Zone-Based Routing Protocol for High-Mobility MANET

by

Hongyan Du

A thesis submitted to the
Department of Electrical & Computer Engineering
in conformity with the requirements for
the degree of Master of Science

Queen's University
Kingston, Ontario, Canada
May 2003

0-612-81060-7

ABSTRACT

A mobile ad hoc network (MANET) is a collection of wireless mobile hosts forming a temporary network without the aid of any centralized administration or standard support services regularly available in wired networks where hosts are normally connected. In such an environment, it may be necessary for one mobile host to seek the aid of others in forwarding a packet to its destination, due to the limited propagation range of each mobile host's wireless transmissions. In high-mobility wireless ad hoc networks, the fast change of topology increases the complexity of routing. Many location-aided or location-based routing protocols [MJH01] have been recently proposed, such as LAR [KV00] and GRID [LTS01]. In this thesis, we propose a new routing protocol for high-mobility wireless ad hoc networks, namely, *Zone-Based Routing (ZBR) protocol*, where the network area is divided into fixed non-overlapping square zones, each of which has a unique zoneID to identify the zone. Every zone has a mobile terminal (MT) that acts as a zone-head whose responsibility is to be a router in the network and to maintain information of member MTs within its zone. A path is a collection of ID numbers (rather than IP numbers), which specifies the zones, rather than specific MTs, the path traverses.

Assume that every MT can obtain its location information using a positioning system, such as GPS(Global Positioning System) receiver, so that each MT can get its ZoneID through its location information obtained from GPS. We define the mobility factor of an MT to be the number of zones an MT traverses during a given fixed time unit, thus every node can collect its own mobility information. The MT with the smallest mobility factor in a zone is chosen to be its zone-head. We also define a new path stability parameter as a value related to both the mobility of nodes a path includes and the density of the zones the path covers. In this case, we select the most stable path for routing packets. Simulation results show that the probability of broken links becomes very low, which in turn greatly increases the throughput of the network. Indeed, we show that ZBR outperforms existing location-based routing protocols. As well, ZBR adapts better to large-scale, high-density and high-mobility wireless ad hoc networks.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude towards my supervisors Dr. Hossam Hassanein and Dr. Chihsiang Yeh for giving me the opportunity, guidance and supporting me throughout my research. Your tireless work efforts and serious attitude toward research were truly inspiring.

I would like to appreciate my wonderful colleague and classmates in Telecommunication Research Laboratory for their help. Special thanks to Ahmed Safwat, Nidal Nasser, Kenan Xu, Quanhong Wang, Mohammed Riyami, Hung Tam for their advice and great comments on the thesis. I am indebted to all my friends in Kingston, your friendship make my stay in Canada such a memorable experience in my life.

I wish to express my deep gratitude to my family, especially my parents. Your continued love and encouragement always make me feel active and do better.

I would like to thank my husband for his love, support and understanding. Special thanks to my son, Lele. I am very glad to see we both make big progress during the past year. I will remember the days and nights we spent together in Kingston. Without your company, I could not have accomplished this piece of work.

LIST OF ACRONYMS

λ	Packets arrival rate
ABR	Associativity Based Routing protocol
AODV	Ad Hoc On-Demand Distance Vector
AP	Access Point
BSC	Base Station Controller
CBRP	Cluster Based Routing Protocol
CBR	Constant Bit Rate
D	Width of each zone
DCF	Distributed Coordination Function
DSDV	Destination-Sequenced Distance-Vector
DSR	Dynamic Source Routing Protocol
GPS	Global Positioning System
IEEE	Institute of Electrical and Electronic Engineers
ID	Identity
IP	Internet Protocol
LAN	Local Area Network
LAR	Location Aided Routing Protocol
MAC	Medium Access Control
MANET	Mobile Ad hoc Networks
MS	Mobile Station
MSC	Mobile Switching Center
MT	Mobile Terminal

LIST OF ACRONYMS

N	Number of mobile nodes
P	Pause time
PSTN	Public Switched Telephone Network
SSA	Signal Stability-based Adaptive routing protocol
ZBR	Zone-based Routing Protocol
ZHLS	Zone-based Hierarchical Link State Routing Protocol

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

In recent years, wireless mobile networking technologies have become widely used in the computing industry and in people's daily life, mostly due to the growing prevalence of mobile computing devices. Due to its convenience, wireless networking is becoming more popular in traditional desktop computers as well, and it is expected that in the near future most computing devices will be equipped with some form of wireless technology.

There are currently two types of wireless networks. One is infrastructure-based networks such as cellular networks and wireless Local Area Networks (wireless LANs), where a static infrastructure such as wireless base stations or access points (APs) are set up ahead of time to provide the wireless devices with connectivity. In cellular networks, the whole service area is partitioned into several smaller service regions called cells (Figure 1.1). In mobile-telephone networks these cells are usually hexagonal. Frequencies allocated to the service are re-used in different cells. The reason for cells to be hexagonal is related to the efficiency of frequency reuse.

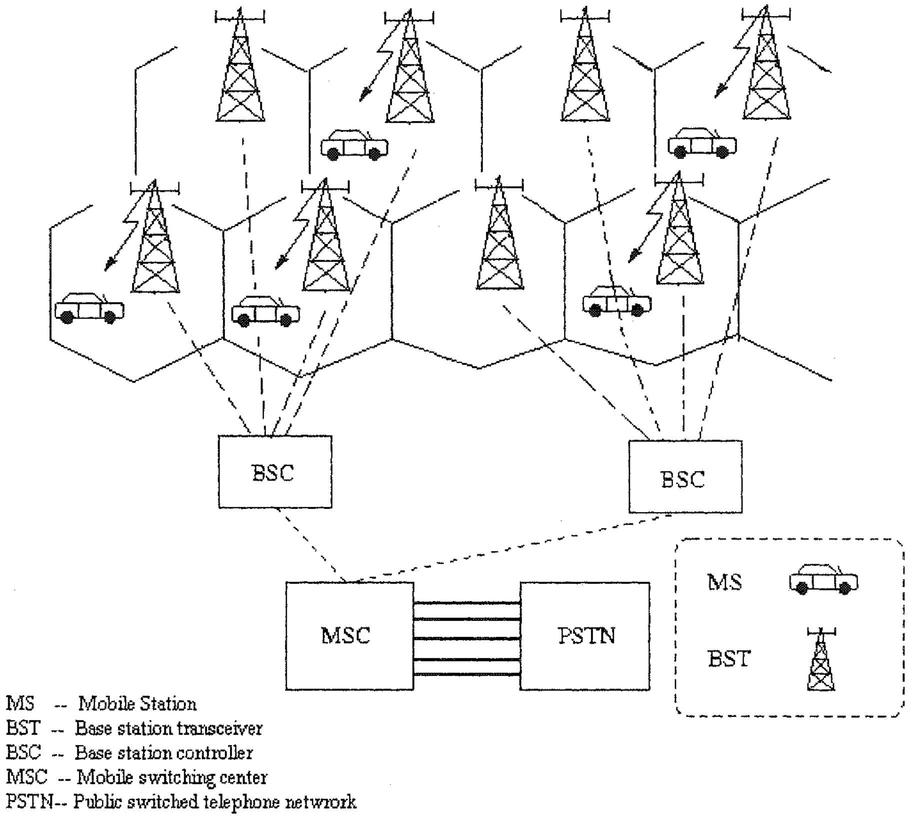


Figure 1.1 Cellular networks

In infrastructure-based wireless LANs, computing devices are connected to the Internet through wireless APs, as shown in Figure 1.2.

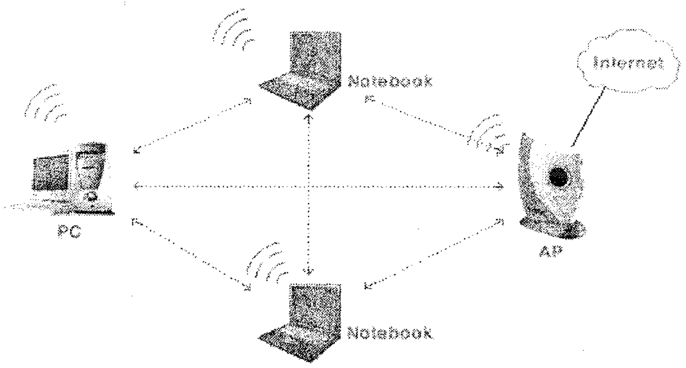


Figure 1.2 Infrastructure-based Wireless LAN

The other type is infrastructureless wireless LANs, namely *wireless mobile ad hoc networks (MANET),* which are a collection of wireless mobile hosts forming a temporary network without the aid of any centralized administration or standard support services regularly available in wired or cellular networks. Wireless ad hoc networks are fast to deploy and suitable for situations where setting up or maintaining a communicating infrastructure is difficult or infeasible, such as disaster areas, conference and battlefield environments. Due to considerations such as radio power limitation, power consumption and channel utilizations, a mobile terminal (MT) may not be able to communicate directly with other MTs in a single-hop manner.

In this case some intermediate nodes must act as routers to forward packets from the source node to the destination node. A simple example is shown in Figure 1.3. When node A wants to communicate with C, it must depend on node B to be the router in between. Since mobile nodes in such networks can move arbitrarily, the ad-hoc network topology changes frequently and unpredictably. Moreover, the bandwidth of the wireless channel is limited. The scarce bandwidth is not efficiently utilized by the effects of multiple access, signal interference and channel fading. Battery power is also a limited resource. Such limitations and constraints make research on wireless ad hoc network more challenging. A working group called "MANET" has been formed by the Internet Engineering Task Force (IETF) to study the related issues and stimulate research in wireless mobile ad hoc networks.

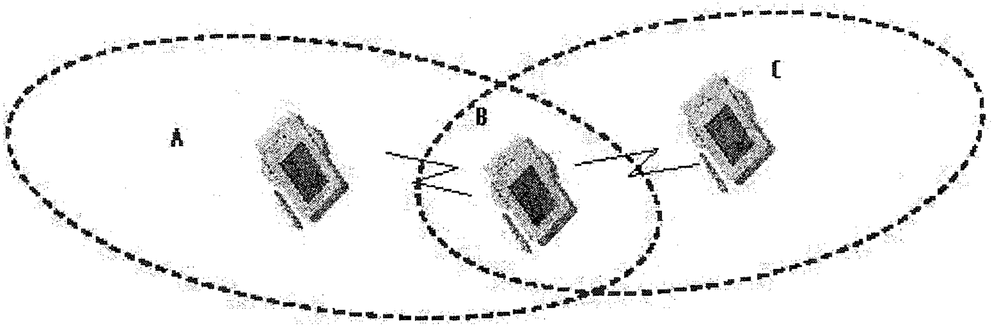


Figure 1.3 Simple example of a wireless ad hoc network

Since routes in ad hoc networks may span multiple hops because of the limited propagation range of wireless radio waves and since nodes in the network move freely and randomly, routes often get disconnected. Routing protocols are responsible for maintaining and reconstructing the route in a timely manner as well as establishing a durable path.

In high-mobility networks, the probability of route breakage may be very high and the overhead of frequent re-routing may waste bandwidth and power resources. Efficient and stable route discovering is of great importance. In this thesis, we propose a novel routing protocol, *Zone-Based Routing Protocol (ZBR),* which establishes an "infrastructure" in mobile ad hoc networks by partitioning the network area into smaller fixed areas. We refer these small areas to as zones in this thesis, which are similar to cells in cellular networks. A zone-head is chosen in each zone, which is responsible for maintaining the information of its member nodes, as well as acting as a router to forward packets. We introduce two new parameters—*mobility-factor of a node* and *stability of a path*, which are collected by mobile nodes and broadcast to other nodes. A more stable route is chosen according to the parameters. We evaluate the performance of ZBR and compare it with other protocols based on computer simulation. Simulation results show that ZBR outperforms existing routing protocols in terms of lower probability of link breakage and higher network throughput. Moreover, ZBR adapts better to large-scale, high-density and high-mobility wireless ad hoc networks.

This thesis is organized as follows. Chapter 2 introduces related work of routing protocols for wireless mobile ad hoc networks. In Chapter 3, the zone-based routing protocol is described. In Chapter 4, a simulation model is developed to evaluate the proposed ZBR protocol. Finally, Chapter 5 presents conclusions and future work.

CHAPTER 2

RELATED WORK

Routing is one of the functions of the network layer. Routing protocols are used to establish a route from a source node to a destination node for a traffic flow and deliver the packets along the route to the destination.

Routing protocols in conventional wired networks can generally be categorized into two classes: *link-state routing* and *distance-vector routing* [TA96]. *Link-state routing* keeps a routing table for the complete topology of a network, which is built up to find the shortest path (in terms of the sum of link costs). All nodes using flooding periodically transmit link cost information. Each node updates its table using the new link cost information gathered. *Distance vector routing* keeps a routing table only for its outgoing transmissions. It only transmits the link cost information to its neighbors by broadcasting. All nodes calculate the shortest path to the destination using this broadcast information.

In mobile ad hoc networks, due to the limited wireless transmission range, it is usually the case that paths between source nodes and destination nodes require multiple hops.

Since there are no fixed base stations in mobile ad hoc networks, every node may act as a router to forward packets from the source to the destination. Since each mobile node can move arbitrarily, the topology of the network may change frequently and unpredictably. Changes to existing routes may be necessary in order to maintain the connection. Therefore, the routing protocol must be able to quickly detect and respond to such state changes in order to minimize degradation in services provided to existing sessions. Conventional routing protocols such as link state and distance vector routing are designed for static topologies. They may encounter difficulties to converge to a steady state in an ad hoc network with a frequently changing topology. Since the power and bandwidth are limited resources in wireless ad hoc networks, one objective for routing protocols is to use a minimal amount of network resources by decreasing the routing overhead in order to reach maximum network throughput.

Numerous routing protocols for mobile ad hoc networks have been proposed in recent years [RS96] [RT99] [JM96] [PR99] [PH99] [LTS01]. Most of them use the information concerning connectivity relations between the mobile nodes in the network to perform packet forwarding. A basic graph-theoretical, *Unit graph model* (as shown in figure 2.1), is widely used. Two nodes in the network are neighbors and thus joined by an edge if the Euclidean distance between them is at most R, where R is the transmission range of a mobile node.

Besides the connectivity between the nodes, location information of the nodes obtained from a positioning system such as the Global Positioning System (GPS) is used by some protocols [MJH00] to assist routing.

In a peer-to-peer network, every node may act as a router, as the number of mobile nodes increases, the routing tables and topology information in mobile nodes may get tremendously large. Some routing schemes use clustering techniques, which divide the nodes of the network into clusters. A cluster-head is selected in each cluster to serve as a router in the network and acts as the function of base stations in cellular networks.

In high-mobility MANET, choosing a stable path will decrease the overhead in re-routing, thus reaching a higher network performance. Some routing protocols determine a route based on the stability of the nodes in the path.

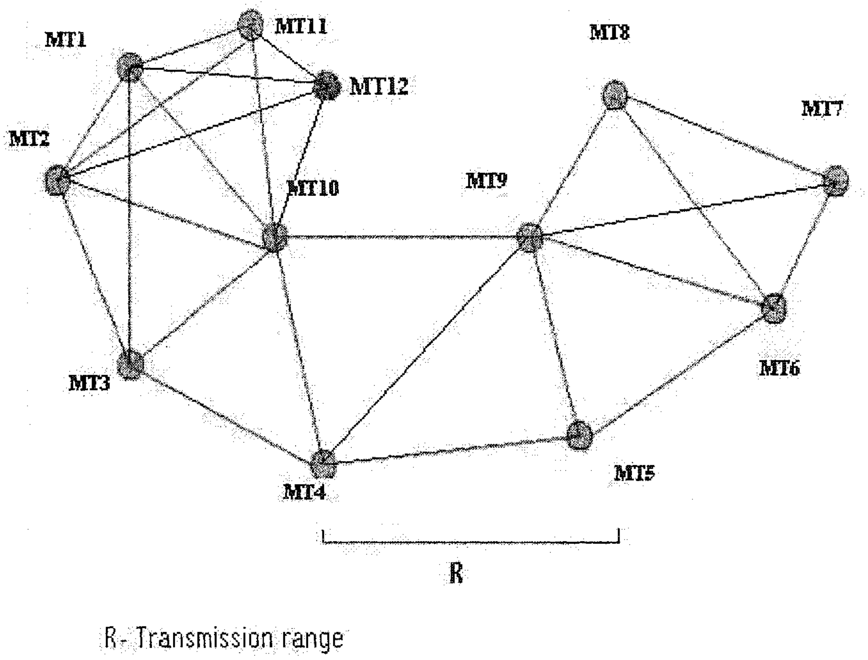


Figure 2.1 Unit graph representation of multi-hop wireless network

The rest of the chapter is organized as follows. In Section 2.1 we give an overview of existing routing protocols in flat-routed MANET. In Section 2.2, we introduce some cluster-based and hierarchical routing protocols. Section 2.3 presents location-based routing protocol. Section 2.4 introduces some routing protocols with the stable path selection. We will then give a summary and describe several potential problems of existing routing protocols in Section 2.5.

2.1 Overview of existing flat-routing protocols

In flat-routed MANET, every node may act as a router. Many routing protocols have been proposed for MANET in the past years based on such peer-to-peer communication. According to when the route discovery procedure is invoked, they can be generally classified as *proactive, reactive* or *hybrid.* In what follows we give a brief introduction and comparison among these approaches.

2.1.1 Proactive (Table-driven) routing protocols

In *proactive routing protocols*, each node maintains one or more tables containing information of every other node in the network. All nodes update these tables periodically for every change in the network so as to maintain consistency. They employ classical routing strategies such as distance-vector routing, e.g. DSDV [PB94] or link-state routing, e.g. OLSR [JMQ+01] and TBRPF [BOT01].

The *Destination-Sequenced Distance-Vector* (DSDV) routing protocol is based on the distance vector approach with some adjustment to make it more suitable for MANET. In DSDV, every mobile node maintains a routing table with entries for every possible destination node, the number of hops to reach the destination and the sequence number assigned by the destination node. The sequence number is used to avoid looping and distinguish stale routes from new ones. To maintain consistency, the update of routing table in DSDV is both time-driven and event-driven, which means that the mobile nodes transmit their routing tables to their immediate neighbors periodically or when a significant change has occurred in its table from the last update. To reduce the amount of information transmission, DSDV defines two types of updates for the routing table: full dump update and incremental update. The *full dump update* transmits all available routing information to the neighbors, whereas the *incremental update* only transmits the information that has changed since last dump. When the network is relatively stable, incremental updates are used more frequently to avoid extra traffic and full dump are relatively infrequent. In a fast-changing network, full dumps are more frequent.

Proactive routing protocols maintain routing information about all the available paths in the network even if these paths are not currently in use. The main drawback of this approach is that the maintenance of unused paths may occupy significant amount of bandwidth if the topology of the network changes frequently. As well, it does not adapt to high-mobility and low traffic MANET.

2.1.2 Reactive (on-demand) routing protocols

Reactive routing protocols (e.g. DSR [JM96] and AODV [PR99]) create route only when desired by the source node.

2.1.2.1 Dynamic Source Routing (DSR) protocol

DSR is a reactive routing protocol. The key feature is the use of source routing, which means that the source knows the complete hop-by-hop route to the destination. These routes are stored in the route cache table of source nodes.

There are two major phases: *route discovery* and *route maintenance*. When a node in the ad hoc network attempts to send a data packet to the destination, if there is no unexpired route in its cache table, it uses a route discovery process to determine a route by flooding the *route request* (RREQ) message. Each node receiving a RREQ rebroadcasts it. The procedure continues until the RREQ reaches the destination node or a node which has a route to the destination. Then the node sends a *route reply* (RREP) message back to the source.

If any link on a route is broken, the source node is notified by a *route error* (RERR) message. The source node removes any route using this link from its cache.

2.1.2.2 Ad Hoc On-Demand Distance Vector (AODV) protocol

AODV is also a reactive routing protocol, which discovers a route only when needed. However, the mechanism to maintain routing information is quite different from DSR.

A node broadcasts a RREQ when a new route to a destination is needed. The RREQ message contains the source address, a broadcast id, the destination address, the destination sequence number and a hop count. The <source address, broadcast id> pair uniquely identifies a RREQ. The RREQ is flooded through the network until it reaches a node that has a route to the destination. Each node that forwards the RREQ creates a reverse route for itself back to node S.

When the RREQ reaches the destination node or a node with a fresh enough route to that destination, that node sends a RREP message back along the reverse route and the nodes along this route set up forward route entries in their route tables which point to the node from which the RREP came. The state that is created in each node along the route from source to destination is hop-by-hop state; which is, each node remembers only the next hop and not the entire route.

In order to maintain routes, AODV requires that each node periodically transmit a HELLO message. Three consecutive failures for receiving HELLO messages from a neighbor are taken as an indication that the link to the neighbor is broken. Then any upstream node that has recently forwarded packets to the destination using that link is notified. The node that receives such message must acquire a new route to the destination using Route Discovery as described above.

2.1.2.3 Comparison of DSR and AODV

DSR and AODV both are on-demand routing protocols, which discover routes only when data packet lacks a route to the destination. Route discovery in either protocol is based on the procedures of route request and route reply. The major differences between the protocols are that DSR uses source routing and route caches, and does not depend on any periodic or timer-based activities. As well, DSR exploits route caching aggressively and maintains multiple routes for each destination. AODV, on the other hand, uses routing tables, which keep one route for each destination, and uses destination sequence number to avoid loop and to determine the freshness of routes.

Several researches have been done to compare the performance of DSR and AODV by simulations [CHA01] [DMJ+98]. Simulation results show that DSR outperforms AODV in terms of delay and throughput in less stressful situations (i.e. smaller number of nodes and lower traffic load or lower mobility). AODV, however, outperforms DSR in more stressful situations (e.g. heavier traffic load and higher mobility). DSR generates less routing overhead than AODV.

The main reasons that cause the significant performance differences of DSR and AODV are:

1. The use of source routing allows DSR to obtain a greater amount of routing information than AODV. In DSR, using a single request-reply cycle, the source node can get routes to each intermediate node on the route. However, in AODV, a very limited amount of routing information can be gathered, which causes AODV to rely on a route flood more often than DSR and generate significant routing overhead.
2. DSR makes use of route caching aggressively. In DSR, the destination node replies to all RREQ arriving at it. Thus, the source node learns many alternate routes to the destination from a single cycle, which will be useful when the primary route fails. In AODV, on the other hand, the destination node replies only once to the RREQ arriving first and ignores the rest RREQ packets. The routing table maintains at most one entry for each destination.
3. The specification of DSR does not contain any explicit mechanism to remove stale routes in the cache, or give priority to "fresher" routes when faced multiple

choices. In AODV, however, the fresher route is chosen (based on the destination sequence number) when faced with multiple choices.

4. AODV requires periodical HELLO messages in order to detect and respond to route breakage, hence it adapts to high mobility networks, but it incurs more communication overhead.

Reactive routing protocols maintain only the routes that are currently in use, thereby reducing the overhead on the network when only a small subset of all available routes is in use at any time. However, they still have some inherent limitations. First, since routes are only maintained while in use, it is required to perform a route discovery before packets can be forwarded between mobile hosts. This leads to a delay before the first packet to be transmitted. Second, even though route maintenance is restricted to the routes currently in use, it may still generate a significant amount of network traffic when the topology of the network changes frequently.

2.1.3 Hybrid Routing protocols

The hybrid routing approach combines local proactive routing and global reactive routing for achieving higher level of efficiency and scalability.

The *Zone Routing Protocol* (ZRP) [HP97] [PH99] is an example for this kind of routing protocols. In ZRP, the network is divided into several routing zones. A *routing zone* of a mobile station is defined as the space, which covers the MT's a pre-determined maximum number of hops away from the mobile station. The maximum number is called the *zone radius*. The protocol requires that every node knows the exact topology within its routing zone.

Inside the routing zones, the *IntrAzone Routing Protocol* (IARP) is used. This protocol can be any table-driven protocol such as DSDV. Thus, each node within a zone knows about its zone topology very well. Different zones may operate with different protocols.

When the source and destination are not in the same zone, the *IntErzone Routing Protocol* (IERP) is used to find routes between the routing zones. A source node that wishes to send a packet to a destination node will send a query to all the nodes on the border of its routing zone. The border nodes, upon receipt of the query packet, re-broadcast the query packet to their border nodes until the packet reaches the destination, or a node that lies in the routing zone of the destination. Then, the receiving node responds to the query packet and indicates the forwarding path in the reply packet.

2.2. Clustering and hierarchical routing protocols

Generally, since there is no fixed infrastructure in MANET, every mobile node may act as a router. In flat-routed MANET, the amount of information required to keep each node up-to-date with respect to the changes of network topology is proportional to at least rN, where r is the rate of the topology change and N is the number of nodes. Since r is proportional to N, routing information grows at least as fast as N^2. But network capacity increases slower than, or at most linearly with, N. Hence flat routing is not scalable to large N [RS96].

The hierarchical network structure is an effective way to organize a network comprising a large number of nodes. In a single hierarchy (as shown in Figure 2.2), nodes are divided into clusters which can be overlapping (e.g. CBRP [JLT98]) or non-overlapping (e.g. Grid [LTS01], FZBR [DHY01]). There usually is a cluster-head in each cluster. A multi-level hierarchy [ICP+99] has nodes organized in a tree-like fashion with several levels of cluster-heads.

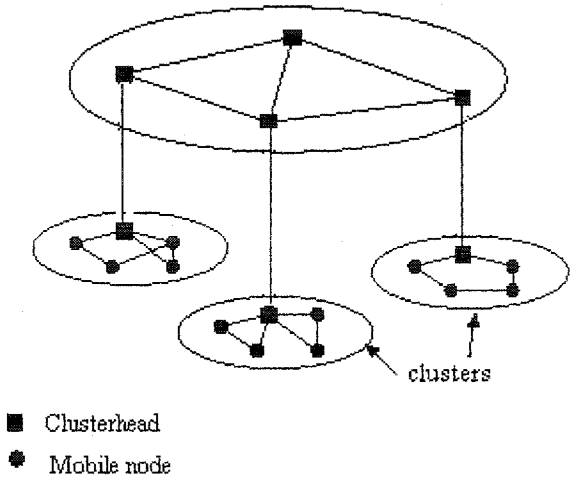


Figure 2.2 Hierarchical networks

Several clustering algorithms have been proposed in the literature [HT01] [EFB84] [L95] [KK77] [TRT+89]. These protocols group the mobile nodes and select cluster-heads in different manners. Some use the properties of nodes, such as the lowest ID [S00]; Some of them use the properties of links, such as the highest connectivity within its neighbors [MZ01]; some others are access-based, such as [HT01] where a new node will join the cluster whose cluster-head's HELLO message is received first.

Many routing protocols based on hierarchical network architecture have been proposed, such as CBRP [JLT98], max-min [DRT+00], and VBS protocol [S00]. In these protocols, nodes are divided into clusters, and a cluster-head is elected in each cluster. Only cluster-heads can act as routers. By hierarchically structuring the network topology, updating routing information can be reduced to the exchange of aggregated information between clusters and the exchange of detailed topology information within clusters. Besides, other advantages of topology clustering lie in medium access control [YHM02] and hierarchical address assignment.

In *Cluster Based Routing Protocol* (CBRP), the network is divided into a number of overlapping or disjoint clusters. The node with the lowest node ID is elected as the cluster-head. A node regards itself as a member node to a particular cluster if it has a bi-directional link to the cluster-head. Cluster-heads maintain the membership information for the cluster. Intra-cluster routes (source and destination are in the same cluster) are discovered using the membership information. Inter-cluster routes (source and destination are not in the same cluster) are found by flooding RREQ message similar to DSR except that RREQ messages are only processed and forwarded by cluster-heads.

Due to the mobility of mobile stations, mobile stations can join and disjoin the cluster dynamically, and the partition of clusters in the network is also dynamic. In high-mobility ad hoc networks, however, the overheads in clustering cannot be ignored.

Fixed partition of network has also been proposed in the literature [JL99] [LTS01] [DHY01]. In such schemes, it is assumed that each mobile node is equipped with GPS receiver and knows its current location. The physical network area is partitioned into smaller square areas. We will introduce such a scheme in more details in the next section.

Hierarchical routing protocols are often better suited for large-scale networks than flat routing protocols. Because hierarchical routing protocols provide global routes to network clusters, rather than individual nodes, routing table storage is more scalable. Moreover, the amount of route update messages is also more scalable. To some extent, the reduction in routing traffic is offset by extra mobility management overhead in cluster formation and cluster-head selection. However, it is quite common that mobile nodes are naturally clustered together in the existing organizational structure where there may be a high degree of intra-cluster mobility, while inter-cluster mobility is less common.

2.3 Location-based routing protocols for MANET

Location-based routing protocols eliminate some of the limitations of topology-based routing by using additional location information. They require that information about the physical location of the participating nodes be available.

2.3.1 Positioning Systems

One approach to obtain the location of a device is using GPS (Global Positioning System) [KD96], which is a worldwide, satellite-based radio navigation system. The system consists of 24 satellites in six orbital planes operating in circular 20,200 km orbits at an inclination angle of 55 degrees and with a 12-hour period. The satellites transmit navigation messages including their orbital elements, clocks and statuses, which can be used by a GPS receiver to determine its position, velocity and time. To determine the receiver's longitude and latitude, three satellites are needed. To determine the receiver’s 3-D position, 4 satellites are required. More satellites can increase the accuracy of the location information. The error ranges are typically up to a few tens of meters. To improve the accuracy, differential GPS (DGPS) uses ground stations to assist. The system can reduce the error range to be less than a few meters [K99]. GPS can be used anywhere near the surface of the Earth.

Short-range radios or infrared sensors can be used for indoor location identification. An example is the *Active Badge System*, which was developed by the Cambridge University [HH94]. A badge is a small device that transmits a unique infrared signal periodically. A set of sensors which can detect the badges' signals are connected by a wire network. From the information provided by these sensors, the badge's location can be determined.

Another approach to obtain the MTs' location was proposed in [CHH01]. The distance between neighboring nodes can be estimated according to the incoming signal strengths. Relative coordinates of neighboring nodes can be obtained by exchanging such information between neighbor nodes.

2.3.2 Location-based routing protocols

Several location-aware or location-based routing protocols have been proposed by using location information obtained from GPS receivers. There are three main packet-forwarding strategies for location-based routing: greedy forwarding, hierarchical routing and restricted directional flooding. We briefly introduce these strategies in the following sections.

2.3.2.1 Greedy Packet Forwarding

Using greedy packet forwarding, the sender of a packet includes the approximate position of the destination node in the packet. This location information can be gathered by a location service. When an intermediate node receives a packet, it forwards the packet to a neighboring node which is in the general direction of the destination. The process is repeated until the destination has been reached.

There are different strategies a node can use to decide to which neighboring node a packet should be forwarded:

1) *Most Forward within R* (MFR) [TK84] (R is the transmission range of a node)--Forwarding the packet to the node that makes the most progress towards the destination. It tries to minimize the number of hops a packet has to traverse.

2) *Nearest with Forward Progress* (NFP) [HL86] --The packet is transmitted to the nearest neighbor of the sender which is closer to the destination. NFP performs better than MFR in situations where the mobile node can adapt its signal strength. If all nodes use NFP, the probability of packet collisions will be decreased significantly. Hence, the throughput of the network will be increased.

3) Forwarding packets to the neighbor closest to the straight line between sender and destination [KSU99]. It tries to minimize the spatial distance that a packet travels.

4) The sender randomly chooses a node which is closer to the destination than itself and forwards the packet to that node [NK84]. This strategy reduces the number of

operations required to forward a packet. Besides, it minimizes the accuracy of location information needed about the neighboring nodes.

5) *Least Backward (negative) Progress (LBP) [TK84]* --The packet is forwarded to the node with the least backward progress if no nodes can be found in the forward direction. It can solve the problem shown in Figure 2.3. In this example, there exists a valid path from S to D. However, S is closer to D than any of the nodes in its transmission range. Using the above greedy location-based algorithms, the route cannot be found even though it exists.

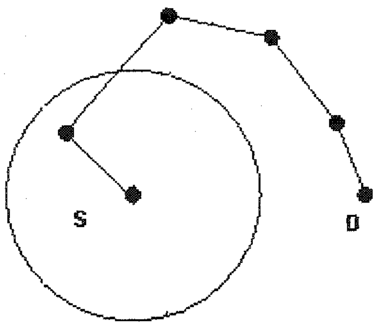


Figure 2.3 Greedy routing failures

2.3.2.2. Location-based Hierarchical Routing

Location information can also be used to assist the cluster formation in hierarchical routing. We give some examples as following:

2.3.2.2.1 Zone-based Hierarchical Link State Routing Protocol

In *Zone-based Hierarchical Link State Routing Protocol* (ZHLS) [JL99], the network is divided into non-overlapping zones. Unlike other hierarchical protocols, there is no zone-head. Therefore, ZHLS is not a cluster-based routing protocol. It defines two levels of topologies: *node level topology* and *zone level topology*. A node-level topology describes how nodes of a zone are connected to each other by physical links. A virtual link between two zones exists if at least one node of a zone is physically connected to a node in the other zone. A zone-level topology describes how zones are connected together by these virtual links. As well, there are two types of Link State Packets (LSP): *node LSP* and *zone LSP*. The *node LSP* of a node contains its neighboring nodes information, while the *zone LSP* contains the zone information. ZHLS is a proactive protocol. Any intra-zone link state change (i.e. node LSP) will be propagated locally to all other nodes within the zone, and any inter-zone link state change (i.e. zone LSP) will be propagated globally throughout the network. So each node has complete node connectivity information about the nodes in its zone and only zone connectivity information about other zones in the network. Given the <*zone ID*, *node ID*> of a destination, the packet is routed based on the zone ID until it reaches the correct zone. Then in that zone, it is routed based on the node ID.

2.3.2.2.2 GRID

GRID [LTS01] is a reactive protocol. Similar to ZHLS, it treats the geographic area as a number of logical grids, each occupying a square. In each grid, the mobile host (if any) nearest to the center will be elected as the leader (gateway). Three different zone sizes, $d_1=\frac{2r}{\sqrt{10}}$, $d_2=\frac{\sqrt{2}r}{3}$ and $d_3=\frac{r}{2\sqrt{2}}$, where r is the transmission range of each mobile node, were used by GRID-1, GRID-2, and GRID-3 respectively, and compared by simulation.

The responsibilities of a gateway include:

- Forwarding route discovery requests to neighboring grids;
- Transmitting data packets to neighboring grids
- Maintaining routes that pass the grid.

A gateway is elected following *the gateway election protocol*:

1. A gateway periodically broadcasts *GATE (g, loc)* packet to inform other nodes its existence, where *g* is its coordinate and *loc* is its current location.
2. If a mobile node doesn't receive the *GATE* packet for a period of time, it will broadcast a *BID (g, loc)* packet. If there is a gateway in the grid, the gateway will reply a *GATE* packet to reject the bid. If another node located in *loc'* in the grid which is closer to the center receives the *BID* packet, it will reply with a *BID (g, loc')* to reject the former bid. If no such packets are received by the bidding node for a predefined time period, the bidding node will elect itself as the current gateway.
3. When a gateway leaves its current grid, it broadcasts a *RETIRE (g, T)* packet, where *T* is its routing table. Every other node in the grid, when receives this packet, will inherit the routing table and compete to be the new gateway following Rule 2.

Routing is performed in a grid-by-grid manner through grid gateways. AODV (next-hop routing) is used in the protocol.

To reduce the overhead in route discovery, several strategies were proposed to confine the search range (See Section 2.3.2.3.2 for details). When a source node S needs a route to a destination node D, it broadcasts a RREQ packet to request a route to D. When a gateway node in the confined range receives this RREQ packet, it sets up a reverse pointer to the grid of the previous sending gateway and rebroadcasts this packet. When D receives the RREQ, a RREP packet will be sent from D to S following the reverse pointers, which were established earlier when propagating RREQ. When a gateway receives the RREP, it will add an entry to its routing table indicating that there is a route from itself to D via the grid from which it received the RREP packet.

2.3.2.3. Restricted Directional Flooding

Broadcasting is usually involved in the route discovery procedure. Since in mobile ad hoc networks, radio signals may overlap with each other in a geographical area, re-broadcasting nodes are usually close to each other and the timing of re-broadcasts is highly correlated, due to the host mobility, message broadcastings are executed very frequently in MANET, which may lead to storm effect traffic and cause serious redundancy, contention, and collision. These problems are called the *broadcast storm effect problem* [NTC+99].

Blind flooding may cause *broadcast storm effect problem.* Using location information, we can restrict the packet flooding within a range, thus reduce the overhead in discovering a route. Several approached have been proposed:

2.3.2.3.1 Location Aided Routing (LAR) Protocol

The Location Aided Routing protocol [KV00] proposes the use of position information to enhance the route discovery phase. Assume that mobile nodes have information about other nodes' positions, which can be used by LAR to restrict the flooding to a certain area.

When node S wants to establish a route to node D, S computes an expected zone for D based on the previous position information. A *request zone* is defined as the set of nodes that should forward the route discovery packet. The request zone typically includes the expected zone. Two request zone types have been proposed in [KV00]:

1) Rectangular geographic region. In this case, nodes will forward the route discovery packets only if they are within the rectangular region that covers the expected zone. (Figure 2.4), where $R=v\ (t_1-t_0)$. Assuming S knows the location of D at time t_0, t_1 is the current time, v is the average moving speed of D.

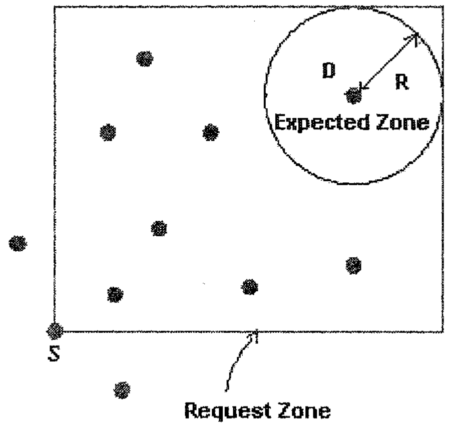


Figure 2.4 Example of request and expected zones in LAR

2) Specifying destination coordinates and the distance to the destination in the RREQ

packets. Each forwarding node fills in the distance field of the RREQ with its current distance to the destination. A node is allowed to forward the packet only if it is at most δ farther away from the destination node than the previous node. δ is a system parameter, which can be used to trade-off the probability of finding a route on the first attempt with the total cost of finding the route. It is also related to the mobility of the mobile nodes.

2.3.2.3.2 Restricted flooding in GRID

GRID [LTS 01] proposed 4 different approaches to restrict packet flooding into a smaller area for locating the destination node, see Figure 2.5:

1) Rectangle: the smallest rectangle that covers the grid of S and the grid of D (Figure. 2.5a).

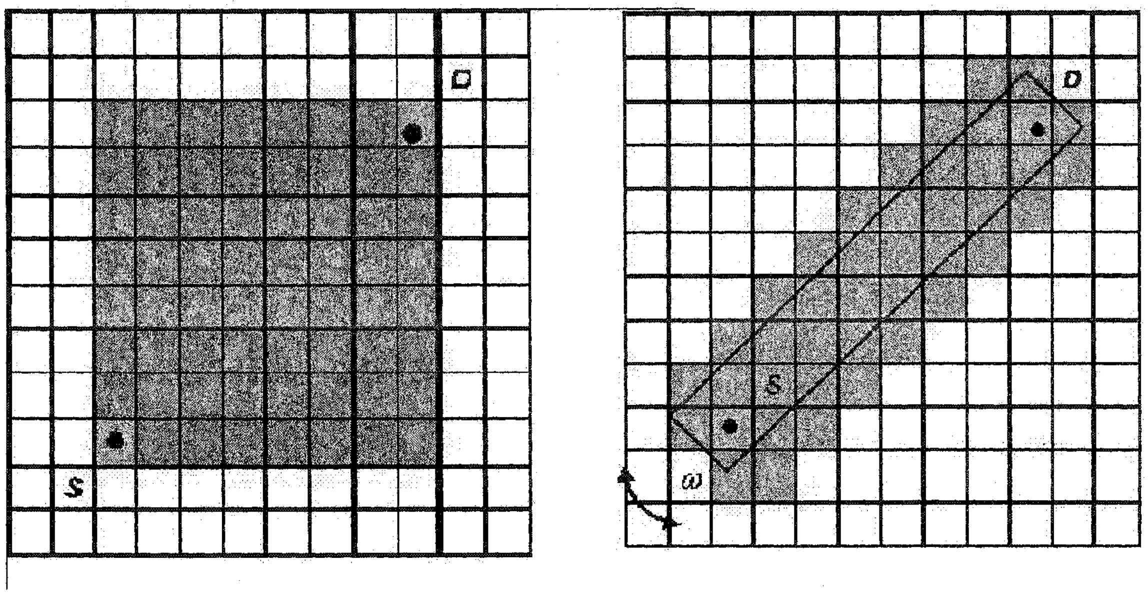
2) Bar(ω) : a bar from the grid of S to the grid of D with width ω (Figure 2.5b).

3) Fan(θ, r) : a fan from the grid of S to the grid of D with angle θ and radius r (Figure 2.5c).

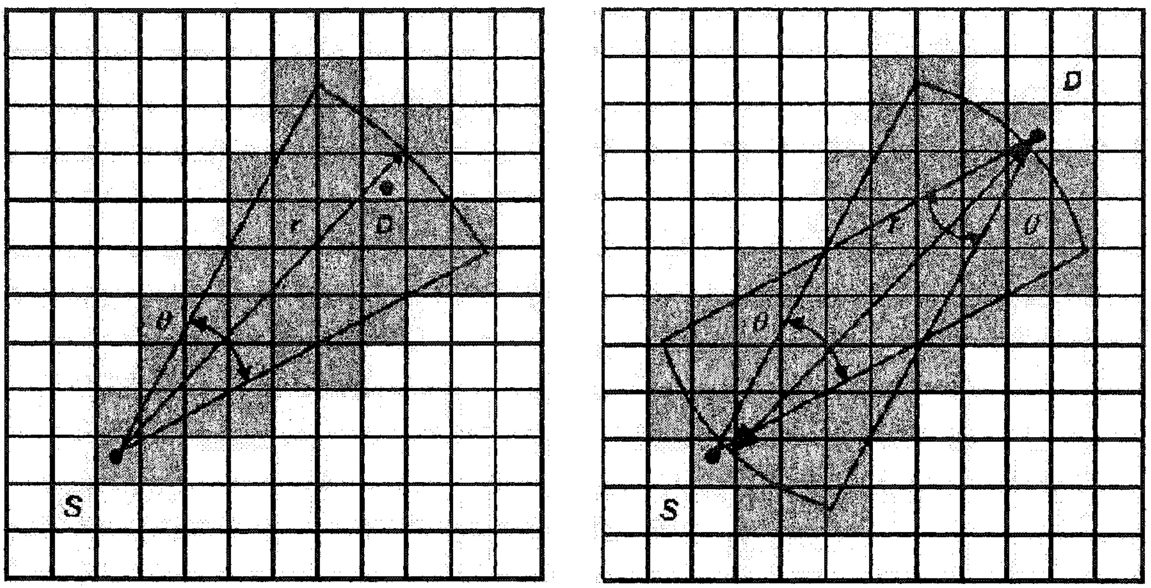
4) Two_Fan(θ, r): the intersection of two fans, one from the grid of S to the grid of D, and the other from the grid of D to the grid of S, both with angle θ and radius r (Figure 2.5d).

Routes may fail to exist in the search area. A timeout mechanism is employed, so that when this happens, another route search procedure can be initiated to search the whole area for a route.

Restricting the searching range may reduce the overhead of communications for discovering a route. But if a route cannot be found in the range, the source must rediscover in the whole area. This will incur a longer delay in discovering a route.



(a) Rectangle **(b) Bar(*w*)**



(c) Fan(*θ*, *r*) **(d) Two_Fan(*θ*, *r*)**

Figure 2.5 Ways to confine search area in GRID [LTS 01]

2.4. Stable routing protocols for MANET

The stability is a key factor to the performance of ad hoc networks. It could improve the system performance by selecting more stable routes. A stable path is helpful for reducing the probability of link breakage and the overhead in rerouting. So the stability should be an important character to be considered in routes selection criterion.

While determining the stability of a path, some routing protocols choose the stable node to be the router to deliver the packets from source node to destination node. There are some routing schemes studying on nodes' stability. The stability of a node can be defined as a value related to the node's mobility[LSG01], packets delivered number [ZWM+02], signal strength [D97] and associativity [T99].

Associativity Based Routing (ABR) Protocol [T99] defines a new metric called the *association stability* which means the connection stability of a node with respect to another node. In ABR, all the nodes periodically generate beacons to signify its existence. When a neighbor node receives a beacon, it updates its associativity tables by increasing the *associativity* tick with respect to the node from which it received the beacon. A high value of associativity tick indicates a low state of node mobility.

The route discovery phase in ABR contains the *broadcast query (BQ)* and *await-reply (REPLY)* cycle. The source node broadcasts a BQ message, each intermediate node which receives the BQ message appends its address and associativity ticks to the query packet and re-broadcasts the BQ packet to its neighbors, the procedure continues until it reaches the destination node. Then, the destination node selects the path with the highest association stability and sends a REPLY packet back to the source along this path. The route with smallest number of hops is selected if multiple paths have the same association stability. In this way, ABR chooses longer-lived routes for ad hoc networks

In *Signal Stability-based Adaptive (SSA) Routing protocol* [D97], each node maintains the *Signal Stability Table* which stores the signal strength of neighboring nodes obtained by periodic beacons from the link layer of each neighboring node. Each node classifies its neighbor as *strongly connected* or *weak connected* according the signal strength it received. In SSA, each intermediate node forwards the route request packet only if the packet is from the node with *strongly connected* and has not been previously processed (for avoid loops). The destination chooses the first arriving route request packet, and send route reply packets back along the path to the source. In this manner, SSR select a path with the strongest signal stability.

A stable adaptive optimization to DSR protocol is proposed in [ZWM+02]. In the enhanced DSR protocol, the *packets successfully delivering numbers* is used to measure the nodes' stability. The most stable route is selected using this history delivery record to send data packets from source to destination.

We know from the above arguments that the overhead of cluster formation and cluster-head election is an important factor to affect the performance of the network in cluster-based routing protocols. Some routing protocols [JLH+99] consider the mobility of nodes as a criterion to select cluster-heads to reduce the routing overhead by forming more stable clusters and cluster-heads.

2.5. Summary

In flat-routed ad hoc networks, routing protocols can be classified as proactive, reactive and hybrid approaches. Proactive routing protocols maintain consistent, up-to-date routing information from each node to every other node in the network, but the maintenance of unused paths may consume a large amount of bandwidth. Such protocols do not adapt to high-mobility networks. Reactive routing protocols create routes only when necessary, thus reduce the overhead in maintaining unused paths, but it may lead to a longer latency for the first packet to be transmitted. Hybrid routing protocols use local proactive routing and global reactive routing to achieve a higher level of efficiency. For a network comprising a large number of nodes, hierarchical network structure is effective in organizing the network, which can limit the routing tasks to a small group of nodes.

In cluster-based routing, the network nodes are grouped into different clusters, and usually a cluster-head is selected in each cluster. Only nodes chosen as cluster-heads act as routers (respond to route request messages and forward route reply messages). Thus, the communication overhead can be reduced. But the overhead in maintaining the clusters may consume large amount of bandwidth in high-mobility networks. Besides, since the traffic is concentrated at cluster-heads these cluster-heads become the traffic "hot-spots", potentially resulting in network congestion, single point of failure and power over-use. Some hierarchical routing protocols, such as the ZHLS, avoid these problems by distributing routing information to all cluster nodes, rather than maintaining a single cluster-head. These can also be solved by using load–balancing techniques in routing [HZ01]. Some routing protocols use location information to partition the network area.

With the fixed partition, the overhead in cluster formation is lower than those with dynamic clusters. A more stable path is selected in some routing protocols to reduce the probability of link breakage and overhead in re-routing, thus increase the network performance.

CHAPTER 3

ZONE-BASED ROUTING PROTOCOLS

In this chapter, we describe the Zone-based Routing protocol (ZBR). First we give an overview of the ZBR in Section 3.1. We then describe the Zone Creation, Zone-based Routing and Zone-based Route Maintenance Protocols in Section3.2 and 3.3, respectively. In Section 3.4, we present a summary of ZBR protocol.

3.1 Overview

3.1.1. Problems and motivations

The motivation of our research is to design an efficient routing protocol for large-scale, high-mobility and high-density wireless ad hoc networks. Our goal is to use simple and efficient algorithms to reduce the impact of mobility on the network performance, thus reaching a higher performance -- lower overhead, lower probability of link breakage and higher throughput for the network.

In Chapter 2, we argued that cluster-based routing schemes help to reduce the routing overhead in large-scale mobile ad hoc networks by only letting cluster-heads be responsible for forwarding route discover request. However, the overhead of cluster formation and maintenance cannot be ignored. Efficient cluster partitioning and cluster-head selection strategies are critical.

In high-mobility wireless ad hoc networks, the fast change of topology increases the complexity of routing. Location information of each MT should be important information for routing. Fast change of topology may also cause routes to be broken more often, and re-routing becomes more frequent. Therefore mobility of MTs should be an important parameter to be considered while designing routing protocols.

In this chapter, we describe a new cluster-based routing protocol based on effective use of location information. We will describe the details in the following sections concerning how to partition the network area, how to collect the mobility of MT and stability of a path, and the strategies for determining a route.

3.1.2 ZBR overview

3.1.2.1 Network partitioning

Zone-Based Routing (ZBR) Protocol partitions the ad hoc network area into non-overlapping square zones of the same size (see Figure 3.1). Each zone has a unique zoneID = (a, b). Suppose every MT equipped with a GPS receiver knows its location coordinates (x, y). Each MT can get its ZoneID by using the following formulas (Eq3-1, Eq3-2):

$$a = (\text{Location.x} - \text{Origin.x})\ \mathbf{DIV}\ D \qquad (3\text{-}1)$$

b= (Location.y- Origin.y) **DIV** D (3-2)

Where

(Location.x, Location.y) is the location of an MT obtained from its GPS receiver;

(Origin.x, origin.y) is the location of the origin in the network area. It can be set up to any location in the network in the initial stage;

D is the width (length) of each zone.

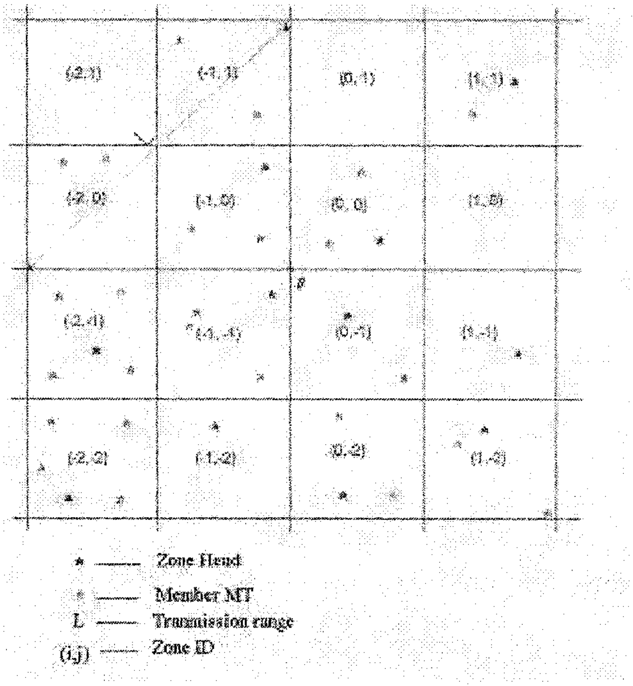


Figure 3.1 Partition of Network Area

We use two different zone-sizes (Eq. 3-3, Eq. 3-4) as shown in Figure 3.2.

$$\text{Zone-Size}_1 = (L/(2\sqrt{2})) \times (L/(2\sqrt{2})) \qquad (3\text{-}3)$$

$$\text{Zone-Size}_2 = (L\sqrt{2}/3) \times (L\sqrt{2}/3) \qquad (3\text{-}4)$$

Where L is the transmission range of an MT (assuming every MT has the same transmission range).

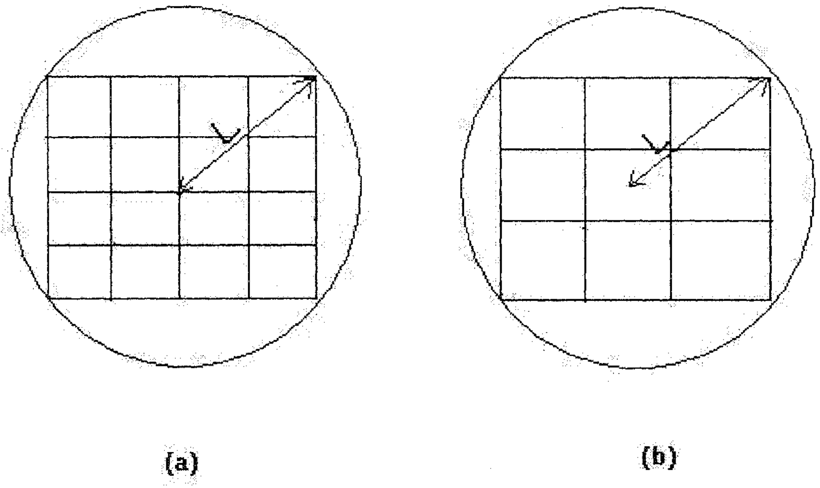


Figure 3.2 Different zone-sizes

In the first case (Figure 3.2a), where zone size=Zone-Size$_1$, any two MTs in two neighboring zones are within the transmission range of each other and can communicate directly. In other words, a node in any position of a zone can communicate directly with any other nodes in its 8 neighboring zones. In the second case (Figure 3.2b), we have a larger zone size (Zone-Size$_2$), which means a node in the center of a zone can communicate directly with any other nodes in its 8 neighboring zones. There is a zone-head in every zone. Zone-heads act as routers in the network and manage the information of the member nodes. In the first case, each zone-head is within the transmission range from all the zone-heads in its neighbor zones.

The responsibilities of a zone-head include:

- Forwarding route discovery requests to neighboring zones;
- Forwarding data packets to its neighboring zone-head (next hop) along the path;
- Maintaining routes that passes the zone: select a new zone-head and "hand off" its information to the new zone-head when it moves out of the zone;
- Maintain the information of the member nodes in the zone includes the number of member nodes, the IP address and mobility factor of each member node.

Obviously, for the same source-destination pair in a network, larger zones will have a shorter path (i.e., smaller number of hops), which can lead to a lower end-to-end delay. In the second case (Figure 3.2b), however, this may cause the *sub-optimal* problem. More precisely, since we only let zone-heads to be the routers, the zone-heads in the neighboring zones may not be in the transmission range of each other for larger zone size. Figure 3.3 illustrates an example, where ZH1 and ZH2 are two zone-heads, L is the transmission range (assume that each node has the same transmission range). We can see from the figure that there could be a link existing between zones (2, 3) and (3, 4), but since we only let the zone-head to be the router and take the responsibility of forwarding messages through the zone, but the zone-heads (ZH1 and ZH2) are not in the transmission rage of each other, in this case there is no route between these two zones. In this case, we may not find a route from the source to the destination even if there is a route from the source to the destination. The sub-optimal problem will not happen in case one where every two zone-heads in neighboring zones are in the transmission rage of each other. We will show the effect of zone-size on the performance of the protocol in Chapter 4.

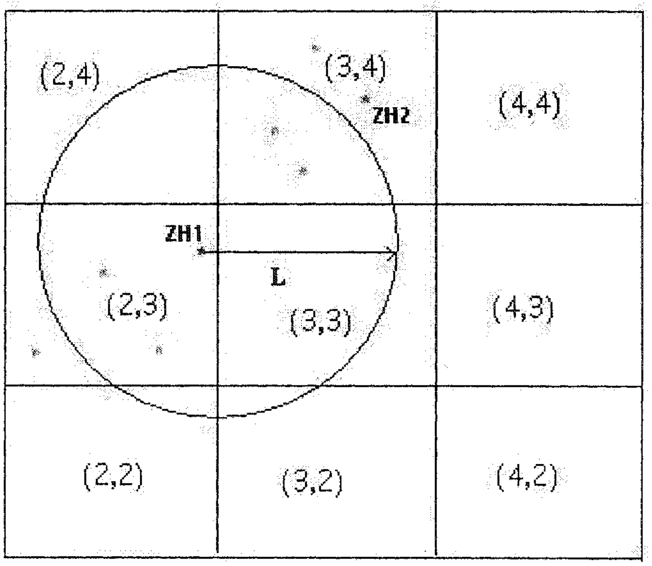


Figure 3.3 Sub-optimal problem

3.1.2.2 Routing strategy

In ZBR protocol, only zone-heads acts as routers for the network. Routing is performed in a zone-by-zone manner through zone-heads. We use source routing, which is similar to DSR, with the following differences:

1. Only zone-heads act as routers in the network, which can avoid the *broadcast storm effect problem;*

2. The destination node collects several paths when it receives RREQ from the source node in a period of time, selects the most stable path and sends RREP back along the path (unlike DSR in which whenever a RREQ reaches the destination, the destination unicasts RREP back along the path to the source. Multi-paths are collected at the source node. The source node will select a path through which data packets will be transmitted). Since unicasting is expensive in IEEE 802.11, ZBR actually reduces overhead in the transmission of RREP messages.

3. Path is a collection of zone-ID numbers which stand for which zone the path will traverse unlike DSR in which the path is a collection of IP addresses which represent which MT will be in the path and responsible for forwarding packets for the traffic.

Another way to eliminate the broadcast storm effect is to use limited broadcasting. In ZBR, we use the Rectangle Strategy, as we mentioned in Section 2.3.2.2.1, to confine the searched zone. Only zone-heads that fall in this area rebroadcast RREQ to its neighboring zones. We do not use other strategies mentioned in Section 2.3.2.2.2 because in high-mobility network, nodes move more frequently so that confining the search area to a smaller range will more likely cause failure of route discovery. This may lead to re-invoking of the route discovery, therefore increasing the overhead of routing and increasing the delay.

When a packet is to be sent from one node to another node, and if a route is not currently available in its cache table, the source node broadcasts a RREQ message to its zone-head.

When a zone-head receives a RREQ, if it is not the destination and the destination is not its member node, it rebroadcasts the RREQ; otherwise, the return path is cached in the zone-head, and when the WAITING_PERIOD_FOR_COLLECTING_ROUTES expires, the destination zone-head chooses one path with the most stable path, and sends back a RREP along the path to the source. RREP is sent in a unicasting manner. We can see that one difference from DSR is that in DSR when RREQ reaches the destination, it sends RREP back to the source immediately. In this case, more paths can be collected by source node. In ZBR, the destination node only chooses the most stable path. Since in the IEEE 802.11 protocol, unicast is expensive, we can reduce the overhead significantly.

The protocol has three components. *Zone-Creation Protocol* is used for zone creation and maintenance and zone-head election. *Zone-Based Routing protocol* is used to discover a route and send data packets following the discovered route. When a route is broken, a *Zone-based Route Maintenance Protocol* is used to fix the route. If the route can be fixed, messages follow the repaired route to the destination node; otherwise an error message is sent back to the source to find another route. We will describe the protocols in more details in Sections 3.2 and 3.3.

3.1.3 System parameters

We provide the system parameters in ZBR as follows:

(1) ORIGION; *// The position of the origin of the deployment area*

(2) L; *//Maximum transmission range of MTs*

(3) MAX_MOVING_VELOCITY;

// The maximum velocity of any MT

(4) UPDATE_TIMER;

// Time to periodically update location info.

(5) INTERVAL_OF_ROUTE_REQUEST_MSG;

//The time interval between two RREQ messages from the same source to the //same destination.

(6) WAITING_PERIOD_FOR_COLLECTING_ROUTES

//The waiting time period at the destination node from the arrival of the first //RREQ. The destination, then select a route and send RREP

Every **mobile node** maintains the following information:

(1) **IP**; *// IP address of the MT*

(2) **Location** ; *// its location from GPS receiver*

(3) **ZoneID**; *// zone_ID of its zone*

(4) **Zone_Head**; *// IP address of its zone-head*

(5) **Status**; *// = 1, stands for member MT*
// =0, stands for zone-head

(6) **Mobility_Factor ;** *//mobility of the node (see below for the definition)*

(7) **Move_out_times;** *// number of times the node is notified (when update timer expires) to move to a new zone*

(8) **Update_times;** *// number of times the location information of the node is updated*

Every **zone-head** maintains the following information:

(1) **IP;** *// IP address of the zone-head*

(2) **Location;** *// location from GPS receiver*

(3) **ZoneID;** *// The ID of the zone where it is in*

(4) **Mobility_Factor ;** *// mobility of the zonehead*

(5) **Member_MTs_list;**

// It consists of the IP addresses and Mobility_Factors of its member nodes

(6) **Number_of_member_MTs;**

// The number of member MTs in the zone

(7) **Route_cache_table;**

// Record the (S, Seq_No) pair of routing to avoid loop

(8) **Route_cache_table_at_dest;**

// list of paths collected within the waiting time from source node to destination

// node at the zone-head of destination node

Mobility_Factor is an indication of a node's mobility. It is defined as the average number of times a node moves from one zone to another during a specific period of time, which can be set to be the same as the Update_timer (See 3.2.2). This factor is used in selecting a node to be a zone-head. The mobility factor is defined as follows:

$$\text{MT.Mobility\_Factor} = \text{MT.moves\_out\_times/MT.update\_times} \qquad (3\text{-}5)$$

Only when the update timer expires, a node calculates its new location and zoneID. So in one update_timer, a node is notified of the move to a new zone at most once. By definition, the Mobility_Factor is always less than 1. From the definition, we can also see that the mobility_factor is approximate to the probability of a node moves out of a zone if the collecting time is long enough. A node with low mobility_factor means it either moves in a low speed or moves locally in a specific region. In this case, a node even if with a high moving speed, but moves locally in a zone, its mobility_factor is 0.

3.2 Zone Creation Protocol

3.2.1 Network initialization

In initialization stage, assuming every MT knows the origin of the deployment region, which can be obtained either by broadcasting the origin's information or by manual setup. Every MT calculates its zoneID through its location from the GPS receiver (see Figure 3.4).

```
Initialize( )
{ //During the initialization period, zones are created, every MT gets its ID, zone heads are chosen and informed to their member MTs
        MT.Status =0;
                // all MTs are initialized as Zone-head at the beginning
        MT. Number_Of_Member_MTs =0;
        MT.Mobility_Factor=0;
        MT.ZoneID=Get_Zone_ID(ORIGION, Location) ;
```

```
            //gets its Zone_ID by  calculating
        MT.update_times=0;
        MT.move_out_times=0;
        if Find_Zone_Head(Zone_ID)
        { // There is a zone_head in this zone
            sends Join_Msg(IP_Address, Zone_ID);
            MT.Status=1; // becomes member MT
        }
        else
        {
                MT.Status=0; //becomes Zone_head
                sends Inform_Zone_Head_Msg(IP_Address, zoneID);
                //informs its neighbor it becomes the zone head
        }
}
```

Figure 3.4 The initialization routine

3.2.2 Zone maintenance

Every MT should update its information periodically to guarantee that any member MT is in the transmission range of its zone-head. Note that it is not necessary to guarantee each node to be in the zone of its zoneID. Efficient update_timer can reduce the overhead to update information.

3.2.2.1 Considerations of update_timer

Usually update mechanisms of information can be classified as time-based, distance-based or event-based. *Time-based update* updates information periodically. *Distance-based update* updates information whenever a node moves for a certain distance. *Event-based update* updates information when an event occurs. An update-timer is used in our protocol, which is set up for every MT. When the timer expires, the MT (member MT or Zone-head) will calculate whether it is in the same Zone as before. If not, the MT joins the new zone and disjoins the old one.

To determine the value of update_timer, let us see the worst situation in Figure 3.5, where the zone-head is in one corner of the zone and a member MT is in the opposite corner of the zone. They may move in opposite direction at the maximum speed. To guarantee that any member MT is in the transmission range of its zone-head, the value of update_timer is given by:

$$\text{update\_timer} <= (L-\sqrt{2}\times D) / (2\times V) \qquad (3\text{-}6)$$

Where

L is the transmission range;

D is the width (length) of each zone;

V is the maximum_moving velocity of each mobile node.

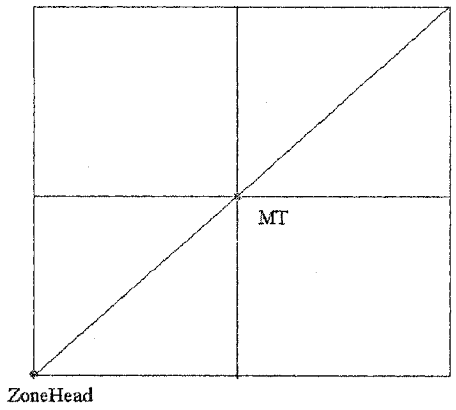


Figure 3.5 Update-timer considerations

According to equation (3-6), and for zone-size equal to zone-size$_1$ or zone-size$_2$ (defined in equations (3-3) and (3-4)), we obtain expressions for the update timer as given equations (3-7) and (3-8), respectively.

$$\text{update\_timer} = L/(4\times\text{maximum\_moving velocity}) \quad (3\text{-}7)$$

$$\text{update\_timer} = L/(6\times\text{maximum\_moving velocity}) \quad (3\text{-}8)$$

We can see from the formulae that update-timer is determined by the maximum-moving velocity, transmission range and zone size. In high-mobility networks, location information should be updated more frequently than in low-mobility networks. With the same transmission range, when the zone-size is larger, the update timer is shorter. This guarantees that at any time any member MT is in the transmission range of its zone-head. But it does not mean that member MT has to be in the same zone as its zone-head at any time. We can see that right after the zone-head information is updated after the update-timer expires, member MT may or may not be in the same zone as its zone-head since the update-timer of member MT may not expire yet. So the actual partitioning line is not fixed, and is just a reference. Thus we can avoid the *Ping Pong effect* efficiently by choosing an appropriate update-timer. (*Ping Pong Effect* means that when a node moves back and forth around a partitioning line frequently, it moves from one zone to another zone frequently and may waste resources on updating.)

3.2.2.2. Member MT routines

For every member MTs,

- When the update-timer expires, every member MT calculates its new zoneID. If it moves to another zone, it sends a Join_Message to its new zone-head, becomes its member MT or zone-head; increases its move_out_times, calculates its mobility factor, and then sends a Disjoin_Message to its old zone-head. (Figure 3.6)

```
For Every Member_MT
{
If update_timer expires {
MT.update_times++;
new_zone_ID=get_zone_ID( Location );          //calculates its new ZoneID
If  (new_zone_ID != MT.Zone_ID)
 {
        sends Join_Msg (MT.IP, new_zone_ID); //join  the new zone,
        If receives Inform_Zone_Head_Msg(zonehead_IP, new_zone_ID)
          // if there is a zone head in the new zone   becomes its member
          {
                MT.status=1;
                MT.Zone_Head= zonehead_IP;
```

```
                }
 9.         else          // if there isn't a zone head in the new zone
10.                 MT.Status=0; //become the zone head itself
11.         sends Disjoin_Msg( MT.IP, MT.zoneID );
                    // Sends disjoin-message to its original zone -head
12.          MT.moves_out_times++;
13.          MT.Mobility_Factor= MT.moves_out_times/MT.update_times;
                    //re-calculate the Mobility_Factor of the MT
        }
}
```

Figure 3.6 Timer expiry for member MT routine

- When a member MT receives Hand_Over_Msg from the previous zone-head to inform it that it will be the new zone-head, it changes its status to 0 (zone-head), updates its member_MTs_List, and informs its member MTs that it becomes the zone-head. (Figure 3.7)

```
If receives Hand_Over_Msg(Old_Zone_Head, New_Zone_Head, member_MT_list)
 {
    If (MT.IP ==New_Zone_Head)
            // if I am chosen to be the new zone-head
          {
            MT.Status=0;
                  // become the new zone_head
            updates Member_MTs_list ;
            sends Inform_Zone_Head_Msg( MT.IP, MT.ZoneID);
                //informs other MTs of the IP address of its zone_head
           }
 }
```

Figure 3.7 Member MT receives Hand_Over_Msg routine

- When a member MT receives Inform_Zone_Head_Msg , it update its Zone-Head (Figure 3.8)

```
If  receives Inform_Zone_Head_Msg(zonehead_IP, ZoneID)
{
     if (zoneID==MT.Zone_ID)
  {
   if (MT.Zone_Head!=zonehead_IP)  //another zonehead is selected
     {
         MT.Zone_Head=zonehead_IP;
         Sends Join_Msg( MT.IP, MT. zoneID);
```

```
        }
}
5. else
6.       discards the packets;
}
```

Figure 3.8 Member_MT receives Inform_Zone_Head_Msg routine

3.2.2.3 Zone-head routines

For every zone-head,

- When the update-timer expires, every zone-head calculates its new zoneID. If it moves to another zone, it calls the hand_over routine to transfer its information of its member MTs to a member-MT (if any) with **smallest mobility-factor,** which becomes the new zone-head. Then it sends Join_Msg messages to its new zone-head and becomes its member MT or new zone-head if there isn't a zone-head in the new zone (Figure 3.9).

```
For every zone-head
{
1. If update-timer expires
{
2.     MT.update_times++;
3.     new_zone_ID=get_zone_ID( Location );    //calculates its new ZoneID
4.     if (new_Zone_ID != MT.ZoneID)
         {
```

```
    old_Zone_ID=MT.ZoneID;

    MT.ZoneID=new_zone_ID;
    MT.move_out_times++;
    if (Member_MTs_List!=Null)
     { // there is at least one  member MT
        new_Zone_Head=get_new_zone_head(Member_MT_List);
          //choose the member MT with the smallest mobility_factor to be the
          //new zone_head
        sends hand_over_Msg (old_Zone_Head, new_Zone_Head,
                                                    Member_MTs_list);
          //transfers member_MTs_List to the new_zone_head
     }
    sends Join_Msg (MT.IP, MT.zoneID);  // joins the new zone
    If receives Inform_ Zone_Head_Msg (zone_head_IP, MT.zoneID)
        // if there is a zone_head in the new zone
    {
        MT.status=1;            // becomes its member MT
        MT.Zone_Head= zone_head_IP;
    }
    else        //if there isn't a zone head in the new zone
    {
        MT.status=0;            //becomes the zone head itself
        sends Inform_Zone_Head_Msg (MT.IP, MT.ZoneID);
            //Informs its member MTs
    }
sends Disjoin_Msg( MT.IP, old_Zone_ID );
    // Sends disjoin-msg  to its old  zone-head
MT.mobility_factor=MT.move_out_times/MT.update_times;
```

```
    }
}
```

Figure 3.9 Timer expiry for zone-head routine

- When the zone-head receives a join message from other MT in its zone, it adds the MT to its Member_MTs_list (Figure 3.10).

```
If receives Join_Msg (IP_Address, ZoneID)
{
    If (ZoneID==MT.ZoneID)
        { // I am in the zone
        Sends Inform_Zone_Head_Msg(zonehead_IP, ZoneID);
        updates Member_MTs_List;
        Number_Of_Member_MTs ++ ;
        }
    else //not sent to me
        discards message;
}
```

Figure 3.10 Receives Join-Msg routine

- When the zone-head receives a Disjoin_Msg from its member MT, it deletes the MT from my Member_MTs_list (Figure 3.11).

```
if receives Disjoin_Msg (IP_Address, zone_ID)
{
```

```
if (zone_ID==MT.ZoneID)
    {
    updates Member_MTs_List;
    Number_Of_ Member_MTs --;
    }
else //the message is not sent to me
    discards the message;
  }
```

Figure 3.11 Receive Disjoin_Msg routine

- When a zone-head receives Inform_Zone_Head_Msg from another zonehead in the same zone which means there are two zone-heads in the same zone. In this case the one with the smaller IP address will be the zone-head, while the other one becomes one of its member MTs(Figure 3.12).

```
If  receives Inform_Zone_Head_Msg (zonehead_IP, ZoneID)
{
  if (ZoneID==MT.ZoneID)
  {
      if (zonehead_IP<MT.IP)
      {
          MT.status=1;                //becomes member MT
          MT.Zone_Head=zonehead_IP;
          Sends Join_Msg (MT.IP, MT.zoneID);
```

```
}
6.    else
   {  //keep to be the zonehead
7.          sends Inform_Zone_Head_Msg (MT.IP. MT.zoneID);
                 //re_inform the zonehead to its member nodes
      }
}
8.else
9.    discards the packet;
}
```

Figure 3.12 Zone-head receives Inform_zone_head_Msg

- If a zone-head fails or powers-off, when a new node joins the zone, since it can't receive Inform_zone_head_Msg, it will be the zone-head itself, and send Inform_zone_head_Msg to its neighbors to inform others that it is a new zone-head.

- Neighboring zones are determined by executing the algorithm in Figure 3.13 below.

```
Boolean Is_Neighbor_Zone (ZoneID1, ZoneID2)
{
if (|ZoneID1.x-ZoneID2.x|<=1) and (|ZoneID1.y-ZoneID2.y|<=1)
    return TRUE
else
    return FALSE;
}
```

Figure 3.13 Is_neighbor_zone function

3.3. Zone-Based Routing Protocol

The *Zone-Based Routing protocol* includes *Zone-based route discovery protocol* which is used to discover a route from source to destination, and *Zone-based route maintenance protocol* which is used to fix a route when it is broken.

3.3.1. Zone-based route discovery protocol

Route discovery allows any mobile terminal to dynamically discover a route to any other mobile terminal in the ad hoc network, whether they are directly reachable within wireless transmission range or reachable through one or more intermediate network hops through other MTs.

3.3.1.1 Overview of route discovery procedure

We use the source routing approach in our routing protocol. There are two phases in the route discovery procedure: route request and route reply. When the source node S wants to transmit messages to the destination node D, it looks up its route cache to determine if it already contains a route to the destination. If it finds that a valid route to the destination exists, it uses this route to send the packet. If not, it checks whether a Route discovery procedure has already been initiated in the recent time period, which is INTERVAL_OF _ROUTE_REQUEST_MSG. Note that this can avoid frequent RREQ when a route does not exist in the network. If not, it initiates the route discovery process. That is, it sends Route_Request_Msg (RREQ) to its zone-head, and the zone-head broadcasts the RREQ to its neighbor zone-heads. To avoid loops, when a zone-head receives RREQ, it first checks whether its zoneID appears in the path. If so, it discards the RREQ; if not, it appends its zoneID to the path in the RREQ, broadcast the RREQ to its neigboring zone-heads, and this procedure continues until it reaches the destination or a zone-head which the destination node is in its Member_MTs_List.

When the destination zone-head receives the first RREQ, it sets up a timer WAITING_PERIOD_FOR_COLLECTING_ROUTES and restores the route in its Routing_table_at_dest. When the timer expires, the destination zone-head chooses the path with the highest stability (See Section 3.3.1.2) from all the paths it has collected so far and sends Route_Reply_Msg (RREP) back along the path to the source node.

When a RREP arrives at the source node, if there are data packets in the data-queue, it begins to send data packets along the path to the destination node. If an intermediate zone-head finds the next-hop cannot be reached during transmission, which means a link is broken, the current zone-head first invokes route-maintenance routine to maintain the route (for details see Section 3.3.2). If the route cannot be fixed, current zone-head sends Error-Msg back along the path to the source. The source node invokes another Route-Discovery routine to find a new route.

3.3.1.2 Definition of path stability

In ZBR, a path is a sequence of zoneID numbers, which contains the zoneIDs of the zones the RREQ traverses. We define a new parameter, *path stability*, to represent the stability of a path. The higher the stability, the smaller the probability the path will be broken. Definitions 1 and 2 below are alternative representations of path stability:

Definition1:

Assume a path is a sequence of ($ZoneID_1$, $ZoneID_2$, ..., $ZoneID_N$), indicating the route will traverse the zone-head of $zone_1$, $Zone_2$, ... $Zone_N$, whose zoneID is $ZoneID_1$, $ZoneID_2$,..., $ZoneID_N$, respectively. We define the path STABILITY as:

$$\text{STABILITY} = \frac{1}{\sum_{i=1}^{N} (MF_i / (Num_i+1))} \qquad (3\text{-}9)$$

Where,

MF_i: Mobility_Factor of the zone-head in $Zone_i$;

Num_i: Number_of MTs in $Zone_i$;

N: the number of zones the path traverses.

We use MF_i / (Num_i+1) to stand for the mobility of the $Zone_i$, and $\sum_{i=1}^{n} MFi/(Numi+1)$ to stand for the total mobility of the path.

We can see from the definition that the stability of a path is a value which is related to both the mobility-factor of the zone-heads and the density of the zones the path traverses. The higher density in the zones and the smaller mobility_factor of the zone-heads the path goes through, and the more stable the path is.

Definition 2:

Assume a path is a sequence of ($ZoneID_1$, $ZoneID_2$, ... $ZoneID_N$), which means the route will traverse the zone-heads of $Zone_1$, $Zone_2$, ... $Zone_N$, whose zoneID is $ZoneID_1$, $ZoneID_2$,... $ZoneID_N$ and in which the number of nodes is Num_1, Num_2,..., Num_N, respectively. The path stability may be defined as:

$$\text{STABILITY}= \prod_{j=1}^{N}(1-\prod_{i=1}^{Numj} MFij) \qquad (3\text{-}10)$$

Where,

MF_{ij}: Mobility_Factor of the i^{th} mobile node in a $zone_j$;

Num_j: the number of nodes in $Zone_j$;

N: the number of zones the path traverses.

In this definition, we consider the mobility of all the mobile nodes (including zone-heads and member nodes). We know from the definition of mobility_factor that it is a value to be approximate to the probability of a mobile node moves out of its zone. $\prod_{i=1}^{Numj} MFij$ is approximate to the probability of all the mobile nodes moves out of its zone in Zone$_j$, thus the zone$_j$ becomes empty (Note it is an approximation, we don't consider other nodes move in here), $1-\prod_{i=1}^{Numj} MFij$ is approximate to the probability of the zone$_j$ is not empty, which means the route is not broken here. $\prod_{j=1}^{N}(1-\prod_{i=1}^{Numj} MFij)$ is approximate to the probability of the route available.

Note that the second definition is more accurate than the first definition. However it requires that every zone-head maintain the accurate mobility information of each member node. Thus each member node must update its mobility-factor periodically even if it doesn't moves out or in. This will consume extra bandwidth. The first definition is simpler and doesn't depend on the information of member nodes. (Our simulation in Chapter 4 is based on the first definition)

3.3.1.3. Description of Zone-based Routing protocol

In route discovery procedure of ZBR, there are two kinds of control packets: Route_Request_Msg and Route_Reply_Msg.

The Route_Request_Msg consists of

1. Source_IP; *// IP address of Source node S*
2. Dest_IP; *// IP address of destination node D*
3. SequenceNo; *// for guarantee loop-free operation*
4. ZoneID; *// The current zoneID of the zone-head which send the*
   // Route_Request_Msg
5. path ; *// collection of ZoneIDs of the route the Route_Request_Msg*
   //traverses till now
6. lifetime *// remaining lifetime for guarantee to the route discovered*
   // is fresh enough
7. Stability; *//the stability info.(according to the definition of stability) that it*
   // collects through the path till now.

The Route_Reply_Msg consists of

1. Source_IP; *// IP address of Source node S*
2. Dest_IP; *// IP address of destination node D*
3. Sequence_No; *//(source_IP, Sequence_No) pair to identity a RREQ*
4. path; *//collection of ZoneIDs the route chosen by destination node*

We will show the algorithms for source node, intermediate zone-heads and destination node in the Route request procedure for source using pseudo code as follows. Figure 3.14 shows the algorithms for source node. Figure 3.15 shows the algorithms for intermediate zone-heads. Figure 3.16 shows the algorithms for destination node.

For source node S

```
SequenceNo=0;
Path="";
lifetime=LIFETIME;
        // When source node S generates a packet to transmit  to destination node D
if there is a path from S to D in the route cache
    sends_data( Source_IP, Dest_IP,path); //send data packet from S to D along the path
else  //no path from S to D in the route cache
{
        if ((Current_time-time_of_last_Route_request_MSg)< INTERVAL_OF_ROUTE_REQUEST_MSG)
                Stores data into data_queue;
        else   //begins Route_discovery
        {
            Sequence_No++;
            if S is zone-head
                broadcasts Route_Request_Msg(Source_IP, sequenceNo, Dest_IP, ZoneID, path,  stability,  lifetime);
            else  //not zone-head
                sends Route_Request_Msg  to its zone-head
        }
if  Receives Route_Reply_Msg(source_IP, sequenceNo, dest_IP,  path, lifetime)
            Starts transmission data packets along the path
if Receives Error_Msg
```

```
18.            goto10; //invokes route_discovery
```

Figure 3.14 The algorithm for source node

For intermediate zone-head

```
1.  if Receives Route_Request_Msg(Source_IP, sequenceNo, Dest_IP, ZoneID, path,
                                                                stability,
    lifetime)
2.  {
3.  if my_ZoneID∈path
4.        discards the packet; //avoid loop
5.  else
6.  {
7.    if ( [Source_IP, sequenceNo] ⊂  Route_cache_table)
8.            discards the packet; //the same request has been sent before
9.    else
10.       {
11.    path=path+ZoneID;
12.    calculate the stability of the path ;
13.    lifetime=lifetime - time_consumed; //time remaining
14.    broadcasts Route_Request_Msg (Source_IP,  sequenceNo, Dest_IP, ZoneID,
                                                   path, stability,   lifetime)
                  // broadcasts Route_Request_Msg to its neighbor zone heads
          }
       }
    }
```

```
if Receives Route_Reply_Msg (Source_IP,  sequenceNo, Dest_IP,
                                        ZoneHead, path, lifetime)
 { //Start send Route_Reply_Msg along the path to previous zonehead
if (MT.ZoneID!=ZoneHead) //it is not sent to me
        discards the packet;  //not for me
else
{
  if Source_IP∈Member_MT_List    // reach source node
          sends Route_Reply_Msg( Source_IP,  sequenceNo, Dest_IP, Source_IP,
                                  path,  lifetime)  //inform source the found path
  else
    {  //not reach S
        lifetime=lifetime - time_consumed;
        //calculating the remaining time
        if (lifetime>0)
            sends Route_Reply_Msg to its neighbor zone head
    }//else
  }
}

if Receives Send_data(Source_IP, sequenceNo. Dest_IP, path, data)
     // transmit data packet
{
   if next hop is reachable
            transmit data packet to next one
   else
            path_maintence(path[I]);
   if the path can be repaired, get the repaired path
            transmit data to next router
```

```
else
        Sends Error_Msg(Source_IP, sequenceNo,Dest_IP, error_sender, path)
                //send errorMsg along the path back to source node

 if Receives Error_Msg(Source_IP, sequenceNo, Dest_IP, error_sender, path)
    sends Error_Msg back along the path to the source node
    }
 }
```

Figure 3.15 The algorithm for intermediate zone-heads

For destination zone-head

```
if it receives the first Route_Request_Msg from the source to the destination
   {
 sets up the timer to be WAITING_PERIOD_FOR_COLLECTING_ROUTES
lifetime=lifetime - time_consumed; //time delay
if (lifetime>0)
      stores the path in the Route_cache_table_at_dest
   }
When timer expires
path=PathSelect( ); //selects the most stable path
sends Route_Reply_Msg(Source_IP, sequenceNo, Dest_IP,
                              previous_ZoneHead,  path,  lifetime );
        //sends back Route_Reply_Msg along the path last zone head
```

Figure 3.16 The algorithm for destination zone-head

3.3.2. Zone-based route maintenance protocol

Route Maintenance is used for repairing a route when it detects that a problem within a route in use occurs. Such as:

- Source node or destination node move out of wireless transmission range of next or previous hop along the route, the route can no longer be used to reach the destination;
- Any of the intermediate zone-heads in the path fails or powers off or moves out of its zone and no other mobile node in the zone can act as a new zone-head.

Route maintenance can be done by

(1) Jumping the broken node if the next-next hop in the path is reachable;

(2) Choosing another reachable node to be the next hop which is reachable by the previous node and the next node in the path.

If the route can be maintained, the data packets can be transmitted along the corrected route to the destination node; otherwise it sends Error_Msg back to source node.

Figure 3.17 shows the procedure of route maintenance.

```
Path_maintenance(path[I])      //path[I] means the $I^{th}$ node in the path
{
if path[I+1] is reachable
      path[I]=path[I+1];    //jump the failed zone head
else
{
```

```
    if $\exists$ ZoneHead $\in$( path[I-1].Neighbor_Zone_Head_List&&
                                        path[I+1].Neighbor_Zone_Head_List)
    // if there is another zone-head is both in the neighbor_Zone_Head_List of
    // the  previous and the  following zone_head of the failed node
        path[I]=ZoneHead;
    else
        sends Error_Msg(Source_IP, sequenceNo, Dest_IP, error_sender, path)
            // it sends Error_Msg back to source node.
  }
}
```

Figure 3.17 The algorithm for route maintenance

3.4 Summary

In this chapter, we provided an in-depth description of our zone-based routing protocols. We introduced how to partition the network area into zones and the strategies to select a zone-head. We presented how to use the zone-creation protocol to maintain the partition and update the information to all the MTs while topology changes and how to use zone-based routing protocols to discover a more stable route. We also introduced the zone-based route maintenance protocol which is used to fix a route while it is broken.

CHAPTER 4

PERFORMANCE EVALUATION

In this chapter, we study the performance of the ZBR protocol and compare its performance to GRID. A packet-level simulator was developed using the Java programming language to evaluate the performance of ZBR. Both ZBR and GRID use the same partition scheme to partition the network into fixed non-overlapping smaller square area. In both protocols, a path is a collection of ID numbers, rather than IP addresses, which means that which areas the path will traverse. Even if one router moves out of the transmission range of its previous or next hop, a new zone-head (gateway in GRID), if it is not the only node in its area, will be selected as the new zone-head (gateway), thus the link is still available. Both protocols adapt to large-scale, high-density and high-mobility ad hoc networks. The differences between ZBR and GRID are as following:

1. In ZBR, there is a zonehead in every zone, which is addition to the responsibilities of the gateway in GRID, is responsible for maintaining the information of the member MTs in the zone. This includes the number of member nodes and the IP address and mobility factor of each member node.
2. When a zonehead moves out of a zone, the most stable member node (if any) is chosen as the new zonehead. In GRID the one which is nearest to the center is chosen

as the gateway. Thus the number of "hand off" is reduced and the overhead in maintaining the zones can be decreased.

3. In ZBR, destination node chooses the most stable path. The probability of link breakage is, therefore, significantly reduced in high-mobility networking environments.
4. Unlike GRID, which uses AODV, ZBR uses the source routing approach as its routing protocol. Since no periodical hello messages are required, overhead in routing can be reduced.
5. Zone maintenance is based on an update-timer, which is a value related to mobility of the nodes and zone size. Only when the update timer expires and the node calculates its zoneID, then it sends Join and Disjoin message to its neighboring nodes (only if its ZoneID has changed). Thus ZBR can perform zone maintenance with lower communication overhead.

We, therefore, argue that ZBR outperforms GRID in terms of lower probability for link breakage and less overhead in routing (see numerical results in Section 4.3). Thus, ZBR can achieve a higher delivery rate, higher throughput and smaller end-to-end delay, especially in high-mobility networking environments.

4.1. Simulation model

4.1.1. Experimental setting

Our protocol evaluation is based on the simulation of 50-100 wireless mobile nodes forming a mobile ad hoc network moving about over a rectangular flat space of 500m×

300m for 200 seconds of simulation time. For each simulation, experiment is run 10 times, confidence intervals within 10% at a 90% confidence level are obtained. A rectangular area rather than a square one is chosen so as to force some paths to be longer. The Distributed Coordination Function (DCF) of IEEE 802.11 is used as the MAC layer protocol. All mobile nodes have the same transmission range of 100m. We simulate the performance according to two different network partitions (according to the parameter D described in Section 3.1.2.1). In our simulation, for the first case, D =36m, for the second case, D =47m.

4.1.2. Mobility model

In our simulation, mobile terminals move at two different speeds -- 2m/s and 20m/s. The former simulates the mobile node that in the movement of walking, while the later one simulates an MT in a moving car. We assume 50% of the MTs move with the speed of 2m/s and the other 50% of the MTs are with the speed of 20m/s. There are two mobility states for each MT: moving and pausing. An MT randomly chooses an initial position, then randomly chooses a direction, moves for a random time from 1-10 seconds, then stays for a period of time (pause time), then randomly chooses another direction and so on. Since ZBR adapts to high-mobility ad hoc networks, in the simulation, we set up the mobility of the network to be relatively high.

4.1.3. Traffic model

There are two different traffic packets in our simulation: control packets and data packets. Control packets for routing include route request messages, route reply messages and error messages. Control packets for zone maintenance include zone-head update, hand-over, join and disjoin message. We define the length of these control packets to be 32 bytes. We fix the size of data packet to be 512 bytes. There are 10 traffic pairs in our simulation. We randomly choose 10 MTs from all the mobile nodes in the network to be source nodes and randomly choose 10 other nodes to be destination nodes. Every source node generates CBR (Constant Bit Rate) traffic at a rate of 4-16 packets/sec. Table 4.1 provides our simulation parameters and their nominal values.

Parameter	**Meaning**	**Nominal Value or Range**
Tx_radius	Transmission range	100m
TX_speed	Transmission speed	2 Mbps
Length	Length of the network area	500m
Width	Width of the network area	300m
Len_data	The length of a data packet	512bytes
Len_control	The length of a control packet	32bytes
Interval_of_RREQ_Msg	The minimum interval between 2 RREQ message from the same source if there is no route existed before	2s
Waiting_period_for _collecting_routes	The waiting time period at the destination node after the first RREQ arrives the destination, then select a route and send RREP	$9x10^{-4}s$
V_motion1	The moving speed of faster nodes	20m/s
V_motion2	The moving speed of slower nodes	2m/s
Update_timer	The time of the MT to update its zone information	1.25s for faster node 12.5s for slower node

Number_of_MTs	The number of mobile nodes in the network	50~100
Zone size$_1$	The length(width) of each zone (larger zone partition)	47m
Zone size$_2$	The length(width) of each zone (smaller zone partition)	36m
Number_of_traffic_pairs	The number of the traffic pairs in the network	10
Arrival_rate	The arrival rate of packets of the CBR source	4~16packets/s
DataSendingQueueSize	The size of the mobile node's MAC queue	20 packets
Simulation time	The length of the entire simulation time	200s

Table 4.1 The simulation parameters

4.2. Discussion of the results

In our simulations, numerous experiments were conducted in order to evaluate the performance of ZBR and compare it with GRID. We change different input parameters such as the number of nodes in the network, pause time of mobile nodes, zone size and traffic load to evaluate their impacts on performance of the mobile ad hoc networks. The results are shown in Figures 4.1-4.4, where P is pause time, D is the width (length) of each zone, λ is the arrival rate and N is the number of mobile nodes in the network.

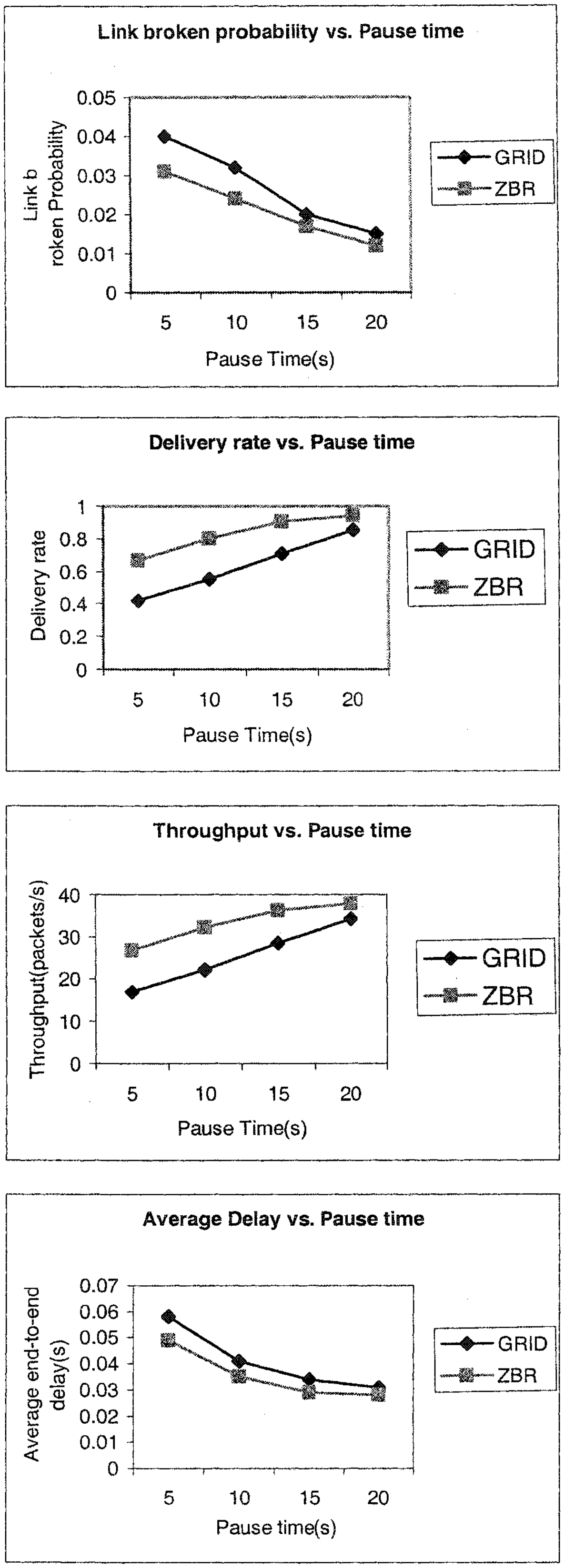
4.2.1 Comparisons of ZBR and GRID

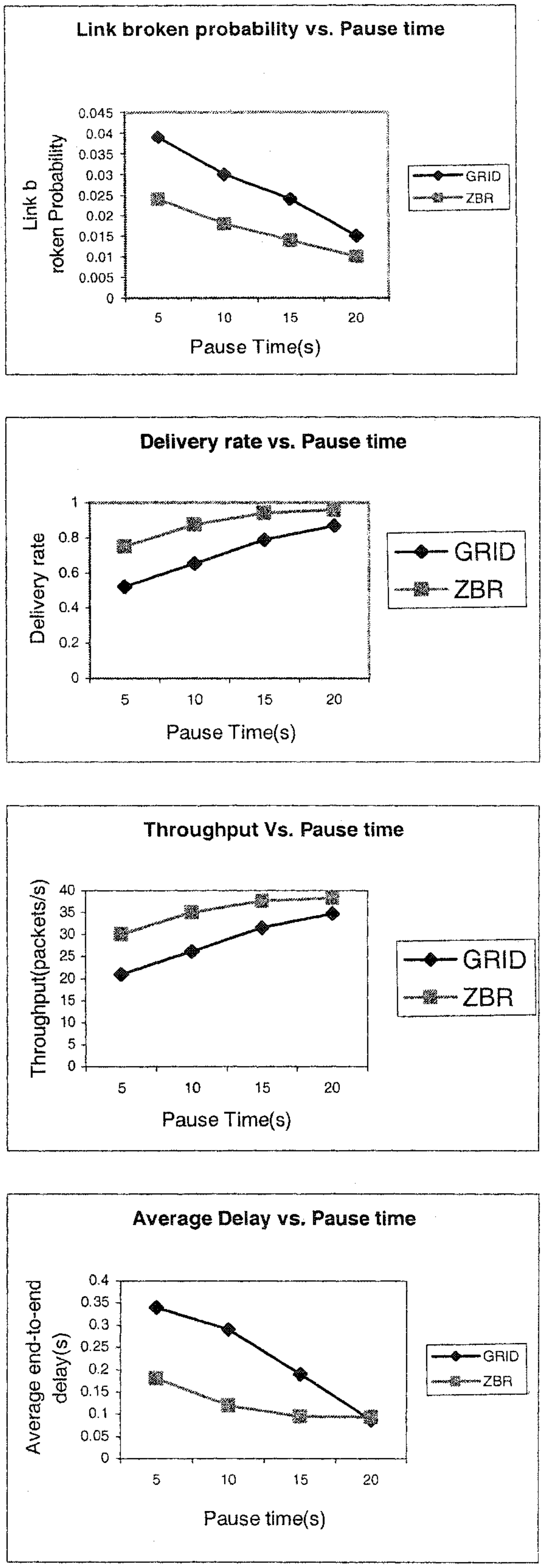
In this experiment, we compare the performances of ZBR and GRID in different network environment. We set pause time to be 5s, 10s, 15s, 20s, zonesize to be 36m and 47m, number of nodes to be 50 and 100, arrival rate to be 4 , 8, 12 and 16 packets/sec. We can see from Figure 4.1 that ZBR outperforms GRID under different traffic and network conditions.

We can see from Figure 4.1.A - 4.1.D that the link breakage probability and average end-to-end delay decrease while the throughput and delivery rate increase with the pause time for both ZBR and GRID at different number of nodes and different zone sizes. ZBR shows better performance than GRID significantly in high-mobility situation (say, P=5s) and show less significant change in performance as the pause time changes than GRID. This is because ZBR considers the mobility of mobile nodes, especially in high mobility situation, and chooses a more stable path. Thus the probability of link breakage is much less than that of GRID. But when the pause time is very large, say 20s, which means the mobility of MTs is relative small, the performance difference between ZBR and GRID is insignificant.

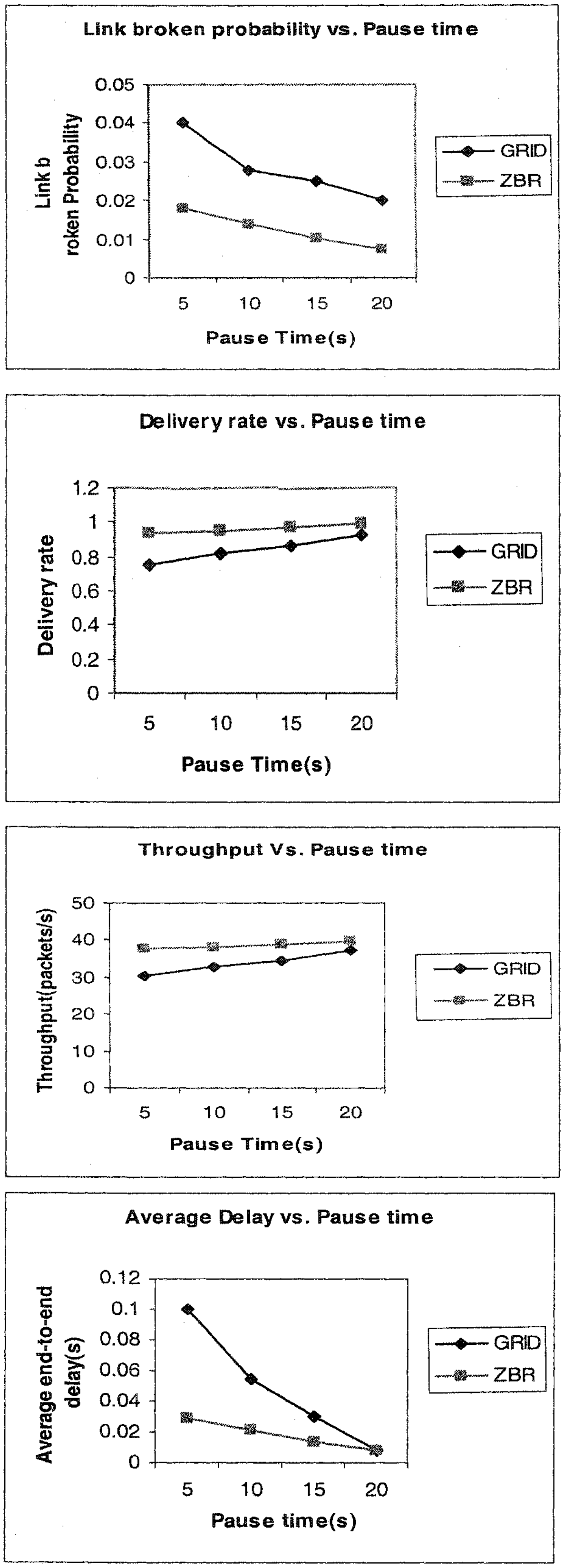
We can see from figure 4.1.E – 4.1.F that ZBR can reach a higher throughput at different arrival rates than GRID and the delay does not increase significantly. We can also see from the figure that in both ZBR and GRID, when the number of nodes in the network is 100, the average end-to-end delay is much less than that when the number of nodes is 50. This is because as the density increases, the probability of a zone to be empty is decreased, thus the link breakage probability is decreased, in turn the probability of rerouting is decreased, which reduces the average end-to-end delay. Besides, since both

ZBR and GRID use only some of the mobile nodes (zoneheads in ZBR and gateways in GRID), the overhead in routing does not increase with the increase of number of nodes as in flat routed MANET.

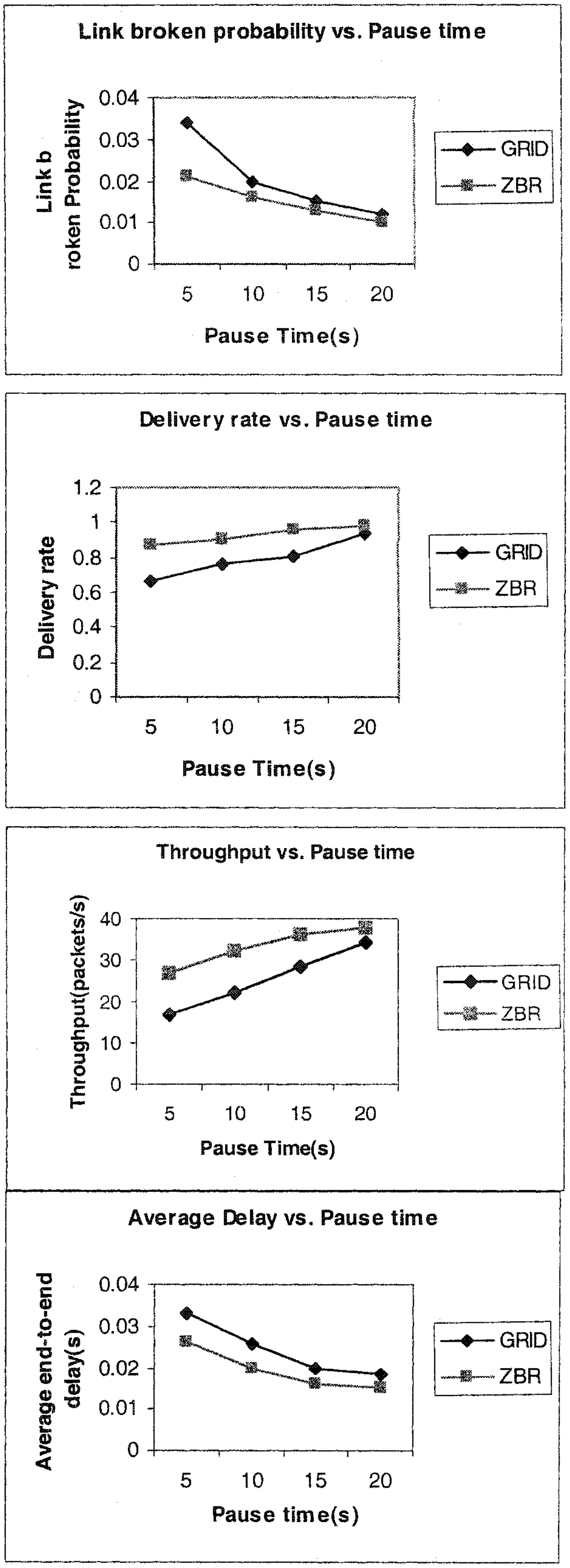




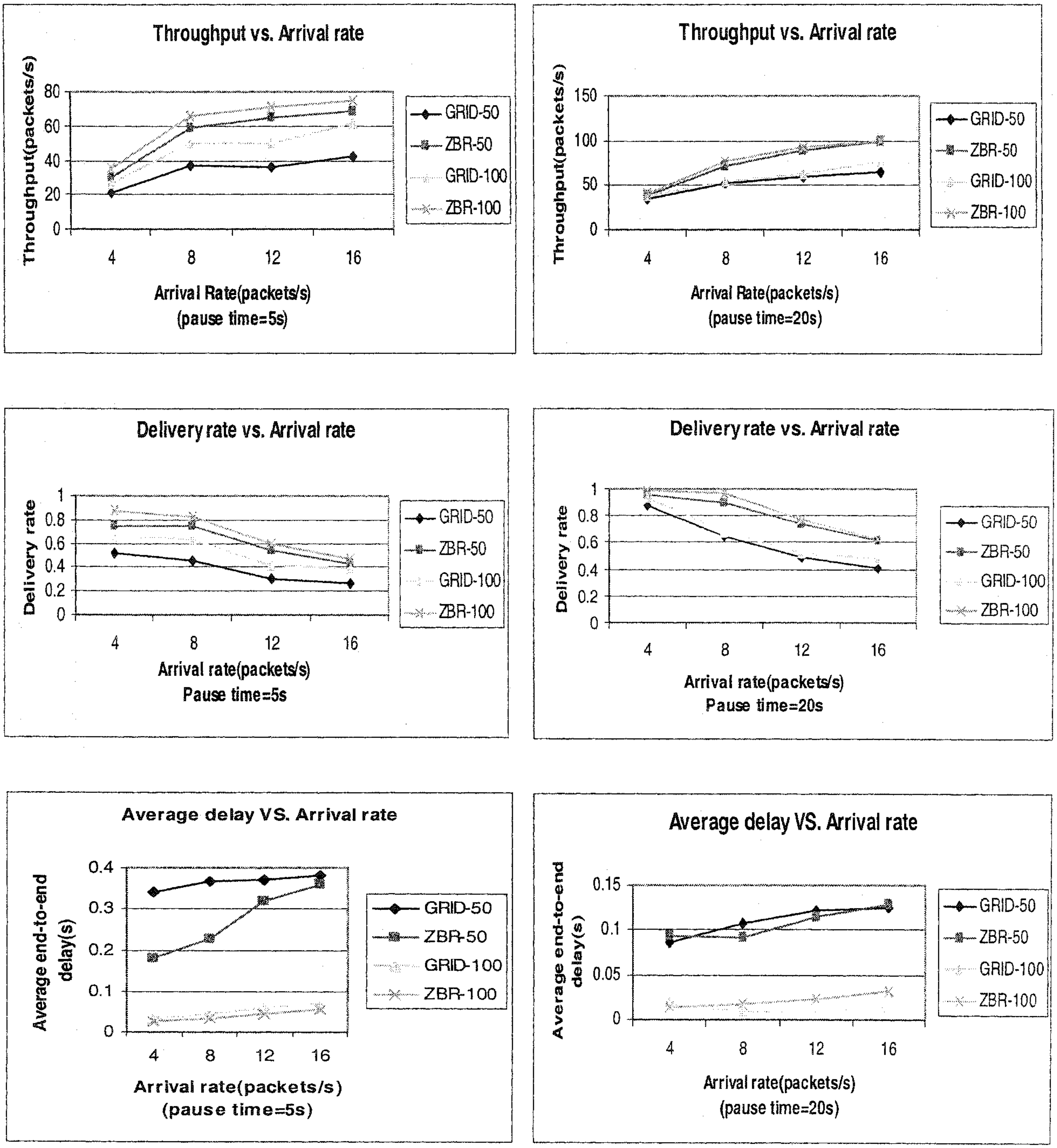
(A) N=50, D=47m, λ =4 packets/s (B) N=50, D=36m , λ =4 packets/s



(C) N=100, D=47m, λ =4 packets/s



(D) N=100, D=36m, λ = 4 packets/s



(E) p=5s, D=36m (F) p=20s, D=36m

Figure 4.1 Comparisons of ZBR & GRID

4.2.2 Discussion of performance of ZBR

In this section we discuss the impact of nodal mobility, the zone size and density and arrival rate on the performance of the network.

4.2.2.1. The effect of mobility

Mobility of a network could be represented by the speed or pause time of each node. We use different pause time to represent different mobility in our simulation The impact of mobility on the performance of network can be observed from Figure 4.1, where we chose 4 different pause times -- 5s, 10s, 15s, 20s.

From Figure 4.1, we can see that as the pause time increases, the probability of link breakage and the average end-to-end delay are decreased, while the delivery rate and throughput are increased. This is because as the pause time increases, the mobility of nodes decreases and the links are more stable, so the probability of link breakage decreases. This reduces the possibility of re-routing, thus reduces the overhead of routing, and consequently, decreases the average end-to-end delay and increases the delivery rate and throughput of the network.

We can also see from Figure 4.1 that ZBR shows better performance than GRID especially in high-mobility networks and its performance is less dependent on the pause time. This is because ZBR considers the mobility of the node while determining the route to forward packets, thus choosing a more stable path reducing the impact of mobility on the network performance.

4.2.2.2 The effect of nodal density

The experiments in this section study the effect of density on the performance of ZBR. We choose the number of nodes in the network to be 50 and 100. In flat-routed ad hoc networks, we know as the number of nodes increases, more nodes could act as routers, more links will exist in the network, hence increasing the complexity of routing. Since more nodes would take part in the route discovery procedure and broadcast the RREQ messages to neighboring nodes, the overhead of routing would become large.

From Figure 4.2, we would like to point out another attractive feature of our protocol -- for both case 1 (smaller size) and case 2 (larger size), the performance of ZBR improves as the number of nodes is increased from 50 to100. As the number of nodes increases, the performance does not degrade as in flat-routed ad hoc networks. This is because we use clustering technique, only the zoneheads act as routers. The number of zoneheads is more dependent on the number of zones than the number of nodes, when the average number of nodes in each zone is at least one (a well-distributed MANET). When the number of the nodes increases, the link breakage probability is decreased. This in turn reduces the overhead of re-routing and increases the delivery rate and throughput. Indeed, this experiment shows the scalability of ZBR, as it adapts to the large-scale MANET.

4.2.2.3. The effect of zone size

The experiments in this section study the effect of zone size on the performance of ZBR. We compare the performance when partitioning the network into smaller zones (zone size=ZoneSize$_1$) or larger zones (zone size=ZoneSize$_2$). We chose the number of MTs in the network to be 50 and 100 (see Figure 4.3).

Figure 4.3a shows that when the number of MTs in the network is 50, ZBR using a smaller zone exhibits better performance in terms of delivery rate, throughput and probability of link breakage. But for the average end-to-end delay, a larger size is better than smaller one. This is because for a same (source, destination) pair, the average length of path (number of hops) is longer when a smaller zone is used. Figure 4.3b shows the performance of ZBR when the number of MTs in the network is 100. We can see ZBR shows good performance in both situation (smaller zone and larger zone). For instance, when the pause time is 5s, the delivery rate can reach 90%. The reason is as the density increases, the route is more stable and the probability of link breakage is decreased. We can also observe that a larger zone shows slightly better performance than smaller zone. This is because in larger zones, nodes are less likely to move out of their zone frequently than in a smaller zone, thus reducing the overhead of information update.

4.2.2.4. The effect of ID vs. IP

In our ZBR protocol, a path is identified by a sequence of zone ID numbers instead of IP addresses that are used in flat MANET routing protocols. The experiments in this section show its effect on the performance of ZBR (see Figure 4.4).

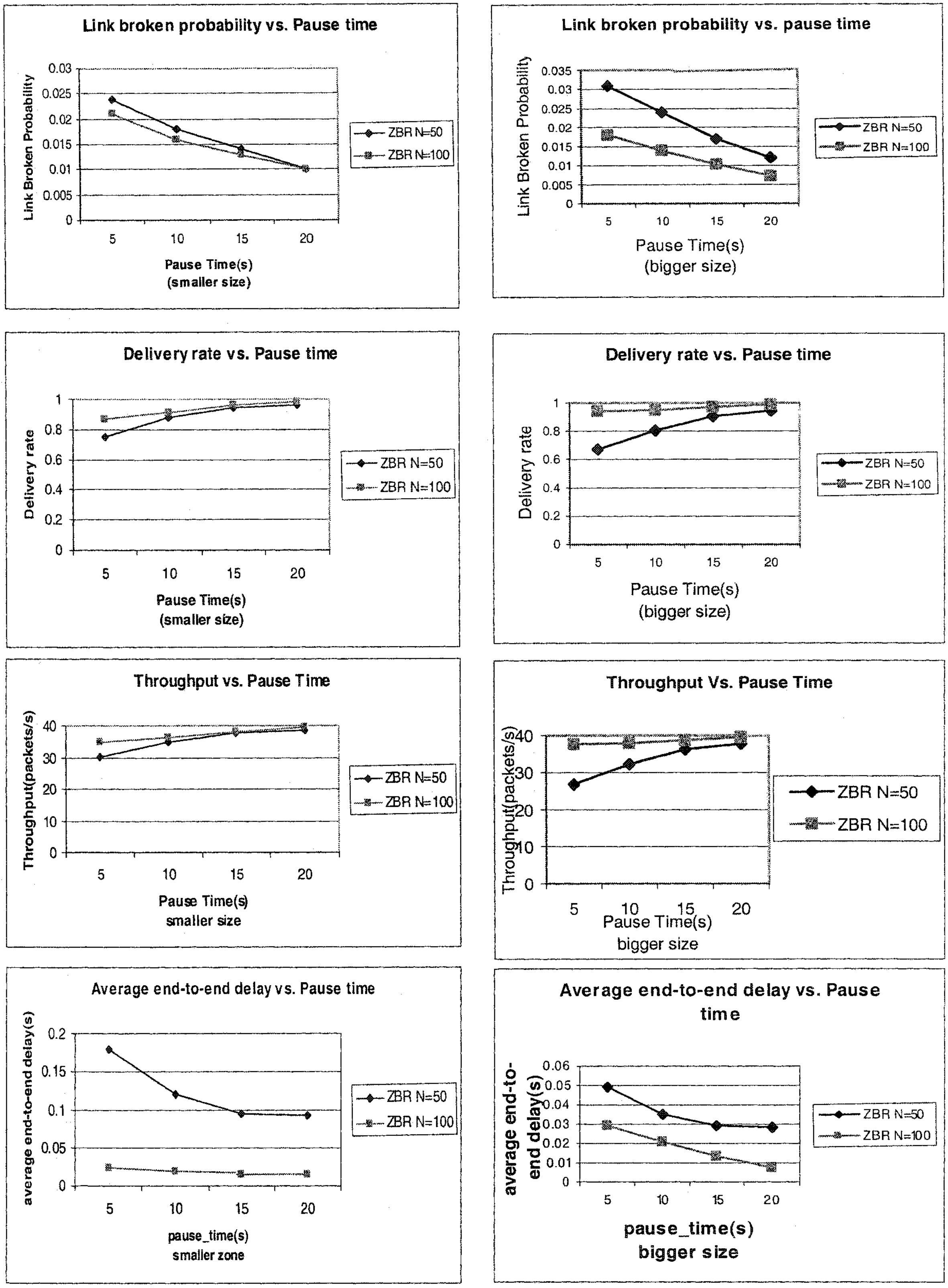
When a path is a collection of zone IDs, the traffic goes through specific zones to reach the destination node. When the zone-head, which acts as the router, moves out of its zone, there will be a new zonehead (if the zone is not empty) and the route will be still available. This increases the lifetime of a route, decreases the link breakage probability, thereby, increasing the throughput and decreasing the average end-to-end delay by decreasing the route cost.

In this experiment, the number of MTs is fixed at 100, Zone-size=ZoneSize$_1$ (smaller zone). We can see from the simulation result that the use of IDs instead of IP numbers reduces the probability of link breakage, thus improves the throughput of the network. The improvement is more significant in high-mobility ad hoc networks.

4.2.2.5 The effect of arrival rate

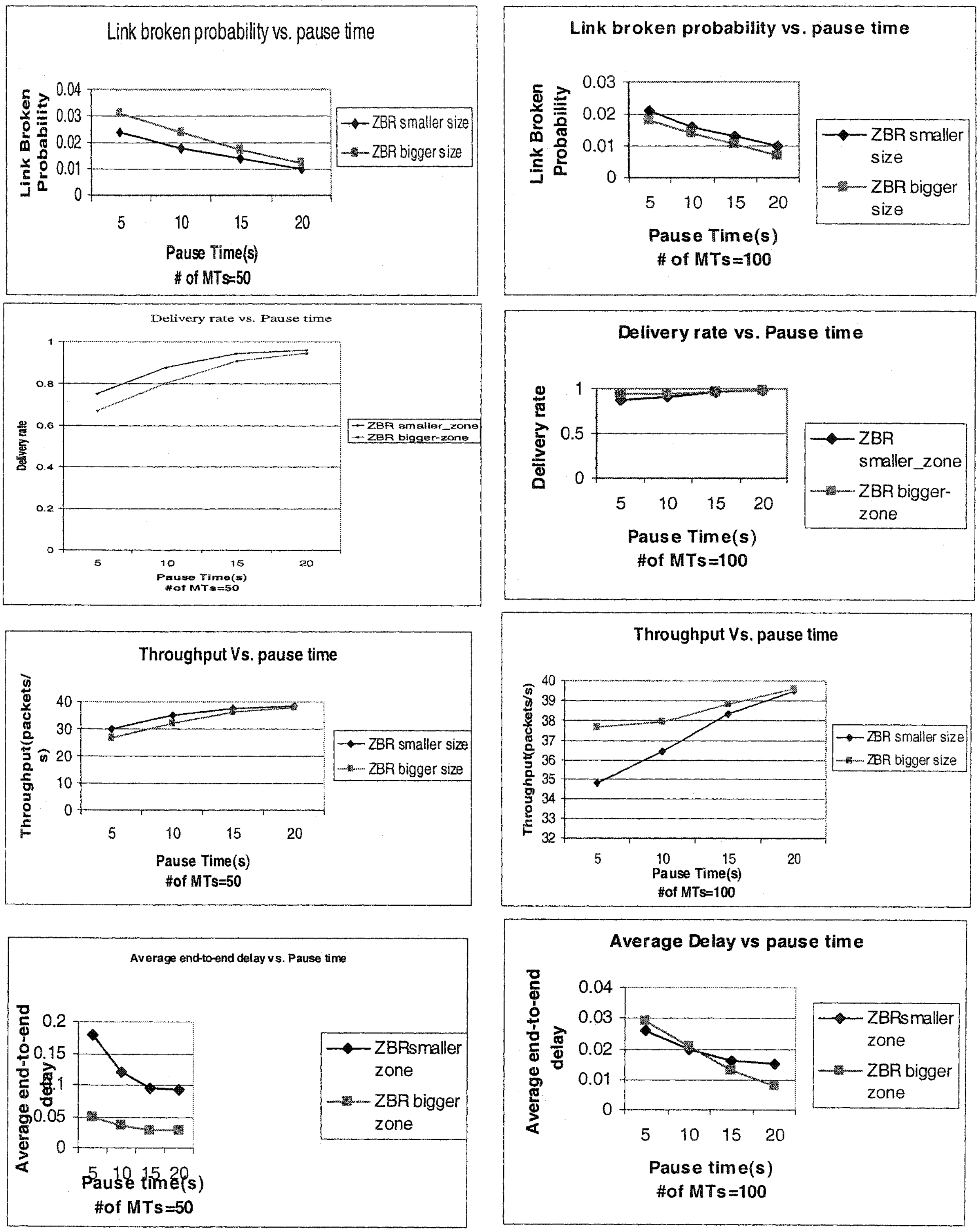
We can see from Figure 4.1.E-4.1.F that as the arrival rate increases, the throughput increases. Of course, the average end-to-end delay increases and the delivery rate decreases as a result of increased contention.

As the arrival rate of packets increases, more packets can go through the network and the throughput of the network is increased. But since the capacity of channel is limited, more packets compete for the limited resources, which increase contentions. As a result we observe larger end-to-end delay, and more packet dropping (lower delivery rate).



(A) D=36cm, λ=4packets/s (B) D=47m, λ =4packets/s

Figure 4.2 Effect of density



(A) N=50, λ =4 packets/s (B) N=100, λ =4packets/s

Figure 4.3 Effect of zone size

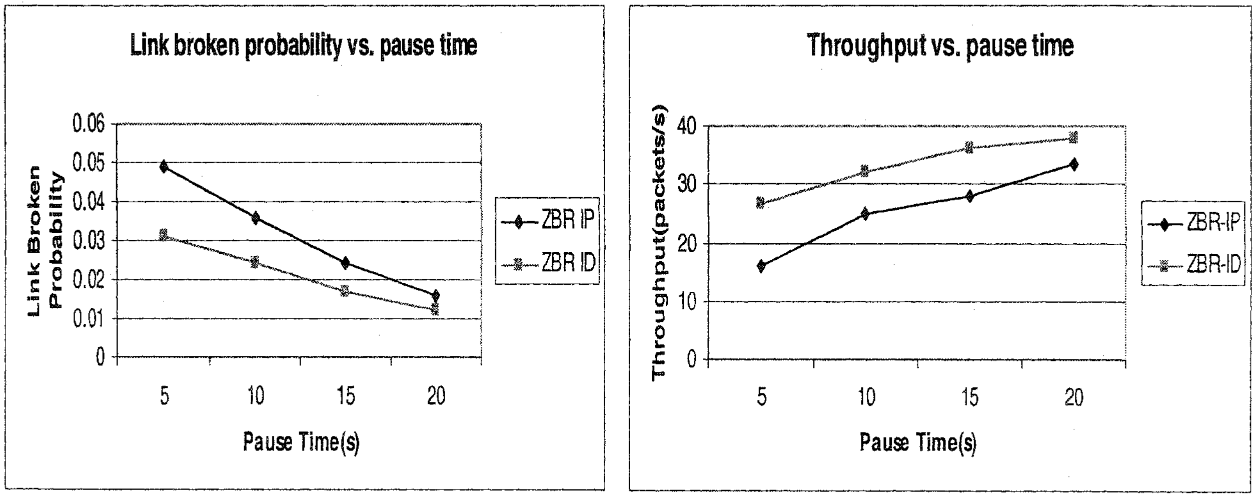


Figure 4.4 The effect of ID instead of IP (N=100, D=36m, λ =4packets/s)

4.3. Summary

In this chapter, we evaluated the performance of ZBR and compared it with GRID through simulation experiments. Simulation results show that ZBR outperforms GRID in terms of link breakage probability, throughput and delivery rates, especially under higher mobility. We also evaluated the impact of mobility, density and number of nodes on the performance of ZBR. It was shown that ZBR has a good performance in large-scale, high-density and high-mobility ad hoc networks.

CHAPTER 5

CONCLUSIONS

5.1 Summary

In this thesis, we proposed a novel zone-based routing protocol for large-scale, high-density and high-mobility wireless ad hoc networks. The main characteristics of ZBR are:

1. We use clustering to adapt to large-scale and high-density ad hoc networks. We use a fixed zone-based partition scheme to partition the network. The algorithm is simple. The overhead in zone maintenance and zone-heads election is low.
2. We use source routing and destination-decided routing strategy. It is not necessary to broadcast HELLO message periodically, and the destination selects the most stable path, and route reply messages are only transmitted along this path. This can reduce the overhead in routing.
3. Self-definition of mobility of nodes by local data. We define the mobility-factor of mobile nodes according to the localized data, which can be collected and estimated at each node, and broadcast it to its zone-head when it joins a new zone. The definition of mobility factor considers both the moving speed and localization of the motion of MTs. The mobility-factor of nodes is used to select a more stable

zone-head and to discover a more stable path, thus leading to lower probability of link breakage, and reaching higher throughput for the network.

4. The definition of the path stability considering both the mobility of mobile nodes and the density of the zones which the path traverses can be used to select more stable routes.
5. We use hybrid routing, where zone maintenance is in a proactive way and route discovery is in a reactive way. This makes ZBR efficient in high-mobility network.
6. The update of nodes' information is both time-driven and event-driven, which means that only when an update-timer expires and a node finds that it has moved to a new zone, it updates and broadcasts its information to its new zone-head. Thus Ping Pong effect can be greatly reduced.
7. Update-timer is flexible, which is a value related to mobility of the nodes and zone size. This can keep the zone maintenance valid with a lower communication overhead.
8. A routing path is a sequence of IDs instead of IP addresses, which can extend the lifetime of a path and reduce the probability of link breakage.

Simulation results show that ZBR has good performance in large-scale, high-density and high-mobility ad hoc networks.

5.2. Future research

We plan to implement the ZBR protocol in a real testbed for mobile ad hoc networks to prove its correctness. By use of fixed partition of the network, ZBR reduces the overhead in cluster formation. However it relies on the accuracy of location information. A cheaper and more accurate positioning system is a necessity to put it into use.

We have shown that a larger zone size can cause sub-optimal routing problem, but a smaller zone may lead to longer paths. So there should be a trade off in zone size. Future research will be done to dynamically change the zone size according to the density of the network. Future research will also include a scheme based on *sub-zones,* which means if the mobile nodes are not well distributed, we partition the high-density area into smaller sub-zones. In this thesis, we assume that the transmission range is fixed. In future protocols, we plan to investigate how to dynamically change the transmission range of mobile nodes according to the zone size, which may result in a higher throughput. In wireless ad hoc networks, battery power is a limited resource. One option is to let the zone-head to maintain the battery power information of each member node, and consider battery power as a criterion in zone-head selection and route determination. Use the square shapes to partition the network can make the numbering and the calculation of zoneID simple. However, it may not reach a high efficiency of power and bandwidth, we will investigate the effect of different shapes and zone sizes on the performance of network in the future. We will also investigate table-driven inter-zone routing protocols and localized inter-zone routing protocols [Y03].

BIBLIOGRAPHY

[BKL01] Prithwish Basu, Naved Khan, and Thomas D.C. Little, "A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks," in Proc. of IEEE ICDCS 2001 Workshop on Wireless Networks and Mobile Computing, Phoenix, AZ, pp. 413-418, April 2001.

[BOT01] B. Bellur, R.Ogier and F. Templin. "Topology broadcast based on reverse-path forwarding (TBRPF)", Internet Draft, draft-ietf-manet-tbrpf-01.txt, work in progress, March 2001

[CHH01] S. Capkun, M. Hamdi and J. Hubaux, "Gps-free positioning in mobile ad-hoc networks", In Proc. Hawaii Int. Conf. on System Sciences, pp. 3481 -3490 January 2001

[CM01] Douglas S. J. De Couto and Robert Morris. "Location proxies and intermediate node forwarding for practical geographic forwarding", Technical Report MIT-LCS-TR-824, MIT Laboratory for Computer Science, June, 2001

[D97] R.Dube et al., "Signal stability based adaptive routing for ad hoc mobile networks", IEEE Pers. Comm., Feb. 1997, pp36-45

[DCY00] S. R. Das, R. Castaneda and J. Yan. "Simulation based performance evaluation of mobile, ad hoc network routing protocols", ACM/Baltzer Mobile Networks and Applications (MONET) Journal, pp. 179-189, July, 2000

[DHY01] Hongyan Du, Hossam Hassainein and Chitsiang Yeh, "Fixed Zone-Based Routing Protocols", IEEE WLAN workshop, Boston, U.S.A. 2001, http://www.wlan01.wpi.edu/proceedings/wlan45d.pdf

[DMJ+98] Josh Broch David A. Maltz, D.B. Johnson, Y.C. Hu, J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols", Proceedings of the fourth annual ACM/IEEE international conference on mobile computing and networking, Oct. 1998, Dellas, Texas, USA pp.85-97

[DPR00] Das, S.R., Perkins, C.E. and Royer, E.M, "Performance comparison of two on-demand routing protocols for ad hoc networks", IEEE INFOCOM 2000. Proceedings of the Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Vol.1 pp. 3 –12

[DRT+00] Alan D. Amis Ravi Prakash Thai H.P. Vuong Dung T. Huynh "Max-Min D-Cluster Formation in Wireless Ad Hoc Networks", pp. 32 - 41 vol.1 Infocom 2000

[EFB84] A. Ephremides, J.A. Flynn and D.J. Baker, "The design and simulation of a mobile radio network with distributed control", IEEE J. of Selected Areas in Communication, 2, 1, 1984, pp. 226-237

[GWF+02] Michael Gerharz, Christian de waal, Matthias Frank, Peter Martini, "Link Stability in Mobile Wireless Ad Hoc Networks", Proceedings of the 27th Annual IEEE Conference on Local Computer Networks(LCN02), 2002, pp: 30- 39

[HB01] Jeffrey Hightower and Gaetano Borriello. "Location systems for ubiquitous computing. Computer", 34(8):57-66, August 2001

[HH94] A. Harter and A. Hopper. "A Distributed Location System for the Active Office", IEEE Network, 8(1), pp. 62 - 70 Jan. 1994

[HL86] T.-C. Hou and V. O.K. Li. "Transmission range control in multihop packet radio networks". IEEE Trans. on Communications, 34(1):38-44, January 1986

[HP01] Z. Haas and M. Pearlman. "The performance of query control schemes for the zone routing protocol". ACM/IEEE Transactions on Networking, 9(4):427-438, August 2001

[HP97] Haas Z.J. and M.R. Pearlman, *"The Zone Routing Protocol(ZRP) for Ad hoc Networks"*, Internet draft - Mobile Ad hoc NETworking (MANET) Working Group of the Internet Engineering Task Force (IETF), November 1997

[HT01] T-C.Hou, T-J.Tsai, "An Access-Based Clustering Protocol for Multihop Wireless Ad Hoc Networks", IEEE Journal on Selected Areas in Comm. Vol.19, No.7, pp.1201-1210, July 2001

[HZ01] H.S. Hassanein and A. Zhou, "Routing with Load Balancing in Wireless Ad hoc networks," ACM Workshop on Modeling, Analysis and simulation of Wireless and Mobile Systems, July 2001.

[ICP+99] A. Iwata, C.-C. Chiang, G.Pei, M. Gerla, and T.-W. Chen, " Scalable Routing Strategies for Ad Hoc Networks", IEEE Journal on Selected Area in Communications, Special Issue on Ad Hoc Networks, pp. 1369-79, Aug. 1999

[JL99] M Joa-Ng and I.T.Lu, "A Peer-toPeer zone-based two-level link state routing for mobile Ad Hoc networks", IEEE Journal on Selected Areas in Communications, Special Issue on Ad_hoc Networks, Aug.1999, pp.1415-1425.

[JLH+99] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek and M. Degermark, "Scenario-based performance Analysis of Routing Protocols for Mobile Ad Hoc Networks," Proceedings of ACM Mobicom 1999, Seattle WA, August 1999. pp. 195-206

[JLT98] M. Jiang, J. Li, Y. C. Tay, "Cluster Based Routing Protocol (CBRP) Functional Specification," draft-ietf-manetcbrp-spec-00.txt, August 1998

[JM96] D. Johnson and D. Maltz. "Mobile Computing, chapter 5 - Dynamic Source Routing", pp. 153-181. Kluwer Academic Publishers, 1996

[JMQ+01] P. Jacquet, P. Muhlethaler, A. Qayyum, A. Laouiti, L. Viennot, and T. Clausen. "Optimized Link State Routing Protocol". Internet Draft, draft-ietf-manet-olsr-04.txt, work in progress, September 2001

[K00] B. N. Karp. "Geographic Routing for Wireless Networks". PhD thesis, Harvard University, 2000

[K96] E. D. Kaplan. "Understanding GPS: Principles and Applications". Artech House, Boston, MA, 1996

[K99] A. Krikelis. "Location-Dependent Multimedia Computing", IEEE Concurrency, 7(2): 13-15, pp. 13 -15 April - June, 1999

[KK77] I. Kleinrock and F. Kamoun, "Hierarchical routing for large networks", Computer Networks, 1, 1977, pp155-174.

[KK00] B. Karp and H. T. Kung. "Greedy perimeter stateless routing for wireless networks". In Proc. of the 6th Annual ACM/IEEE Int. Conf. on Mobile Computing and Networking (MobiCom 2000), pp. 243-254, Boston, MA, USA, August 2000

[KSU99] E. Kranakis, H. Singh, and J. Urrutia. "Compass routing on geometric networks". In Proc. of 11th Canadian Conf. on Computational Geometry, Vancouver, August 1999, pp. 51-54

[KV00] Y.-B. Ko and N. H. Vaidya. "Location-Aided Routing (LAR) in mobile ad hoc networks". ACM/Baltzer Wireless Networks (WINET) journal, 6(4), pp. 307-321, 2000

[L95] G. Lauer, Packet-radio routing, in: "Routing in Communication Networks", ed. M. Steenstrup, Prentice-hall, 1995

[LJC+00] J. Li, J. Jannotti, D. S. J. De Couto, D. R. Karger, and R. Morris. "A scalable location service for geographic ad hoc routing". In Proc. of the 6th Annual ACM/IEEE Int. Conf. on Mobile Computing and Networking (MO-BICOM) 2000, pp. 120-130, Boston, MA, USA, 2000

[LSG01] Sung-Ju Lee, William Su, Mario Gerla , "Wireless ad hoc multicast routing with mobility prediction" Mobile Networks and Applications, Volume 6 , Issue 4,August 2001 pp. 351 - 360

[LTS01] Wen-Hwa Liao, Jang-Ping Sheu, Yu-Chee Tseng: GRID: "A Fully Location-Aware Routing Protocol for Mobile Ad Hoc Networks". Telecommunication Systems 18(1-3), pp. 37-60 , 2001

[MJH01] Martin Mauve, Jorg Widmer, and Hannes Hartenstein: "A Survey on Position-Based Routing in Mobile Ad-Hoc Networks", IEEE Network, Vol. 15 (2001), Nr. 6, S. pp. 30-39

[MJK+00] R. Morris, J. Jannotti, F. Kaashoek, J. Li, and D. S.J. De Couto. Carnet: "A scalable ad hoc wireless network system". In Proc. of the 9th ACM SIGOPS European workshop: Beyond the PC: New Challenges for the Operating System, Kolding, Denmark, September, 2000

[MZ01] A.B.McDonald, T.F.Znati, "Design and Performance of a distributed dynamic clustering algorithm for ad hoc networks", in Proc. Of 34th Annual Simulation Symposium pp.27-35, 2001

[NI97] J.C. Navas and T. Imielinski. "Geographic addressing and routing" In Proc. 3rd ACM/IEEE Int. Conf. on Mobile Computing and Networking (MobiCom'97), Sept. 1997, http://athos.rutgers.edu/~navas/dataman/talks/Chicago_CNRI

[NK84] R. Nelson and L. Kleinrock. "The spatial capacity of a slotted aloha multihop packet radio network with capture", IEEE Trans. on Comm., 32(6), pp. 684-694, June 1984

[NTC+99] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu. "The Broadcast Storm Problem in Mobile Ad Hoc Network", In Proc. Of the Fifth ACM/IEEE International Conference on Mobile Computing and Networking(MobiCom'99), August, 1999. pp. 151-162,

[PB94] C. Perkins and P. Bhagwat. "Highly dynamic destination sequenced distance vector routing (DSDV) for mobile computers", Computer Communication Review, pp. 234-244, October 1994.

[PC97] V. Park and M. Corson. "A highly adaptive distributed routing algorithm for mobile wireless networks", In Proc. of INFOCOM'97, pp. 1405 -1413 Vol.31997

[PH99] M.R. Pearlman and Z.J. Haas, "Determining the Optimal Configuration for the Zone Routing Protocol", *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 8, pp. 1395-1414.August 1999

[PR99] C. Perkins and E. Royer. "Ad-hoc on-demand distance vector routing" In Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, pp. 90-100, February 1999

[RS96] S. Ramanathan and Martha Steenstrup, "A survey of routing techniques for mobile communications networks", Mobile Networks and Applications 1(2), pp. 89-104, 1996

[RT99] E. Royer and C.-K. Toh, "A review of current routing protocols for ad hoc wireless networks", IEEE Personal Communications, pp. 46-55, April 1999

[S84] N. Shacham, "Organization of dynamic radio networks by overlapping clusters: Architecture, considerations, and optimization, Performance", Performance'84, Proceedings of the Tenth International Symposium on Computer Performance Modelling, Measurement and Evaluation, Paris, December, 1984. North-Holland 1985, ISBN 0-444-87680-4, pp. 435-447

[S00] Ahmed Safwat, "The Virtue Base Stations protocol", Master's thesis, Queen's University, 2000

[SLG01] W. Su, S.-J. Lee, and M. Gerla, " Mobility Prediction and Routing in Ad Hoc Wireless Networks", International Journal of Network Management, vol. 11, no. 1, John Wiley & Sons, January/February 2001, pp. 3-30.

[T80] G. Toussaint. "The relative neighbourhood graph of a finite planar set", Pattern Recognition, 12(4), pp. 261-268, 1980

[T99] Chai-Keong Toh, "Long-lived ad-hoc routing based on the concept of associativity", March 1999 IETF Draft

[TA96] Tanenbaum, Andrew S., *Computer Networks*, Prentice Hall Inc, USA, 1996

[TK84] H. Takagi and L. Kleinrock, "Optimal transmission ranges for randomly distributed packet radio terminals" IEEE Trans. on Communications, 32(3), pp. 246-257, March 1984

[TRT+89] W.T. Tsai, C.V. Ramamoorthy, W.K. Tsai and O. Nishiguchi, "An adaptive hierarchical routing protocol", IEEE Trans. Commun. 38, 8, pp. 1059-1075,1989

[WHF+92] R. Want, A. Hopper, V. Falcao, and J. Gibbons. "The Active Badge Location System" ACM Trans. on Information Systems, 10(1), pp. 91-102, January, 1992

[Y03] C.-H. Yeh, "Hybrid and Hierarchical Routing Schemes for Mobile Ad Hoc Networks, Sensor Networks, and 4G Wireless Systems," Dept. Elec. & Comput.Eng., Queen's Univ., Canada, Tech. Rep. no. 0303, 2003.

[YHM02] T. You, H.S. Hassanein and H.T. Mouftah, "Infrastructure-Based MAC for Wireless Ad Hoc Networks," IEEE LCN Workshop on Wireless Local Networks, Nov. 2002.

[ZWM+02] Xiaofeng Zhong, Youzheng Wang, Shunliang Mei, Jing Wang, "A Stable Adaptive Optimization for DSR Protocol" CDMA International Conference 2002, pp: 247-253

APPENDIX

CONFIDENCE INTERVALS

Normally, confidence intervals placed on the mean values of simulation results can be used to describe the accuracy of the simulation results. Consider the results of N statistically independent simulation runs for the same experiment: X_1, X_2, …, X_N. The sample mean, $\overline{X}$ is given as:

$$\overline{X} = \frac{\sum_{i=1}^{N} X_i}{N}$$

The variance of the distribution of the sample values, S_x^2 is:

$$S_x^2 = \frac{\sum_{i=1}^{N} (X_i - \overline{X})^2}{N-1}$$

The standard derivation of the sample mean is given by: $\frac{S_x}{\sqrt{N}}$.

Under the assumption of independence and normality, the sample mean is distributed in accordance to the T-Distribution, which means the sample mean of the simulation runs fall in the interval $\pm\varepsilon$ within the actual mean with a certain probability drawn from the T-Distribution.

$$\varepsilon = \frac{S_x t_{\alpha/2,N-1}}{\sqrt{N}}$$

where $t_{\alpha/2,N-1}$ is the value of the T-distribution with N-1 degrees of freedom with probability $\alpha/2$.

The upper and lower limits of the confidence interval regarding the simulation results are:

$$\text{Lower Limit} = \overline{X} - \frac{S_x t_{\alpha/2,N-1}}{\sqrt{N}}$$

$$\text{Upper Limit} = \overline{X} + \frac{S_x t_{\alpha/2,N-1}}{\sqrt{N}}$$