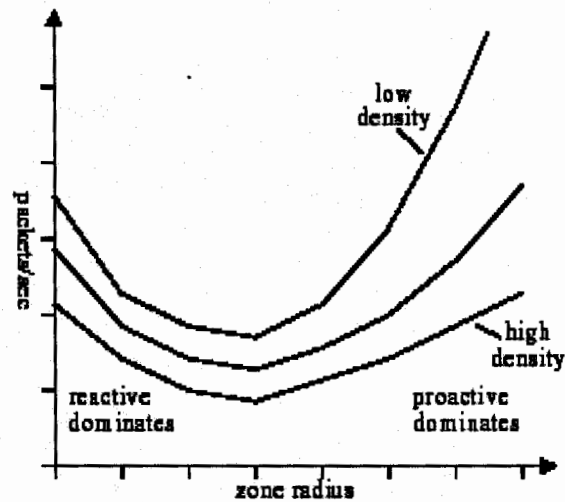


Fig 5.1 Zone Traffic Per Node

The optimal radius seems to be independent of the node density in most cases. Yet, a large increase in the node density increases the cost of IARP routing zone maintenance, which decreases the optimal routing zone radius.

5.3 Zone sizing schemes

As seen, the optimal routing zone radius depends on a number of factors, which varies for different networks and also varies within a network as a function of time. Even with perfect knowledge of all parameters, computation of the optimal radius is complicated. Even though it is possible to estimate the node density, relative node velocity, network size, the performance also depends on other factors, such as route selection criteria, route caching policies and data traffic behavior. Therefore, the paper proposes two zone sizing schemes. The “min searching” scheme searches for a local minimum of the total ZRP traffic. The routing zone radius is either incremented or decremented in steps of zone. The process is repeated in the same direction as long as the new measured traffic amount is smaller than the previous one. The found minimum is maintained until the process restarts later. The paper also suggests an automatic method for determining the time before the process is restarted. The local minimum that is found is also a global minimum since both the IARP and IERP traffic are convex functions of the zone radius. The problem with this technique is that the estimation interval must be long enough to provide accurate measurements, but a long interval may not provide adequate correlation between consecutive intervals.

The other scheme is based on the relationship between IARP and IERP traffic. When the zone radius is less than the optimal and the ZRP traffic is more than optimal, the traffic is dominated by IERP queries. If the zone radius is larger than the optimal and the traffic is more than optimal, most of the traffic is IARP route updates (see Figure 8). This property is used in the “traffic adaptive” scheme. The ratio of IERP and IARP traffic is compared with a threshold. The zone size is increased if IERP/IARP is larger than the threshold and reduced if less. A hysteresis value is used to improve stability. In this scheme only data collected from one measurement interval is used, which improves performance in frequently changing networks. An oscillation problem may appear in the “traffic adaptive” scheme if the zone size is small. It is caused by the fact that a zone of radius one is purely reactive. To solve this problem, both the above schemes can be combined. The min searching scheme is then used when the radius is small (one or two hops) and the adaptive scheme is used otherwise.

5.4 Comparison of Routing Protocols

We graded the routing protocols according to the performance measures we have defined before, and built Table 4.1 for low mobility and Table 4.2 for high mobility accordingly. Grading results are summarized in Table 4.3. Values assigned to the protocols are the ranks, where the highest value means that the assigned protocol is the best, and lowest value one means the protocol performs worst. If protocols perform similar, they are assigned same ranks. Following tables may not be evaluated correctly by the reader, unless the ranks are re-considered by the user according the results and we have obtained by the analysis of the performance measures which we heavily discussed in previous sections. For example, although average power consumption is best for DSDV for very high mobility case, AODV is more successful in packet delivery where it is more successful therefore it has to consume much more power. Another example, TORA seems to have the best load distribution, however we know that it is not, it just consumes excessive power of all nodes in the network.

5.4.1 Route Quality

Route quality is determined by the throughput, the successful packet delivery rate, the network delay, and the hop count performance measures. DSDV has the best successful packet delivery ratio, the best network delay and the best hop count deviation. Its throughput is not satisfactory as others, but yet no protocol is superior to it. For bi-directional traffic case, AODV's success in the average delay makes it second best choice after DSDV.

5.4.2 Protocol Accuracy

It can be represented by the packet delivery ratio performance measure. The best results are exhibited by DSDV for both uni and bi-directional traffic cases.

	DSDV		DSR		AODV		TORA	
	uni	bi	uni	bi	uni	bi	uni	bi
Successful packet delivery	4	4	2	2	3	3	1	1
Average delay	4	4	2	1	1	3	3	2
Average throughput	3	3	3	3	3	3	1	1
Control packet overhead	3	4	3	2	2	4	1	1
Average hop deviation	4	4	3	3	2	2	1	1
Average power consumption	3	4	4	1	2	3	2	3
Std power cons. (Load distribution)	1	2	3	3	2	1	4	4

Table 5.1

	DSDV		DSR		AODV		TORA	
	uni	bi	uni	bi	uni	bi	uni	bi
Successful packet delivery	3	3	3	2	4	4	1	1
Average delay	4	4	2	2	3	3	1	1
Average throughput	3	3	3	3	4	4	2	2
Control packet overhead	4	4	3	3	2	2	1	1
Average hop deviation	4	4	3	3	2	2	1	1
Average power consumption	4	4	2	1	3	3	2	2
Std power cons. (Load distribution)	4	4	2	2	3	3	3	4

Table 5.2

5.4.3 Protocol Efficiency.

How efficient a protocol uses the network resources determines this outcome. The routing overhead, the path optimality, the network load and the power consumption performance measures measure this. For uni-directional traffic case DSDV seems to be the most efficient protocol. For bi-directional traffic, DSR is the most efficient routing protocol, other protocols are almost as efficient as DSR.

	Uni-directional traffic				Bi-directional traffic			
	DSDV	DSR	AODV	TORA	DSDV	DSR	AODV	TORA
LOW Mobility								
Route Quality	15	10	12	6	15	9	11	5
Protocol Accuracy	4	3	2	1	4	3	2	1
Protocol Efficiency	13	11	8	8	9	11	10	9
HIGH mobility								
Route Quality	14	11	12	5	14	10	12	5
Protocol Accuracy	3	3	4	1	3	2	4	1
Protocol Efficiency	16	10	10	7	16	9	10	8

Table 5.3

The values in Table are obtained by raw summation of the performance measures defined for the route quality, the protocol accuracy and the protocol efficiency. No weights are used, and values should be re-evaluated, as we have discussed before.

5.5 CONCLUSION

In this thesis, we have evaluated performances of common ad-hoc network routing protocols: DSDV, DSR, AODV and TORA using a detailed packet-level simulator NS. We have experimented with the bi-directional traffic scheme, TCP

traffic scheme, measured the power consumption and the network load distribution of these protocols. We simulated each protocol for ad-hoc networks with 50 nodes traveling within a 1175x750 m_ geography. Results have shown that quick management of route maintenance is an important factor that effects all the performance measures, especially the successful delivery rate at high workloads and increased speeds. All evaluated protocols created bi-directional traffic on a single route which caused a dramatic performance loss for all performance measures. None of the evaluated protocols were power aware, however AODV was power efficient for the bi-directional traffic scheme. In our simulations, DSDV had the least processing for route maintenance and the highest throughput, and TORA had the lowest throughput. TORA generated a large amount of control messages to manage DAGs, and its control messages encapsulated in IP were being dropped because of collisions which led to much more decrease in performance.

AODV used much less control messages, limited to the hosts involved in the routing process, therefore it has the highest standard deviation, which means that it distributes the load over the network in the least efficient way. Because of the same reason, AODV also had the least power consumption for the lowest workload. For very high speed networks, AODV performed to be the best in successful packet delivery and throughput. The successful packet delivery rate of DSR, which is a source routing protocol, is directly related to the generation of control messages, therefore to the frequency of data packet transmissions. TCP traffic led to multi path transmissions, which is caused by TCP retransmissions through multiple mobile hosts, therefore more throughput has been achieved and more reliable and better packet transmission has been unwittingly done. TCP causes more power consumption with increased load and network speed. For protocols having large update periods, we see that the power consumption is less evenly distributed, because of less frequent updates of routes. As a result, a new adaptive protocol for increased mobility, and a special transmission control protocol is needed. AODV is the best protocol for high-speed networks and DSDV 90 performs to be the best for low speed networks. Protocols such as AODV and TORA, instead of trying to detect the link state and discovering neighbors at IP level, should use the link layer 802.11, because frequent control messages cause collisions and losses more with increased mobility or workload.

5.5.1 Output Graphs on performance of ZRP

The following graphs show the performance of ZRP on different scenerios, the cenerios include the increase in query size, the increase in the packet seize, variation in zone radius etc.

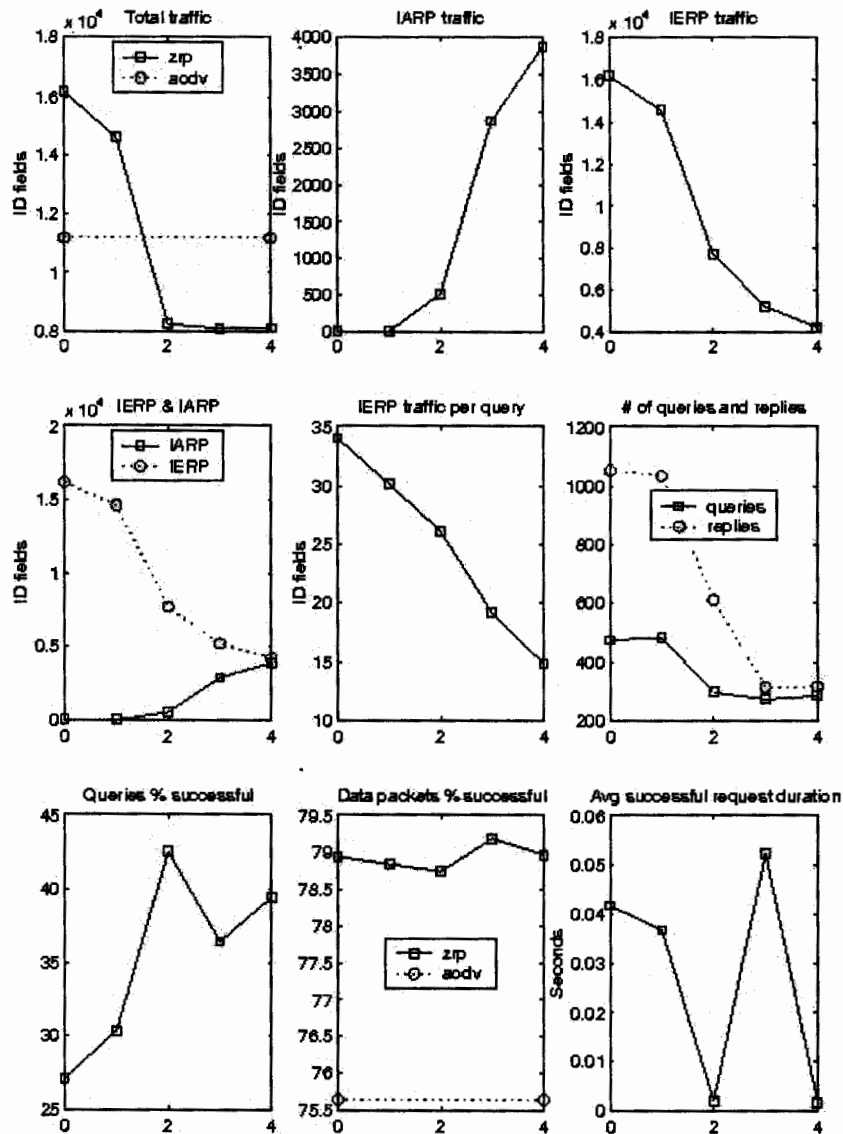


Fig 5.2 Comparative Graphs

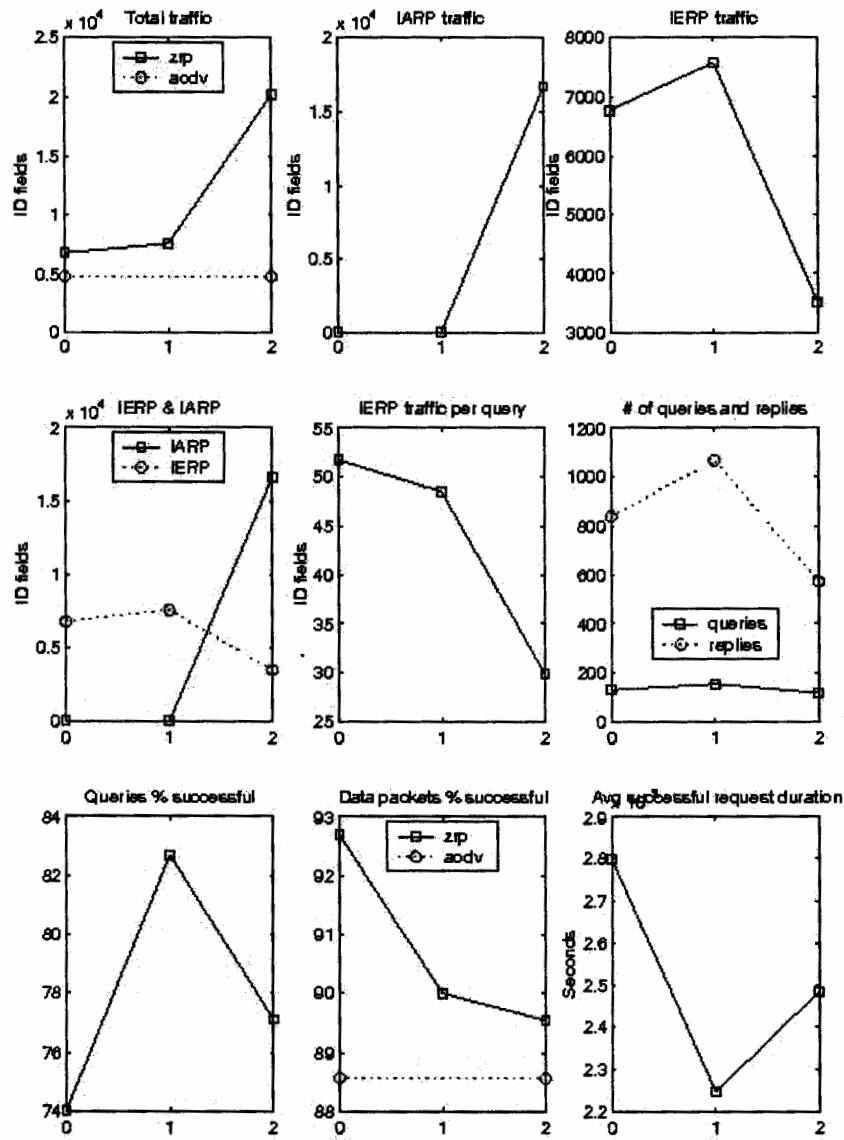


Fig 5.3 Comparative Graphs

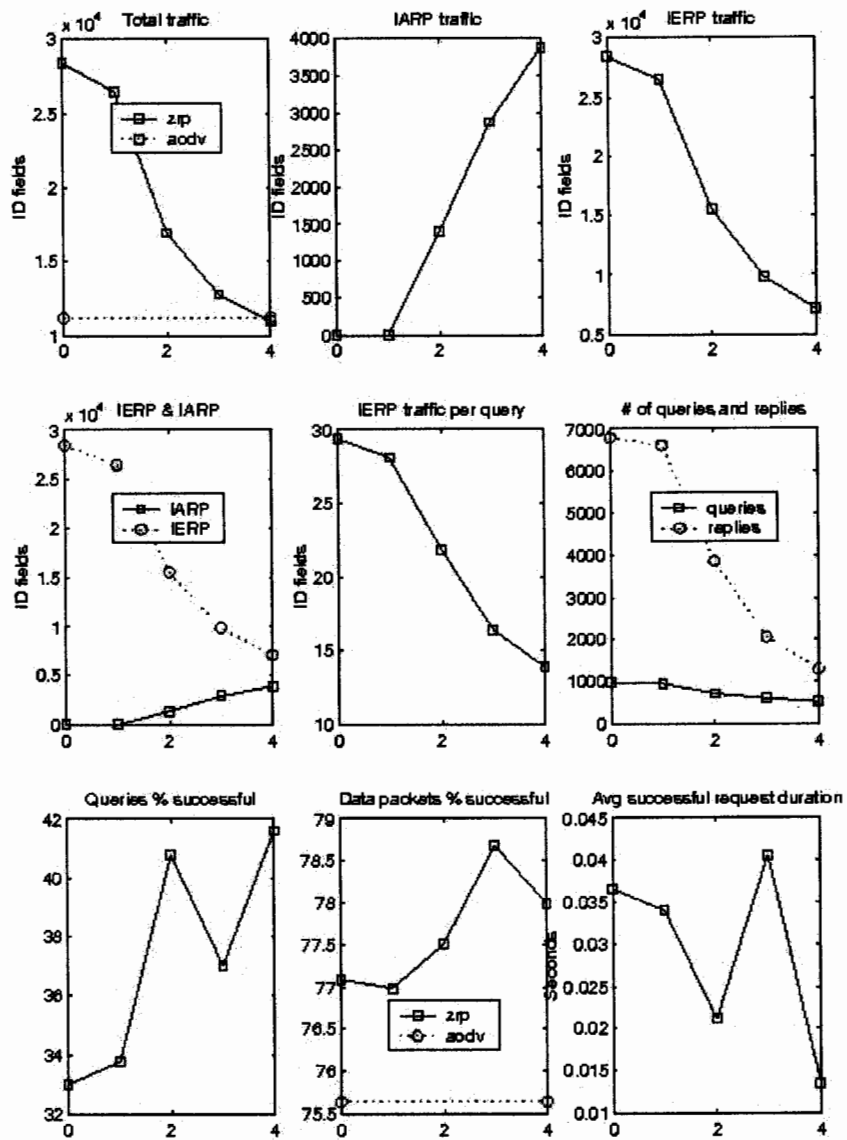


Fig 5.4 Comparative Graphs

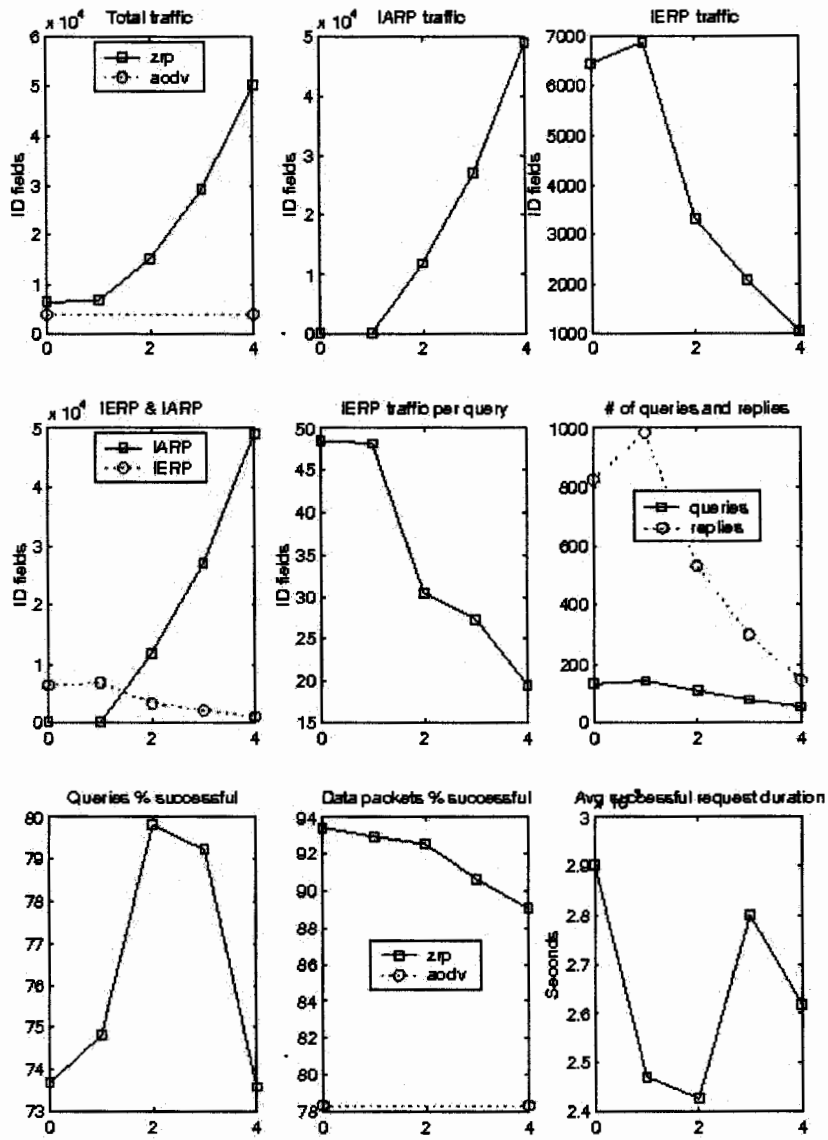


Fig 5.5.4Comparitive Graphs

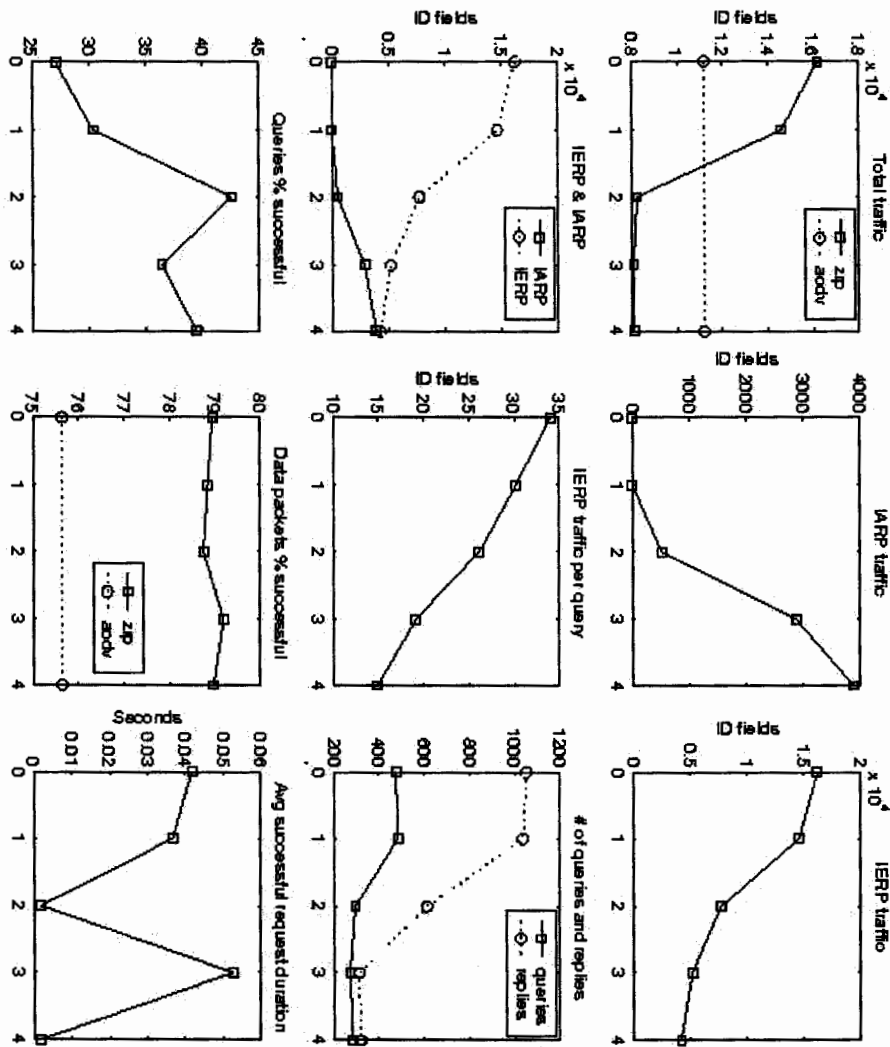


Fig 5.5 Comparative Graphs

Chapter 6

Testing

6.0 Testing

Testing phase is the most phase in the of the software development cycle, it sorts out the bugs in the software and the negatives of the software that is why the software's being developed are tested taking double as time required by implementation. There are different techniques with which software can be tested but the major are described as under.

6.1 Black Box Testing

<p>Test type: <i>Black Box Testing</i></p> <p>Developed by : Bilal</p> <p>Description: In black box testing, the user interface is exercised over a full range of inputs and the corresponding outputs are observed for correctness.</p> <p>Environment: ns-2.26 running cygwin under Win2000</p>
<p>Steps Performed for Test</p> <ol style="list-style-type: none"> 1. Specified the co-ordinates of the nodes 2. Specified the Zone radius of the network 3. The tcl file was compiled which successfully generated the trace file 4. The user opens the trace file. 5. The result can be viewed by plotting the graph out of values obtained after simulation.
<p>Result :</p> <p><input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail</p>

6.2 White Box Testing

<p>Test type: <i>White Box Testing</i></p> <p>Bilal</p> <p>Description: This testing technique do structure analysis, branch coverage, functional coverage, boundary conditions checking, and input generation are used and run against the static code.</p> <p>Environment: ns-2.26 running cygwin under Win2000</p>	<p>Developed by :</p>
<p>Steps Performed for Test</p> <p>We specifies the number of node for a large network with a very small zone radius and checked the results it was not correct</p>	
<p>Result :</p> <p><input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail</p>	

6.3 Unit Testing

Test type: <i>Unit Testing</i>	Developed by : Bilal
Description: In unit testing, different modules of the developed system are tested independently. The purpose is to determine that each module is functioning properly and to locate errors in the modules.	
Environment: ns-2.26 running cygwin under Win2000	
Steps Performed for Test We have done the unit test of each individual unit. An example test of module function parsing is given below, As ZRP is a combination of different protocols we implemented tested the individual protocols and got the correct results.	
Result : <input checked="" type="checkbox"/> Pass <input type="checkbox"/> Fail	

Appendix A

Abbreviations

Appendix A

It includes the abbreviation used for the different terminologies, which are stated as following.

DBF	Distributed Bellman-Ford routing protocol.
DSDV	Distance Source Distance Vector routing protocol
DTDV	Highly Dynamic Destination-Sequenced Distance Vector routing protocol
HSLs	Hazy Sighted Link State routing protocol
HSR	Hierarchical State Routing protocol
MMBDP	Mobile Mesh Border Discovery Protocol
MMLDP	Mobile Mesh Link Discovery Protocol.
MMRP	Mobile Mesh Routing Protocol
OLSR	Optimized Link State Routing Protocol
STAR	Source Tree Adaptive routing protocol
TBRPF	Topology Broadcast based on Reverse-Path Forwarding routing protocol
WRP	Wireless Routing Protocol
ABR	Associativity Based Routing protocol
AODV	Ad hoc On Demand Distance Vector routing protocol
BSR	Backup Source Routing protocol
DSR	Dynamic Source Routing protocol

LMR	Lightweight Mobile Routing protocol
LUNAR	Lightweight Underlay Network Ad hoc Routing
TORA	Temporally-Ordered Routing Algorithm routing protocol
CBRP	Cluster Based Routing Protocol
CEDAR	Core Extraction Distributed Ad hoc Routing
DDR	Distributed Dynamic Routing Algorithm
GSR	Global State Routing protocol
FSR	Fisheye State Routing protocol
HARP	Hybrid Ad Hoc Routing Protocol
HSR	Host Specific Routing protocol
LANMAR	Landmark Routing Protocol for Large Scale Networks
ZRP	Zone Routing Protocol
BRP	Bordercast Resolution Protocol
IARP	Intrazone Routing Protocol
IERP	Interzone Routing Protocol
DREAM	Distance Routing Effect Algorithm for Mobility.
GLS	(Grid) Geographic Location Service
LAR	Location-Aided Routing protocol
ZHLS	Zone-Based Hierarchical Link State Routing.
ABAM	On-Demand Associativity-Based Multicast

ADMR	Adaptive Demand-Driven Multicast Routing protocol
AMRIS	Ad hoc Multicast Routing protocol utilizing Increasing id-numbers.
NS-2	Network Simulator 2
NAM	Network Animator

Reference & Bibliography

1. Haas, Zygmunt J., Pearlman, Marc R.: Providing Ad-hoc connectivity With Reconfigurable Wireless Networks, Ithaca, New York
2. Elizabeth, Royer, Chai-Keong, Toh: A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks, April 1999, IEEE Personal Communications
3. Moy, J.: OSPF Version 2, IETF RFC 2178, July 1997
4. Perkins, C. E., Bhagwat, P.: Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers, October 1994, Computer Communications, pp. 234-244
5. Clausen, T., Jacquet, P., Laouiti, A., Minet, P., Muhlethaler, P., Qayyum, A., Viennot, L.: "Optimized Link State Routing Protocol", September 2001, IETF Internet Draft, draft-ietfmanet-olsr-06.txt
6. Murthy, S., Garcia-Luna-Aceves, J. J.: An Efficient Routing Protocol for Wireless Networks, October 1996, MONET, Vol 1, No 2, pp. 183-197
7. Park, V. D., Corson, M. S.: A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks, April 1997, Proc. IEEE INFOCOM '97, Kobe, Japan
8. Johnson, D. B., Maltz, D. A.: Dynamic Source Routing in Ad-Hoc Wireless Networking, Mobile Computing, T. Imielinski and H. Korth, Eds. Norwell, MA: Kluwer, 1996, pp. 153-181
9. J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Proceedings of the fourth annual ACM/IEEE international conference on Mobile computing and networking*, pages 85–97, 1998.
10. K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash. A feedbackbased scheme for improving TCP performance in ad hoc wireless networks. *IEEE Personal Communications*, (1):34–39, February 2001.
11. T.-W. Chen and M. Gerla. Global state routing: a new routing scheme for ad-hoc wireless networks. In *Proceedings of IEEE International Conference on Communications, ICC 98*, pages 171–175, June 1998.

12. E. Royer and C-K.Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks", IEEE Personal Communications Magazine, April 1999.
13. C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination Sequenced Distance Vector Routing (DSDV) for Mobile Computers", In Proc. of ACM SIGCOMM, pages 234-244, London, England, Aug. 1994.
14. S. Murthy and J.J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks", ACM Mobile Networks and Applications Journal, Special issue on Routing in Mobile Communication Networks, 1996.
15. J.J.Garcia-Luna-Aceves and M. Spohn, "Source-Tree Routing in Wireless Networks", In Proc. IEEE ICNP 99, 7th Intl. Conf. On Network Protocols, Toronto, Canada, Oct 1999.
16. C. E. Perkins, E. M. Royer, and Samir Das, "Ad Hoc On Demand Distance Vector (AODV) Routing", Internet Draft draft-ietf-manet-aodv-04.txt, Oct. 1999.
17. J. Broch, D. B. Johnson, and D. A. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks", Internet Draft draft-ietf-manet-dsr-03.txt, Oct. 1999.
18. V. Park and S. Corson, "Temporally-Ordered Routing Algorithm (TORA) Version I Functional Specification", Internet Draft draft-ietf-manet-tora-spec-01.txt, Aug 1998.
19. Z. J. Haas and M. R. Pearlman, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks", Internet Draft draft-zone routing- protocol-01.txt, Aug, 1998.
20. Z. J. Haas, "The Routing Algorithm for the Reconfigurable Wireless Networks", ICUPC'97, San Diego, CA, Oct 1997.

From : ANSInet Publications <ansinet@ansinet.org>
Sent : 03 January 2005 09:07:31
To : <bilal_nazir@hotmail.com>
Subject : Acceptance Letter: 4-ITJ

Attachment : sarwar.jpg (< 0.01 MB)

INFORMATION TECHNOLOGY JOURNAL
A Quarterly Publication of ANSInet

Dr. Sikander Hayat Khiyal
Department of Computer Science
International Islamic University H-
10, Islamabad, Pakistan

Subject: Acceptance letter of Article # 4-ITJ Entitled: Scalability Of Zone Routing Protocol (ZRP) Extensions for Mobile Ad-Hoc Networks

By: Muhammad Bilal Nazir, Sikander Hayat Khiyal, Tauseef ur Rahman

Dear Dr. Sikander Hayat Khiyal,

It is to notify that the above mentioned manuscript has been examined by the referees, in the light of their comments it has been accepted for publication in INFORMATION TECHNOLOGY JOURNAL.

It is also notified that the above mentioned manuscript will appear in the forth coming issue if author full fill all pre-requisites.

ANSInet cordially invites to please visit the journal's website <http://www.ansinet.org>

With kind regards



Muhammad Sarwar
Executive Editor

Asian Network for Scientific Information, 308-Lasani Town, Sargodha Road,
Faisalabad, Pakistan
Tel: 0092-(0)300-8653123; Fax: 0092-(0)21-5206036 E-mail: ansinet@ansijournals.com