Department of Computer Science & Engineering

Thesis On

Comparison of Different Protocols
&
Strategies for Range Queries
In
Mobile Ad hoc Networks

Kazi Anisur Rahman (ID#02201015)
Fuad Hussain (ID#02201006)
Nilotpal Das (ID#02201037)

Supervised By

Sadia Hamid Kazi

BRAC University, Dhaka, Bangladesh

Declaration

In accordance with the requirements of the degree of Bachelor of Science (B.Sc.) in Computer Science & Engineering program at BRAC University, we present the following thesis entitled “Comparison Of Different Protocols & Strategies For Range Queries in Mobile Ad Hoc Networking”. This work was performed under the supervision of Sadia Hamid kazi.

We hereby declare that the work submitted in this thesis is our own and based on the results found by ourselves. Materials of work found by other researcher are mentioned by reference. This thesis, neither in whole nor in part, has been previously submitted for any degree.

Signature of
Supervisor

Sadia Hamid Kazi
Lecturer,
CSE Department,
BRAC University.

Signature of
Author

Nilotpal Das
ID # 02201037

Fuad Hussain
ID # 02201009

Kazi Anisur Rahman
ID # 02201015

Acknowledgement

The thesis work has done for the fulfillment of Bachelor of Science (B.Sc.) in Computer Science & Engineering program at BRAC University.

At first our heartiest gratitude goes to Almighty Allah, without his divine blessing it would not be possible for us to complete this project successfully.

We must also pay gratitude to our supervisor Sadia Hamid Kazi, faculty of BRAC University who gave us the chance to perform our thesis under her supervision. For her enriched and powerful structured discussion that has been a great help in each step of doing and writing the dissertation.

We would like to thank all of our respected teachers for their valuable advices and kind cooperation and continuous encouragements.

We like to offer thanks to all of our friends and well–wishers for helping us by rigorous reviews of this work and inspiring suggestion.

Abstract

One area, which has already been identified as a focal point of research in MANET is routing with a number of routing protocols proposed in the last couple of years. However, a comparison between them is lacking to help deciding which routing protocol is best suited in specific network scenarios. The paper is presented the performance comparison of the protocols with identical loads and environmental conditions and summarized their relative performance in a tabular form with respect to some basic performance metrics. Moreover, this report focused on a clear view of range query applied in mobile ad hoc networks, different strategies to realize this range query and analysis of the results obtained from the implementation of the best suited strategy into ns2. The strategies to realize range queries are based on the criteria to collect location and other information of different nodes situated in an expected region in an ad hoc network. Several forwarding algorithms have been studied and measured for the accumulation of the suitable ones to realize range queries. Based on the theoretical concept of the different protocols and the strategy of the range query a proposed idea was given that can be used for efficient communication among the nodes.

Appendix

List of Acronyms with abbreviations

AODV	Ad-hoc On-demand Distance Vector
CSCW	Computer-Supported Collaborative Work
CBR	Constant Bit Rate
DAG	Directed Acyclic Graph
DARPA	Defense Advanced Research Projects Agency
DBF	Distributed Bellman-Ford
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
ER	Expected Region
GPSR	Greedy Perimeter Stateless Routing Protocol
GPS	Global Position System
GUI	Graphical User Interface
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
LT	Location table
LAN	Local Area Network
LAR	Location Aided Routing
MAC	Medium Access Control
MANET	Mobile Ad-hoc Network
MN	Mobile Node
NS-2	Network Simulator 2
PAN	Personal Area Network
PDA	Personal Digital Assistant
RQ	Range Query
RREQ	Route Request
RREP	Route Reply
RERR	Route Error
TORA	Temporally Ordered Routing Algorithm

List of Figures

List of Tables

Table of Contents:

1. Introduction

The rapid technological advances and innovations of the past few decades have pushed wireless communication from concept to reality. There are currently two variations of mobile wireless networks, infrastructure and infrastructure less networks. Typical infrastructure networks are cellular mobile networks, which have fixed base stations, which are connected with other base stations through a wired backbone. The transmission range of a base station covers a cell. All the mobile nodes lying inside this cell connect to and communicate with the nearest base station. A "handoff" occurs when a mobile host travels out of range of one base station and into the range of another base station (change of cells).

The other type of network, infrastructure less network, is known as ad hoc network. These networks do not rely on an infrastructure and can operate without any base station or access point and without a backbone network. In mobile ad hoc networks, so called MANET, all nodes are capable of movement and can be connected dynamically in an arbitrary manner.

MANET is such kind of networking where all the nodes are treated independently and each work as a router. This is absolutely different than that of usual networking system. Here each node contributes to update all the possible routing information that can be reached. However how this should be done is the process of different protocols. In some protocols it just broadcast to the nearby neighbor whenever there is a change occurs. In other cases it is broadcast only when it is necessary.

Each of these may be suitable in different situations. Some may be optimized with bandwidth utilizing while others creates a traffic jam. Some utilizes the power of the nodes by sending less information occasionally whereas other broadcast the thing frequently but without any congestion.

All the packet formats are absolutely different from the other ones. All packets formats are different. So if one finds the different protocol packet it just drops. Therefore to communicate between two protocols, one’s information has to be injected into the desired one.

2. Features of Mobile Ad-hoc Networks

2.1 Dynamic Topology: The topology of mobile systems can change very rapidly. Therefore one will find that communication end-points frequently move independently of one another.

2.2 Self-organizing: Every time a mobile host moves, it needs to re-discover which mobile hosts are reachable. It does this by sending a “ping” message in all directions and listens for corresponding “pong” messages. The strength of the “ping” message weakens as distance increases giving the mobile host a limited range within which “ping” messages can be “heard”. This range is called the *scan range* of the mobile host.

2.3 Fully decentralized: No central server exists. Therefore every mobile host is equally important within the network.

2.4 Thin Clients: The host/node has very limited CPU capacity, storage capacity, and battery power. Limited power usage leads to limited transmitter range.

2.5 Low cost: Wireless ad hoc networks are built from low-cost transceivers and do not incur charges for provider access and airtime.

3. Standards of MANET Routing Protocols

The MANET working group [5] of the IETF has detailed a list of desirable qualitative properties of ad-hoc routing protocols [2]. It is necessary that a newly proposed protocol meet some if not all of the standards in the list summarized below

3.1 Distributed Operation

Route computations in the network must be done in a distributed manner as the centralized approach is inappropriate for the dynamic ad-hoc network. Centralized routing would create critical nodes in the ad-hoc network which is a highly undesirable scenario in MANETs.

3.2 Loop-freedom

Although not critical, a routing protocol should determine loop-free routes from source to destination. This prevents the wastage of resource due to a fraction of the packets looping in the network for an arbitrary amount of time.

3.3 Demand based operation

In order to make efficient use of available resources, a routing protocol should be adaptive to the network traffic produced. This means that the protocol should only react when it is required for it to do so and avoid periodic exchange of routing information. There is a possibility with protocols that use periodic exchanges that they maintain routes that are never used. The clear drawback with the on demand approach is the increase in data packet delivery latency due to the route discovery process.

3.4 Proactive operation

In certain applications of ad-hoc networks the packet delivery latency due to the on demand based operation is unacceptable. In such scenarios the use of additional resources should be traded off for lower delays protocols that employ a proactive operation.

3.5 Unidirectional Link Support

The precarious radio environment can result in the formation of unidirectional links in the network. It is a desirable feature in a designed routing protocol that it detects and adapts to such types of links.

3.6 Power conservation

The mobile nodes in an ad-hoc network rely on limited battery power for their operation. Thus it is necessary that a routing protocol be conservative in its use of such resources. In addition, in order to conserve energy, a node may stop transmitting and/or receiving for arbitrary time periods. The routing protocol should support the “sleep” mode functionality in its operation.

3.7 Security

A node in a MANET is susceptible to security attacks in the form of snooping of network traffic, replaying transmissions, redirecting routing messages and manipulation of packet headers. These are all actions that can be easily carried out by a malicious node in the open radio environment. Thus it is necessary that a protocol provide a degree preventive security.

3.8 Multiple routes

A protocol that creates multiple routes between source and destination pairs could theoretically increase the traffic carried by the network. This would also decrease the number of reactions to topology changes and congestion in the network. The availability of alternative routes would make it unnecessary for the routing protocol to re-discover routes that have been broken.

3.9 Quality of service

In order to carry multimedia data traffic in an ad-hoc network it is essential that a routing protocol support some sort of Quality of Service. This becomes pertinent when time critical data such as voice is considered.

It should be noted that no ad-hoc routing protocol proposed thus far satisfies all the desirable requirements detailed above. Each protocol tries to solve a certain sub-set of the problem-set often trading off one requirement for another depending the characteristic of the ad-hoc network for which the protocol is being designed.

4. Application of Ad hoc Routing Protocol:

4.1 Emergency Crisis management

In emergency situations such as natural disasters or accident scenarios communications, using ad-hoc networks would be invaluable. In the event of a natural disaster it is likely that if there was any existing communications architecture in the area, it would be in disarray. On the other hand, a self-organizing dynamic ad-hoc network would restore communications easily and at a moments notice in such a time critical situation. Search and rescue systems in environments where there is no pre-existing communication architecture ad-hoc network provide an ideal, efficient solution.

4.2 spontaneous conferencing

There exist scenarios where collaborative computing becomes a highly useful facility if not essential. In conference venues, ad-hoc networks could provide the functionality for audio and video conferencing. Researchers equipped with mobile computers in the field could use MANETs to share information and communicate. In some business environments, ad-hoc networks would allow users to conduct interactive meetings in environments other than the office. This type of Computer-Supported Collaborative Work (CSCW) is beneficial in improving workflow and enhancing the productivity of collaborative design work.

4.3 Vehicular communications

With Global Positioning System (GPS) information being offered in automobiles today, it is not far fetched to imagine communications systems that would enhance such location based services. As the automobiles travel through an area, a communication system would collect information on weather conditions, possible road accidents and other road information such as locations of petrol stations, garages, hotels and tourist spots. If there was an architecture that allows vehicles to communicate with each other, the system running in the vehicles would be able to share such information. A MANET is ideal for this type of application.

4.4 Personal Area Networks (PAN)

Today wearable computing is very much a reality with the size of chips diminishing with each version revision. In the foreseeable future there will exist wireless nodes on a person in terms of a watch, pen or belt. It will be possible to store information in such devices, which may be shared when users with such devices meet in an ad-hoc fashion. With the advent of the intelligent home and office environment, the idea is to network mobile wireless nodes such as PDAs with other devices of the environment enabled with wireless interfaces. Thus a person entering his home with his PDA will automatically alert the alarm system to deactivate, instruct the temperature controlling system to set the appropriate temperature, activate his entertainment system etc. In the office environment a person's PDA will be able to interact with his desktop system and share necessary information such as memos, emails and data files in an ad-hoc manner without any wired connections.

4.5 Tactical networks

The development of ad-hoc networks, from its incubation stages in the 1970 with DARPA in the US, has been done with military applications in mind. In the battlefield scenario the communications between personnel has to be robust and reliable. Structured communication architectures cannot be relied upon in such an environment, as such structures are vulnerable to physical attacks by an enemy. The self-organizing nature of an ad-hoc network makes it ideal for the treacherous scenario. With such a network, the soldiers would be able to move forward to uncharted territory being able to communicate within their own battalion, with other such battalions and with their command and control centre. This form of situational awareness is vital in organizing operations in a military effort and the ability to instantaneously establish field communications is invaluable. MANETs are by nature very robust and adaptive. The existence of the network is not hampered if one or a few of the nodes in the network disappear either due to damage or mobility. The network changes to adapt to the new topology in order to maintain connectivity. In addition, the multi-hopping functionality enables the network to cover a far wider physical region in the battle field environment than previously possible.

4.6 Sensor networks

Wireless sensor networks are a well-known application of ad-hoc networks. They are used for monitoring and analysis of uncharted generally inhospitable terrain. They are deployed in places that are not easily accessed by humans. Sensor networks consist of many possibly disposable, low power devices equipped with programmable computing. In a military scenario, sensor networks are deployed in enemy territory where they gather information on enemy activities and relay such information back to the central control. However, sensor network applications are not limited to the military as they have found applications in environmental monitoring systems, security surveillance and distributed computing. A futuristic use of sensor networks was imagined in the motion picture "Minority Report" as sensory robots. With the development of artificial intelligence technology, ad-hoc communications can be the only feasible way for such systems. The above mentioned applications are merely a subset of the possibilities of ad-hoc networks. In the near future MANETs will be deployed in a variety of applications and environments.

5. Routing in MANET

Routing in infrastructure-based wireless and wired networks involves centralized routing, which computes optimal routes to a given destination using cost metrics like hop count or bandwidth. While this works well for wired networks, it does not scale well to dynamically changing environments such as a mobile ad-hoc network (MANET). In ad hoc environments, mobile hosts, which want to communicate with one another, are required to form a self-organized network, which requires each node to perform multi hop routing. The dynamic nature of ad-hoc networks, combined with topology changes caused by signal fading, interference loses, unidirectional links and link connectivity changes, needs more effective routing protocols to address the need for faster convergence and low control overhead in mobile scenarios.

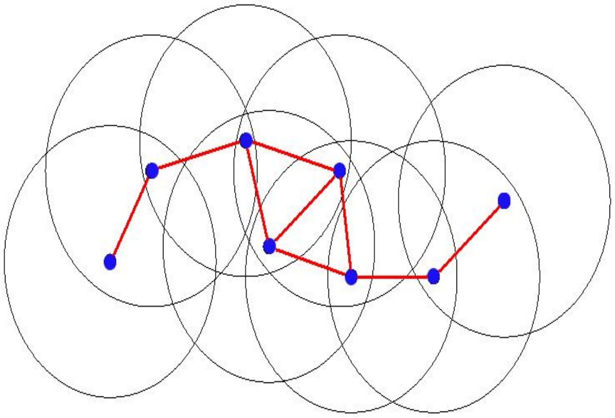

Figure 5(a): Mobile wireless hosts and edges that transfer from one node to another

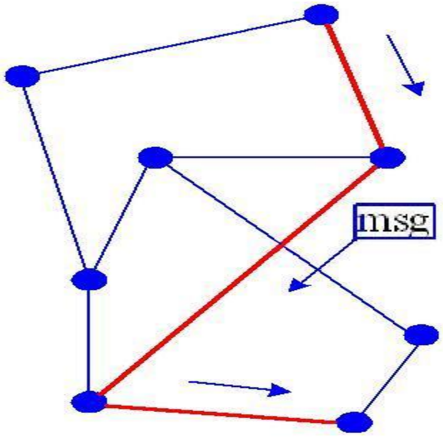

Figure 5(b): Network with nodes and edges that transfer message from one node to another

6. Categories of MANET Routing Protocols

There are actually three main categories of MANET routing protocols. They can be viewed as follows -

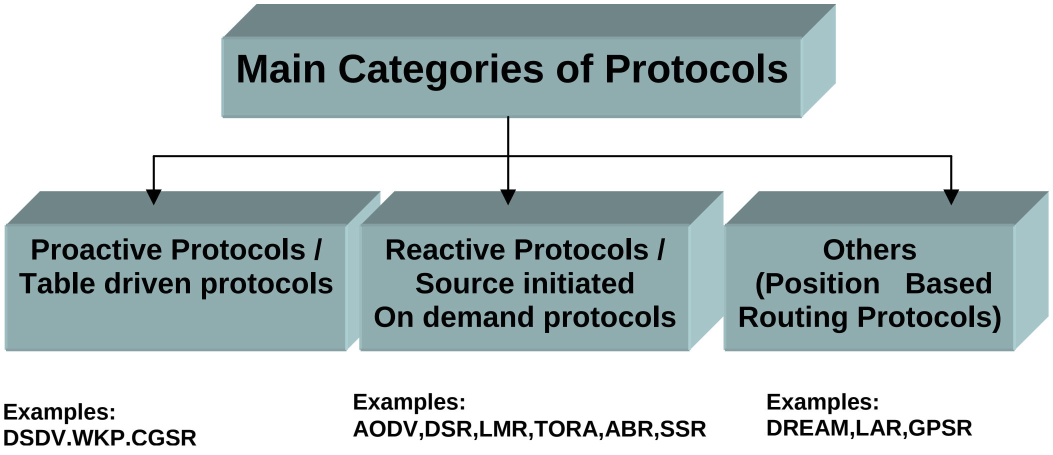
Main Categories of Protocols
Proactive Protocols / Table driven protocols
Reactive Protocols / Source initiated On demand protocols
Others (Position Based Routing Protocols)
Examples: DSDV.WKP.CGSR
Examples: AODV,DSR,LMR,TORA,ABR,SSR
Examples: DREAM,LAR,GPSR

Figure 6: Categories of Ad hoc Routing Protocols

6.1 Proactive

Proactive protocols are, in general, derived from the distance vector and link-state schemes of the wired network protocols**.** Proactive routing protocols use periodic broadcasts to establish routes and maintain them. Since they exchange topology information enabling each node to maintain an up-to-date view of the network, proactive protocols are also called **table-driven protocols**. They try to maintain complete routes from each source in the network to all other nodes. This information is generally cached in tabular form with one or more tables being used by the different protocols. In order to maintain a consistent view of the network at each node, the protocols continuously propagate updates of topological changes throughout the network [3]. Example is Destination-Sequenced Distance Vector routing protocol (DSDV).

Proactive protocols can effectively route packets immediately to any other node in the network and do not suffer from a high starting latency. They have been adapted and modified to solve the problems that the static network protocols faced in the dynamic mobile ad-hoc environment. [3]

However, the periodic topology exchange results in a larger overhead especially when node mobility is high. Pro-active protocols, in order to maintain the constantly changing network graph due to new, moving or failing nodes, require continuous updates, which may consume large amounts of bandwidth. Even worse so, much of the accumulated routing information is never used, since routes may exist only for very limited periods of time.

6.1.1 DSDV (Destination Sequenced Distance Vector)

The Destination Sequenced Distance Vector (DSDV) [11] routing algorithm is the modification of the classic Distributed Bellman-Ford (DBF) algorithm. In a MANET any node in the network may be required to act as router and so each node maintains a routing table that lists all the nodes in the network of which it is aware. Each entry in the table contains the destination and the next hop addresses as well as the cost (in terms of hops) to get to the destination. The reason DSDV is an improvement on the original wired network protocol is that it avoids DBF’s tendency to create routing loops. Each entry in the routing table and a protocol message update is marked with a sequence number. This number is maintained by the destination node of a route entry and is increased whenever the node publishes its routing information. The sequence number value is used by all other nodes in the network to determine the “freshness” of the information contained in a route update for the destination. Since the value is sequentially incremented, a higher sequence number implies that the routing information is newer.

In order to maintain routing information consistency in the network each router shares its routing table with its neighbors by means of routing updates. These updates are done both in a periodic and triggered fashion. The designers of the protocol proposed this method with the aim of alleviating the potentially large amount of network traffic that will be induced by the routing updates. In a periodic update which occurs at predetermined regular intervals, a node broadcasts its entire routing table in a packet termed a full dump. Incremental routing update packets are used when triggered significant topological change. The change could be either due to node mobility or link breakages to next hop neighbors. The incremental update packets only contain those entries which have changed since the last periodic update. The triggered updates with the smaller packet sizes result in the reduced overhead incurred by the protocol. A route table update entry contains the destination address of a node, the cost to reach it and the highest known sequence number for the destination. When a node receives an entry for a particular destination with a higher sequence number its old entry is replaced with the newer route. In the case where a node has to choose between two entries with the same sequence number, it selects the path with the least cost. An intermediate node that detects a broken route to a destination assigns an infinity value to the route’s path cost, increments the entry destination sequence number and immediately broadcasts the information as an update. Using this technique critical network topology information such as link breakages is disseminated quickly across the network.

Advantages and Disadvantages

The main advantage of DSDV over traditional distance vector routing protocols is that it guarantees loop freedom.

The protocol has a number of drawbacks. Optimal values for the parameters like maximum settling time for a particular destination are difficult to determine. This might lead to route fluctuations and spurious advertisements resulting in waste of bandwidth. DSDV uses both periodic and triggered routing updates, which could also cause excessive communication overhead. In addition, in DSDV a node has to wait until it receives the next route update originated by the destination before it can update its routing table entry for that destination. Finally, DSDV does not support multi-path routing.

6.2 Reactive

Reactive routing schemes only become active after there is a request for a route. Reactive routing protocols have also coined the term on-demand protocols since these routing schemes create and maintain routes only when such routers are in demand. That’s why it is also called as the **Source Initiated on Demand Routing protocols.** There is no periodic update of routing information between the nodes in the network with reactive protocols and so it is most often the case that a requested route is not known a priori. When required a node in the network requiring a route has to perform some type of route discovery to find a suitable route. Once a route is found, the node can begin transmission of data packets towards the intended destination. If the conditions in the network remain similar to the instant the route discovery process created the route, the route can be used without disruption as long as it is needed. If however conditions do change, due to link breakages or mobility, the source node has to repair the route or re-create it. Thus reactive routing protocols, in general, have a two phase operation: a route discovery phase and a route maintenance phase. Some well-known reactive protocols are Dynamic Source Routing (DSR), Ad-Hoc On-Demand Distance Vector Routing (AODV), and Temporally Ordered Routing Algorithm (TORA) etc.

The motivation in the design of this ad-hoc routing philosophy is to reduce the protocol routing overhead created by periodic updates of the table-driven schemes. The proactive schemes also use significant resources to maintain certain routes which have the possibility of never being used. This is avoided by the reactive schemes which only create and maintain routes when they are needed.

Reactive (On-demand) protocols cause delays since the routes are not already available. Additionally, the flooding of the network may lead to additional control traffic, again putting strain on the limited bandwidth.

6.2.1 Dynamic Source Routing Protocol (DSR):

This is a simple and self containing protocol that is used in MANET [1]. In this case the entire packet contains the detailed information about the routing path. So no extra processing is required in the middle nodes of the path. More over no administrative things are also not required. This will happen only when a path is required to establish. Since each knows the details of the total view of the networking, it finds the suitable path and adds this path to the packets that are being sent.

Whenever a node changes it’s position it is broadcasted to the all possible node describing how many “hops” are required to reach it.

“The protocol is composed of the two main mechanisms of "Route Discovery" and "Route Maintenance", which work together to allow nodes to discover and maintain routes to arbitrary destinations in the ad hoc network. All aspects of the protocol operate entirely on-demand, allowing the routing packet overhead of DSR to scale automatically to only that needed to react to changes in the routes currently in use.”

The Dynamic Source Routing (DSR) protocol is based on the concept of source routing in which a source node determines the complete sequence of nodes through which to forward data packets. A node sending a packet to a destination node explicitly lists the route to the destination in the header of the packet. The list identifies each “next hop” node that should be taken in order to get from the source to the destination. Each node in the network maintains a route cache that contains source routes that the node is aware of. The route cache is continually updated with old unused routes being purged and new routes inserted as a node learns about them.

Characteristic of an on-demand algorithm DSR has two procedures: route discovery and route maintenance. When a node requires a route to a destination its first action is to consult its route cache to determine if it already contains a route to the destination. If an unexpired route is found, the route is used for data transmission. However, if there is no route in the nodes cache, it initiates a route discovery process by generating and broadcasting a route request (RREQ) packet across the network. The RREQ packet contains the IP addresses of the source and destination nodes, a unique route request ID and a route record which will contain the addresses of the sequence of nodes for the route. To limit the number of route requests traversing the network, each node only processes a route request once. The source nodes address and the unique route request ID are temporarily cached and if the node receives another request with the same details it silently drops the packet.

When an intermediate node (any node other than the source and destination) receives a route request that it can process, its first action is to determine if its address is in the packet’s route record. If the route record already contains the nodes address a routing loop has occurred and the packet is dropped. If there is no routing loop, the intermediate node inspects its route cache for an unexpired route to the destination. It generates and sends a route reply (RREP) packet to the source node if such a route is found. If a route is not found in the route cache, the intermediate node adds its own address to the route record in the RREQ and broadcasts it to its neighbors. The route request packet is thus flooded in the network until either an intermediate node or the destination node itself replies to it. This process is shown in Figure 6.2.1(a). Note that the replying node, given a choice between two routes, chooses the route with the least hop count. The route reply packet is routed back to the source node by reversing the order of the next hops in the route record of the original route request packet. The route reply that is sent back to the source node with the route record included. This can be seen in Figure 6.2.1(b).

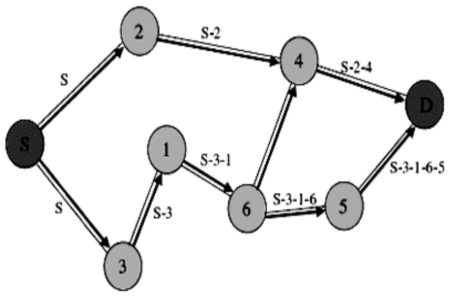

Figure 6.2.1(a): Flooding of the route request to discover route record in DSR

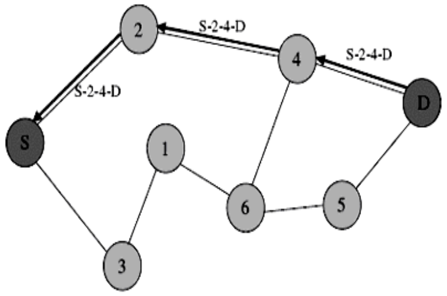

Figure 6.2.1(b): Propagation of Route Reply in DSR

The route maintenance procedure of the protocol monitors the operation of a route and is responsible for making the source node aware of any errors. If an intermediate node detects a failure to transmit a data packet to a downstream link it generates a route error (RERR) packet. When a route error is received by a node, the node in the route error is removed from the nodes route cache and all routes containing that node are truncated at that point. Link errors are detected by means of link layer feedback and/or data acknowledgements.

One of the many optimizations proposed for DSR is the operation of the protocol in a “promiscuous” mode. In this mode the network protocol receives all packets (RREQ, RREP, and RERR) that the node’s wireless interface overhears. These packets are studied for useful source routes or route error messages after which they are discarded [1].

Advantages and Disadvantages

The major advantage of DSR is that there is little or no routing overhead when a single or few sources communicate with infrequently accessed destinations. In such situation, it does not make sense to maintain routes from all sources to such destinations. In DSR, only the sources that desire communication with such destinations need to discover those routes. Furthermore, since communication is assumed to be infrequent, a lot of topological changes may occur without triggering new route discoveries (i.e. has little or no communication overhead).

There are a few drawbacks to the operation of DSR. Even though DSR is suitable for the environment where only a few sources communicate with infrequently accessed destinations, it may result in large delays and large communication overheads in highly dynamic environments. Therefore, DSR may have dynamic scalability problem. As the network becomes larger, control packets and message packets also become larger, since they need to carry the addresses of every node in the path. This may be a problem, since ad-hoc networks have limited available bandwidth. The protocol includes the entire route information in the data packet header which creates significant overhead as the route length increases. DSR also relies heavily on route caches to avoid repeated route discoveries. However, using stale route caches can adversely affect the performance of the protocol. If the routes are not updated a source node may use cached routes which are invalid due to mobility in the network. Intermediate nodes sending route replies using stale cached route could cause pollution of cached routes maintained at other nodes in the network.

6.2.2 TORA (Temporally Ordered Routing Algorithm):

The Temporally Ordered Routing Algorithm (TORA) [10] is based on the concept of link reversal. It was designed with the idea of reducing algorithmic reaction to topological change in a highly mobile ad-hoc network. It is a source initiated protocol that provides multiple routes between source and destination nodes. It detects network partitions quickly and reacts by deleting of invalid routes. There are three basic functions in the operation of TORA

- Route creation
- Route maintenance
- Route erasure

The route creation process establishes a sequence of directed links from the source to the destination node. A logically separate route creation process is run by the source node for each destination with which it communicates. The algorithm creates a Directed Acyclic Graph (DAG) rooted at the destination [10]. In this routing structure, each node in the route is assigned a height metric and the links between neighboring nodes in the DAG are assigned to be either upstream or downstream, depending on the height metric of a node. The height metric is a quintuple comprising of the elements

- Logical time of a link failure
- The unique ID of the node that defined the new reference level
- The reflection indicator bit
- A propagation ordering parameter
- The unique ID of the node

The first three elements of the quintuple collectively represent the reference level of the height metric while the remaining two determine (for each node) the difference with respect to the reference level. It should be noted that “time” is an important factor in the routing operation with the value of the element storing the time of a link failure. The authors of the protocol rely on the fact that all the nodes in the network will have access to synchronized time. This could possibly be provided external time source such as a GPS module.

The route creation is achieved through a query/response cycle. The route creation algorithm starts with the source node broadcasting a QRY packet searching for a route to the destination. This message is flooded across the network until the destination is reached. The destination node then responds by sending a UPD message. The height of the destination is set to 0 and all other nodes are assigned a NULL height value.. A node that receives the UPD packet sets its height value to one more than that of the node from which it received the message. A node with a higher height is considered upstream. The values in brackets next to each node represent the height value. The first value is reference level and the second is the delta with respect to the reference. Note that node 2 does not accept the QRY packet for the destination from node 3 and it has already processed a similar request from node 1.

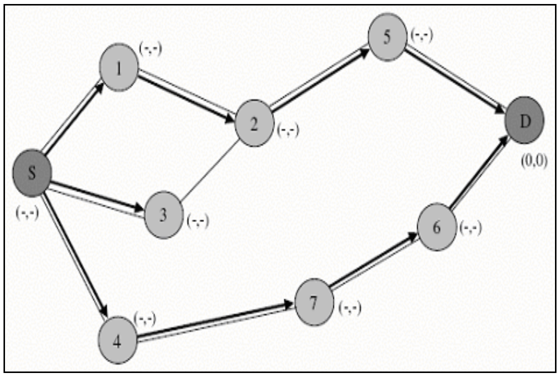

Figure 6.2.2(a): Flooding of QRY

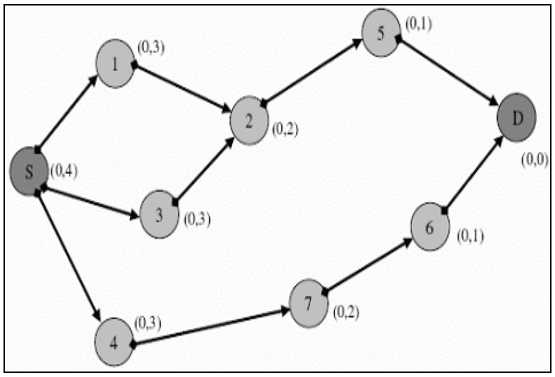

Figure 6.2.2(b): Setting up of the DAG

When the UPD packet reaches the source, all the nodes between the source and destination have been assigned a height value and the source chooses the shortest route for data packet transmissions. This continues unhindered as long as the route is needed or until there is topological change. The novelty of the protocol is in the way it reacts to route breakages due to network mobility. When a node in the network moves, the DAG route is broken and route maintenance function of the protocol is executed to re-establish a DAG for the same destination. If an intermediate node in the DAG loses its last downstream link due to a link failure, it selects a new global maximum height by defining a new reference level.

When the UPD packet reaches the source, all the nodes between the source and destination have been assigned a height value and the source chooses the shortest route for data packet transmissions. This continues unhindered as long

as the route is needed or until there is topological change. The novelty of the protocol is in the way it reacts to route breakages due to network mobility. When a node in the network moves, the DAG route is broken and route maintenance function of the protocol is executed to re-establish a DAG for the same destination. If an intermediate node in the DAG loses its last downstream link due to a link failure, it selects a new global maximum height by defining a new reference level.

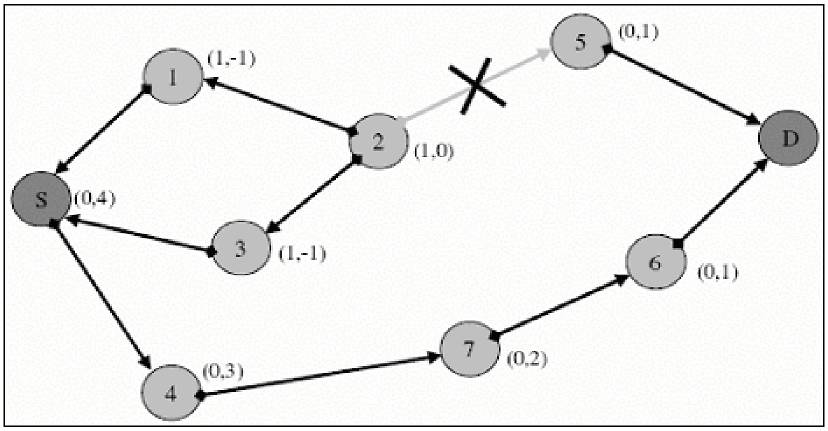

Figure 6.2.2(c): Re-establishment of a route upon a link failure

In Figure 6.2.2(c) the link break between nodes 2 and 5 results in node 2 selecting a new reference level. This results in link reversals which result in other nodes losing their last downstream links to the destination. These nodes perform a partial link reversal to reflect the changes and adapt to the new reference level. The link breakage is only affects a few nodes, thus minimizing protocol reaction to the change. However, in the event of losing all downstream links to the destination, the source node re-establishes a route by means of a new route discovery. The protocols route erasure phase consists of a network flood of broadcast clear packet (CLR) which erases invalid routes in the network.

Advantages and Disadvantages

There is the possibility of oscillations occurring in the operation of TORA. This is caused especially when there are multiple sets of coordinating nodes that are concurrently detecting partitions, erasing routes and creating new routes based on each other. These oscillations are similar to the “count to infinity” problem experienced by traditional distance vector algorithms, except that the problem is temporary and route convergence occurs eventually.

6.2.3 AODV (Ad-hoc On-demand Distance Vector):

AODV [8, 9] uses a route discovery process to dynamically build new routes on an as need basis. AODV is a distributed algorithm using distance vector algorithms, such as the Bellman Ford algorithm. When a route to a destination is unknown, AODV creates a route request packet and broadcasts it to its neighbors. Route request messages contain the source ID, destination ID, source sequence numbers, destination sequence numbers, hop count and broadcast ID. The source sequence number and broadcast ID increment each time a new route request is generated. The destination sequence number is the source sequence number of the destination node as last recorded by the source node.

Each intermediate node receiving a route request caches the previous hop for the particular node originating the request; this helps to create a return path for the reply packets. AODV uses the destination sequence number to maintain freshness of routes. The destination node or any intermediate node can reply to a route request. If an intermediate node has previously learned the path to the destination node, it can reply with the next hop information only if it satisfies the following condition: the locally stored destination sequence number is higher or comparable to the destination sequence number in the route request packet. AODV relies heavily on the sequence numbers to avoid the count-to-infinity problem associated with distance vector protocols. The broadcast ID and source ID pair help in discarding any redundant requests that reach a node. The replying destination or intermediate node unicasts a route reply message to the specific source node that created the route request. Nodes receiving a route reply message store the source ID of the node forwarding the message as the next hop towards the destination in order to forward future traffic toward this destination. The hop count in each message is incremented by one at each forwarding node, which helps track the distance to the source or destination node depending on the type of the message. A node generating a route request or route reply sets the hop count to zero, which is incremented at each intermediate forwarding node. This incrementing helps the intermediate node to determine the number of hops to reach the source or destination using the current path. The source node receiving a number of route replies from different paths uses the hop count in the route reply messages to choose the one with a lower hop count metric as the shortest route to the destination. Once a route is formed, AODV uses the current route until the route expires or any topology changes occur. Each node also maintains a *"precursor list"* of nodes that help it identify the nodes it has to inform of a broken link. The "precursor list" is created from the route request packets and includes a list of nodes that are likely to use the current node as the next hop.

Each node monitors the status of each of its links, and when a link connectivity change occurs, the node creates a route error message and informs the members of the "precursor list" about the non-reachability of specific routes.

AODV relies on medium access control (MAC) layer schemes or the use of beacon packets at periodic intervals to find the status of its directly connected neighbors. Topology changes or expiring timers associated with the route request, reply and beacon packets allow AODV to detect link failures.

AODV uses a progressive ring search technique to control the broadcast domain. Basically, it increases the time-to-live (TTL) value in each broadcast of the initial route request until it receives a route reply.

Example

Figure 6.2.3(a) depicts a network where in node 1 desires to communicate to node 8. The AODV modules running on node 1 flood the network with route request (RREQ) messages. Each node receiving a RREQ message stores the previous hop and distance to source for the originating RREQ and forwards the RREQ to its neighbors.

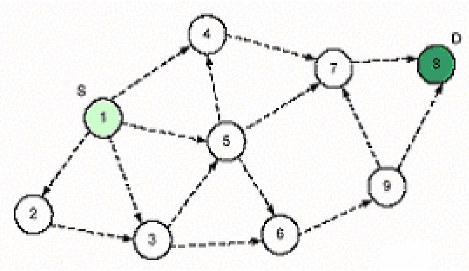

Figure 6.2.3(a): Route request (RREQ) flooding

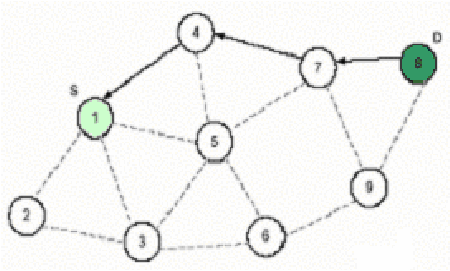

Figure 6.2.3(b): Route reply propagation

When the RREQ message reaches the designation node 8, the destination sends a unicast route reply (RREP) message back to the source using the previous hop on which it received the RREQ. Each node receiving the RREP message in turn forwards it to the next hop with the smallest distance to the source as shown in Figure 6.2.3(b). This process effectively builds the routing table at each node, and when any source destination pair establishes a route, the intermediate nodes learn the route as well.

Advantages and Disadvantages

The advantage of AODV is that it creates routes only on demand, which greatly reduces the periodic control message overhead associated with proactive routing protocols. The disadvantage is that there is route setup latency when a new route is needed, because

ADOV queues data packets while discovering new routes and the queued packets are sent out only when new routes are found. This situation causes throughput loss in high mobility scenarios, because the packets get dropped quickly due to unstable route selection.

6.3 Other Routing Protocols

Other routing protocols are actually hybrid type protocols. This has many sub categories. Among these Geographical or Location Based Routing Protocol is a sub category.

6.3.1 Geographical or Location Based Routing Protocol

The advances in the development of Global Positioning System (GPS) nowadays make it possible to provide location information with a precision in the order of a few meters. They also provide universal timing. While location information can be used for directional routing in distributed ad hoc systems, the universal clock can provide global synchronizing among GPS equipped nodes. Research has shown that geographical location information can improve routing performance in ad hoc networks. An example of Geographical routing protocol is Location-Aided Routing protocol (LAR).

In Geographical protocols additional concern must be taken into account in a mobile environment, i.e., locations may not be accurate by the time the information is used.

6.3.1.1 LAR (Location Aided Routing Protocol):

When using LAR [26], any node needs to know its physical location. This can be achieved by using the Global Positioning System (GPS). Since the position information always includes a small error, GPS is currently not capable of determining a node’s exact position. However, differential GPS[1] offers accuracies within only a few meters.

6.3.1.1.1 Expected zone and request zone

6.3.1.1.1.1 Expected zone:

Consider a node S that needs to find a route to node D. Assume that node S knows that node D was at location L at time t0, and that the current time is t1. Then, the “expected zone” of node D, from the view-point of node S at time t1, is the region that node S expects to contain node D at time t1. Node S can determine the expected zone based on the knowledge that node D was at

[1] Differential GPS (DGPS) is a method of eliminating location errors in a GPS receiver. It makes use of a base station at precisely known coordinates, which computes the difference between the GPS-calculated coordinates and the known location. Thus, the error (which the base station determined) can be transmitted to other GPS receivers and used to correct the signal.

location L at time t0. For instance, if node S knows that node D travels with average speed v, then S may assume that the expected zone is the circular region of radius v(t1 - t0), centered at location L (See Figure 6.3.1.1(a)). If actual speed happens to be larger than the average, then the destination may actually be outside the expected zone at time t1. Thus, expected zone is only an estimate made by node S to determine a region that potentially contains D at time t1. In general, it is also possible to define v to be the maximum speed (instead of the average) or some other measurements of the speed distribution.

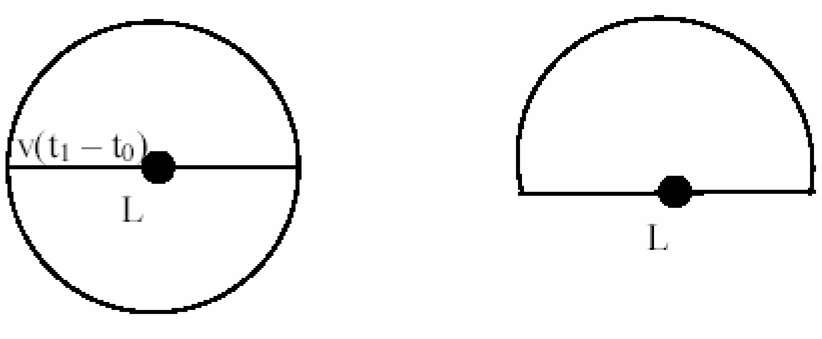

Figure 6.3.1.1 (a): Expected Zone

If node S does not know a previous location of node D, then node S cannot reasonably determine the expected zone – in this case, the entire region that may potentially be occupied by the ad hoc network is assumed to be the expected zone. In this case, their algorithm reduces to the basic flooding algorithm. In general, having more information regarding mobility of a destination node can result in a smaller expected zone. For instance, if S knows that destination D is moving north, then the circular expected circular zone in Figure 6.3.1.1(a) can be reduced to a semicircle, as in Figure 6.3.1.1(a) [22].

6.3.1.1.1.1 Request zone:

Again, consider node S that needs to determine a route to node D. The proposed LAR algorithms use flooding with one modification. Node S defines (implicitly or explicitly) a *request zone* for the route request. A node forwards a route request *only if* it belongs to the request zone. To increase the probability that the route request will reach node D, the request zone should include the *expected zone* (described above). Additionally, the request zone may also include other regions around the request zone. There are two reasons for this:

When the expected zone does not include host S, a path from host S to host D must include hosts outside the expected zone. Therefore, additional region must be included in the request zone, so that S and D both belong to the request zone (for instance, as shown in Figure 6.3.1.1(b)).

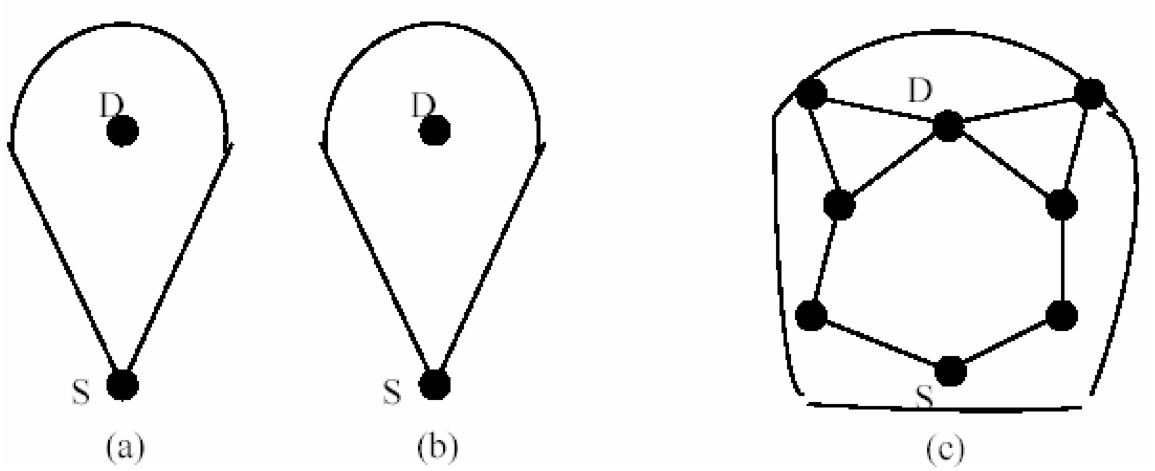

Figure 6.3.1.1 (b): Request Zone in Different Case

The request zone in Figure 6.3.1.1(b) includes the expected zone from Figure 6.3.1.1(a). In some situations this request zone is not adequate. In the example in Figure 6.3.1.1(b), all paths from S to D include hosts that are outside the request zone. Thus, there is no guarantee that a path can be found consisting only of the hosts in a chosen request zone. Therefore, if a route is not discovered within a suitable timeout period, our protocol allows S to initiate a new route discovery with an expanded request zone – in our simulations, the expanded zone includes the entire network space. In this event, however, the latency in determining the route to D will be longer (as more than one round of route request propagation will be needed). Note that the probability of finding a path (in the first attempt) can be increased by increasing the size of the initial request zone (for instance, see Figure 6.3.1.1(b)). However, route discovery overhead also increases with the size of the request zone. Thus, there exists a trade-off between latency of route determination and the message overhead [22].

6.3.1.2 GPSR (Greedy Perimeter Stateless Routing Protocols):

The algorithm consists of two methods for forwarding packets: *greedy forwardin*g, which is used wherever possible and, *perimeter forwardin*g, which is used in the regions greedy forwarding can not be [23].

6.3.1.2.1 Greedy forwarding-

Under GPSR, packets are marked by their originator with their destinations' locations. As a result, a forwarding node can make a locally optimal, greedy choice in choosing a packet's next hop. Specifically, if a node knows its radio neighbors' positions, the locally optimal choice of next hop is the neighbor geographically closest to the packet's destination. Forwarding in this regime follows successively closer geographic hops, until the destination is reached. An example of greedy next-hop choice appears in Figure 6.3.1.2(a). Here, *x* receives a packet destined for D. X's radio range is denoted by the dotted circle about X, and the arc with radius equal to the distance between Y and *D* is shown as the dashed arc about D. X forwards the packet to Y, as the distance between Y and *D* is less than that between *D* and any of X's other neighbors. This greedy forwarding process repeats, until the packet reaches D.

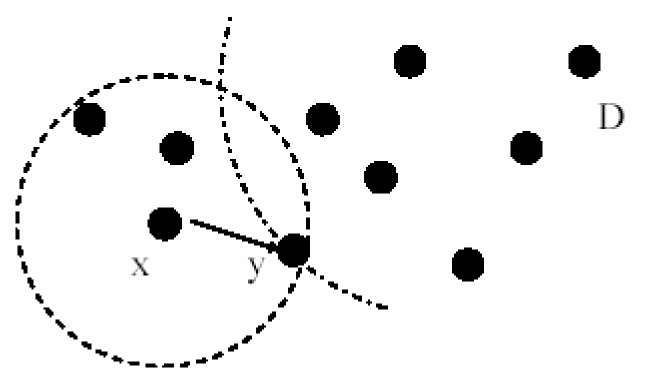

Figure 6.3.1.2(a): Greedy Forwarding Example

Upon not receiving a beacon from a neighbor for longer than time-out interval T, a GPSR router assumes that the neighbor has failed or gone out-of-range, and deletes the neighbor from its table. The 802.11 MAC layer also gives direct indications of link-level re-transmission failures to neighbors.

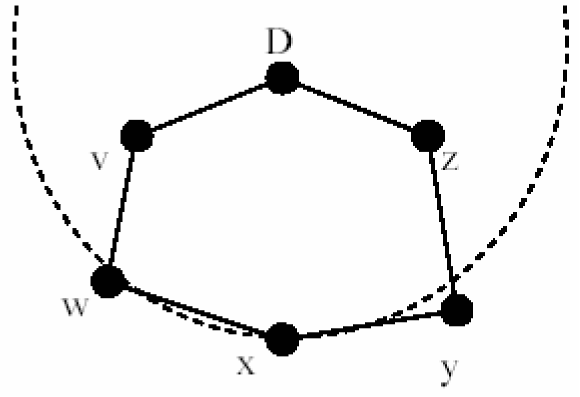

Figure 6.3.1.2(b): Greedy Forwarding Failure

The power of greedy forwarding to route using only neighbor nodes' positions comes with one attendant drawback: there are topologies in which the only route to a destination requires a packet move temporarily *farther* in geometric distance from the destination. A simple example of such a topology is shown in Figure 6.3.1.2(b). Here, X is closer to *D* than its neighbors W and Y. Again, the dashed arc about *D* has a radius equal to the distance between X and D. Although two paths, (X→Y→Z→D) and (X→W→V→D) exist to D, X will not choose to forward to W or Y using greedy forwarding. *X* is a local maximum in its proximity to D. Some other mechanism must be used to forward packets in these situations [23].

6.3.1.2.2 Perimeter forwarding-

It is noted that the *intersection* of X's circular radio range and the circle about *D* of radius | XD | (that is, of the length of line segment XD) is empty of neighbors. X seeks to forward a packet to destination *D* beyond the edge of the void region. Intuitively, X seeks to route *around* the void; if a path to *D* exists from X, it doesn't include nodes located within the void (or X would have forwarded to them greedily).

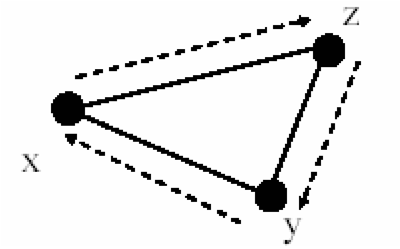

Figure 6.3.1.2(c): Right Hand Rule

The long-known *right-hand rule* for traversing a graph is depicted in Figure 6.3.1.2(c). This rule states that when arriving at node *x* from node y, the next edge traversed is the next one sequentially counterclockwise about *x* from edge (x, y). It is known that the right-hand rule traverses the interior of a closed polygonal region (a *fac*e) in clockwise edge order—in this case, the triangle bounded by the edges between nodes x, y, an*d* z, in the order *(Y→X→Z→Y*). The rule traverses an exterior region, in this case, the region *outside* the same triangle, in counterclockwise edge order [23].

6.3.1.2.3 Combination of Greedy and Planar Perimeter-

The full Greedy Perimeter Stateless Routing algorithm is the combination of greedy forwarding on the full network graph with perimeter forwarding on the planarized network graph where greedy forwarding is not possible. Recall that all nodes maintain a neighbor table, which stores the addresses and locations of their single-hop radio neighbors. This table provides all state required for GPSR's forwarding decisions, beyond the state in the packets themselves. Upon receiving a greedy-mode packet for forwarding, a node searches its neighbor table for the neighbor geographically closest to the packet's destination. If this neighbor is closer to the destination, the node forwards the packet to that neighbor. When no neighbor is closer, the node marks the packet into perimeter mode [23].

7. About the Simulation Environment:

Simulation can be used to analyze the performance of different protocols in different scenario. In this network simulator (ns-2) was used to analyze the performance of the protocols in different scenario and in this way we can compare among the protocols. Basically the main thing behind the simulation is to measure the ability of the routing protocols to react to network topology changes while continuing to successfully deliver data packets to their destinations.

Basically the NS-2 package consists of the following things:

- TCL: (Tool Command Language) an open source scripting language.
- OTCL: An extension of TCL for object oriented programming
- TCLCL: TCL/C++ interface
- NS-2: Network Simulator Version 2

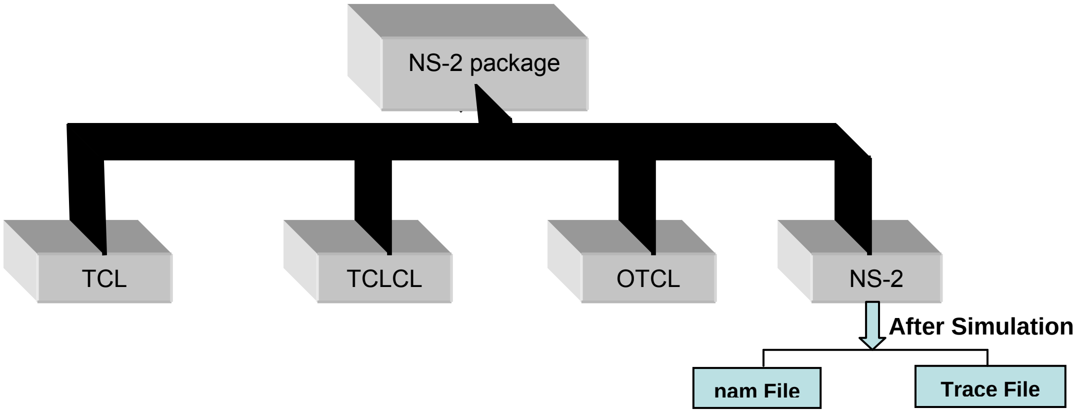

Figure 7: NS 2 package

After the simulation of a particular TCL file generate the nam file that is for the visualization and trace file for the data file that can be analyzed for the performance of

7.1 Mobile Networking in NS

The wireless model essentially consists of the “MobileNode” at the core, with additional supporting features that allows simulations of multi-hop ad-hoc networks, wireless LANs etc. The “MobileNode” object is a split object. The C++ MobileNode../ns-2/mobilenode.h is derived from parent Node../ns-2/node.h (**NS**). A “MobileNode” is the basic Node object with added functionalities of a wireless and mobile node like ability to move within a given topology, ability to receive and transmit signals to and from a wireless channel etc. A major difference between them, though, is that a MobileNode is not connected by means of Links to other nodes or mobile nodes.

8. Performance Comparisons of the Protocols:

The performance comparisons of MANET protocols basically done in two aspects, those are in the following way

- Overall Comparison
- Comparison with a specific performance metric

The "**Overall Comparison**" section in table 3 shows the performance of the routing protocols in different cases.

The "**Comparison with a specific performance metric**" is shown in table 1 & 2. "Best", "Better", "Good", "Bad", and "Worst" complements are used to refer their approximate performance efficiency of around ≈100%, ≈95% and the define nodes are ≤100, ≤16, ≤200, ≈200 and over 350.

Table 1: Performance Comparison Based On The "Packet Delivery Ratio"

AODV	Best (≈100%) [≤100 nodes] / Good [≈1000 nodes]
DSDV	Better (≈95%) [≤16 nodes; Low mobility] / Bad [≈60 nodes; High mobility]
DSR	Good [≤60 nodes] / Worst [≈200 nodes]
TORA	Bad [≤16 nodes; Max speed;8 m/s]
LAR	Best (≈100%) [≤200 nodes]
GPSR	Best (≈100%) [over 350 nodes]

Table 2: Performance Comparison Based On The "Energy Conservation"

AODV	Good (Overall)
DSDV	Worst (Overall)
DSR	Best (Overall)
TORA	Bad (Overall)
LAR	Best (Overall)
GPSR	Best (Overall)

The comparison with different settings provides information about the relative performance of selected protocols based on specific network scenarios. Different scenarios consist of different settings with various types of mobility models, maximum speed of the nodes, packet size etc.

So far, the protocols have been analyzed theoretically, table 3 summarizes and compares the result from these theoretical analyses and shows what properties the protocols have and do not have.

As it can be seen from the table 3, none of the protocols support power conservation or quality service. This is however working in progress and will probably be added to the protocols. All protocols are distributed, thus none of the protocols is dependent on a centralized node and can therefore easily reconfigure in the event of topology changes.

Table 3: Performance Comparison of Different Protocols

Performance Metrics	DSDV	AOSV	DSR	TORA
Loop Free	Yes	Yes	Yes	No
Multiple Routes	No	No	Yes	Yes
Distributed	Yes	Yes	Yes	Yes
Unidirectional Support	No	No	Yes	No
Multicast	No	Yes	No	No
Security	No	No	No	No
Periodic Broadcasts	Yes	Yes	No	Yes
Reliable or Sequenced data	No	No	No	Yes

DSDV is the only proactive protocol in this comparison. It is also the protocol that has most in common with traditional routing in wired networks. The sequence numbers were added to ensure loop-free routes. DSDV will probably be good enough in networks, which allows the protocols to converge in reasonable time. This however means that the mobility cannot be too high. Basically it contained the same conclusions about the DSDV and designed AODV, which is a reactive version of DSDV. They also added multicast capabilities, which will enhance the performance significantly when one node communicates with several nodes. The reactive approach in AODV has many similarities with the reactive approaches of DSR. They both have a route discovery mode that uses request messages to find new routes. The difference is that it supports unidirectional links. DSR has however one major drawback and it is the source route that it supports unidirectional links. DSR has however one major and it is the source route that must be carried in each packet. This can be quite costly, especially when QoS is going to be used.

None of the presented protocols are adaptive, meaning that the protocols do not any smart routing decisions when the traffic load in the network is taken into consideration. As a route selection criteria the proposed protocols use metrics such as shortest number of hops and quickest response time to a request. This can lead to the situation where all packets are routed through the same node even if there exist better routes where the traffic load is not as large.

9. Range Query (RQ):

Range query [22, 29] is a procedure to determine all tracked objects from an expected region in an ad hoc network. In other words, it can be said that a RQ determines ID's and location information of all objects in a certain geographic region defined by the source node of that RQ. According to this definition the functionality of RQ is limited to only the query of location information and node ID's of the mobile nodes. But from a functional point of view RQ can be applied to query not only the location information and ID's of the mobile nodes but also some additional information from those nodes if necessary. Figure 9 depicts a clear view of RQ.

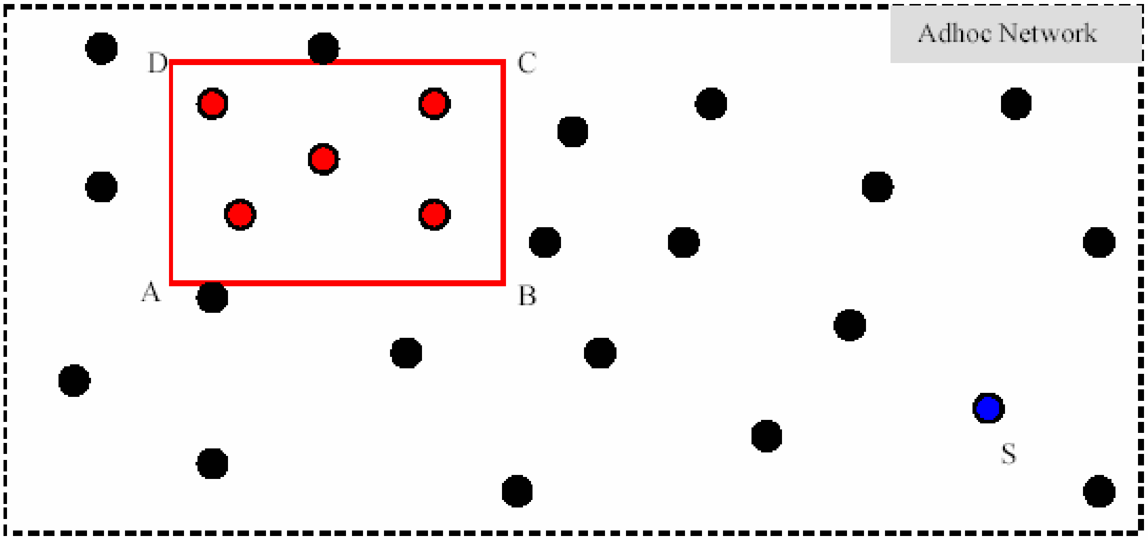

Figure 9: Strategy of the range query

Figure 9 shows a typical scenario of an ad hoc network with mobile nodes partially connected to one another. Let S be the source node. The user of S wants to define an expected region and to collect ID's, location information or other additional information from the mobile nodes situated in that expected region. If in Figure 9, ABCD is the expected region defined by the source, the result of the RQ is supposed to be the collected information from the nodes in the expected region ABCD as shown in Figure 9.

9.1 Mathematical definition of Range Query:

Let,

n be the total number of mobile nodes in an ad hoc network.

A set of n nodes, say AN = {MN1, MN2,…, MNn}, where MN1, MN2,…, MNn are partially connected and any node, say, MNs (called source node) Є AN

An expected region (rectangular shape parallel to axes) defined by MNs is $X_{low} \leq x \leq X_{high}$ and $Y_{low} \leq y \leq Y_{high}$ where X_{low} = value of x of the left corners of the expected rectangle, X_{high} = value of x of the right corners of the expected

rectangle, Y_{low} = value of y of the lower corners of the expected rectangle and Y_{high} = value of y of the upper corners of the expected rectangle.
The set of the unknown number of nodes in that expected region, ER = {all nodes located in the area $X_{low} \leq x \leq X_{high}$ and $Y_{low} \leq y \leq Y_{high}$ } where, ER $\subseteq$ AN and MNs Є ER or MNs Є ER.
Therefore,
Range query at MNs, a set of collected mobile nodes from the expected region, RQ = {all nodes collected from the area $X_{low} \leq x \leq X_{high}$ and $Y_{low} \leq y \leq Y_{high}$ } where, RQ $\subseteq$ ER.

9.2 Application of Range Query:

The interest in wireless ad hoc networks stems from their well-known advantages for certain types of applications. Since there is no need for a fixed infrastructure, a wireless ad hoc network can be deployed quickly. Such a network is tolerant of the failure or departure of terminals, because the network does not rely on a few critical terminals for its organization or control. The application of RQ in ad hoc networks may bring some advantages of RQ to reality. Some examples of RQ applied in ad hoc networks are described as follows.

One example application of RQ in ad hoc networks is emergency search-and-rescue operations in any disastrous area. As the ad hoc network is infrastructure-less and tolerant of any terminal failure, it seems to be consistent in any kind of natural disaster.

RQ may be a part of our daily information services. It can be applied in the area of fleet management to find all trucks that are in a given part of a city. For example, in a city guide application, an information service can be applied with the help of any location service for public transportation that might want to announce the delay of a bus to all users waiting at the next stoppage.

Many mobile applications require some knowledge about the current geographic locations of the mobile objects involved. For example, if the situation is like that a lot of robots are working in an environment which is not suitable for human access during working period, but anyhow the operator needs to collect positions or any information of a group of robots, in this kind of case, querying information from a group of moving robots by a RQ may be a suitable idea. Further future research may make RQ beneficiary to data replication in a group of mobile nodes in a mobile ad hoc network.

10. A Spectrum of Algorithms

Accumulation of suitable algorithms is required to establish the strategy of RQ. As a requirement of that accumulation, a thorough analysis of the existing algorithms is required. The principle of **range queries** is quite related to the principle of **Broadcas**t, **Multicas**t, **Geocast and Unicast** communication paradigms. Any particular information is provided to all the nodes in a network by broadcast. By multicast, any information is delivered to a particular number of known nodes in a network, whereas any information is delivered to all the nodes situated in a particular region in a network by geocast. According to the definition of RQ, it delivers the information of the source node to the nodes in the expected region as well as collects necessary information from those nodes. Here, the first portion of the principle of RQ is partially similar to broadcast, multicast and very similar to geocast. So the algorithms suitable to realize the above methods may also be suitable for realizing RQ.

Plain flooding is suitable for realizing broadcast. Geocast can be realized by plain flooding or **directional flooding**. For unicast, we can use any kind of algorithm like plain flooding, directional flooding, any kind of position based routing like Location Aided Routing (**LAR**), Distance Routing Effect Algorithm for Mobility, **GPSR** etc.

11. Flooding:

11.1 Plain Flooding:

Flooding [28] is a straight-forward approach to perform broadcast. A host, on receiving a broadcast message for the first time, has the obligation to rebroadcast the message. Some drawbacks of flooding are clearly noticeable. When a mobile host decides to rebroadcast a broadcast message to its neighbors, all its neighbors already have the message. After a mobile host broadcasts a message, if many of its neighbors decide to rebroadcast the message, these transmissions (which are all from nearby hosts) may severely contend with each other because of the deficiency of back-off mechanism.

11.2 Directional Flooding:

The principle of directional flooding is quite like plain flooding. If flooding is done under any condition in such a way that the message will move only to a particular direction, then this kind of flooding is called directional flooding. The way of the forwarded packets is not defined but the direction of the destination is defined any way. According to the principle of flooding a host is obligated to rebroadcast the message after receiving any message. In directional flooding the host will rebroadcast the received packet after satisfying any predefined conditions applied for mentioning the direction of the destination.

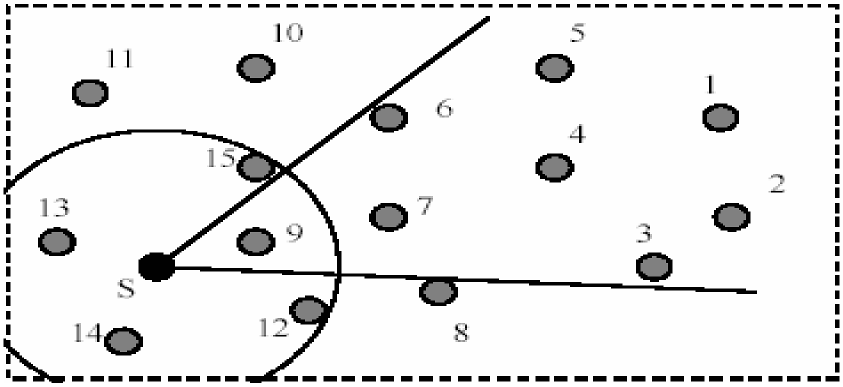

Figure 11.2 Directional Flooding

12. Proposed Idea:

If the source node is in the expected region (**ER**) then the source node will select the **Plain Flooding** mechanism for forwarding packets**,** if it is in the forwarding zone then it will use **Directional Flooding** and the Replying strategy will be **GPSR** algorithm to be more effective communication among the nodes.

The following basic steps are used for realizing Range Query (RQ) in proposed idea:

Step 1: *Selecting an expected region-*
Same as the section 8 of Range query

Step 2: *Flooding by the source node-*
The source node will flood the query packet in the network. The query packet will contain the definition of the expected region, ID and location information of the source node which will be used for replying, packet type and sending time. The source node may use **plain flooding** or **directional flooding** depending on its own position. It will check its own position whether it itself is in the expected region or not. If it prevails in the expected region, it will use **plain flooding** otherwise it will use **directional floodin**g. The use of directional flooding will decrease the network overhead. By applying simple geometry, we can easily determine a forwarding zone. The nodes in the forwarding zone will only actively join the query procedure and thus plain flooding will be reduced to directional flooding. The forwarding zone will be ended by considering a straight line just after the expected zone or by considering an arc after the expected zone with the radius of the length between the source and the furthest point of the expected region. So, the forwarding zone will be a triangular area including the expected region.

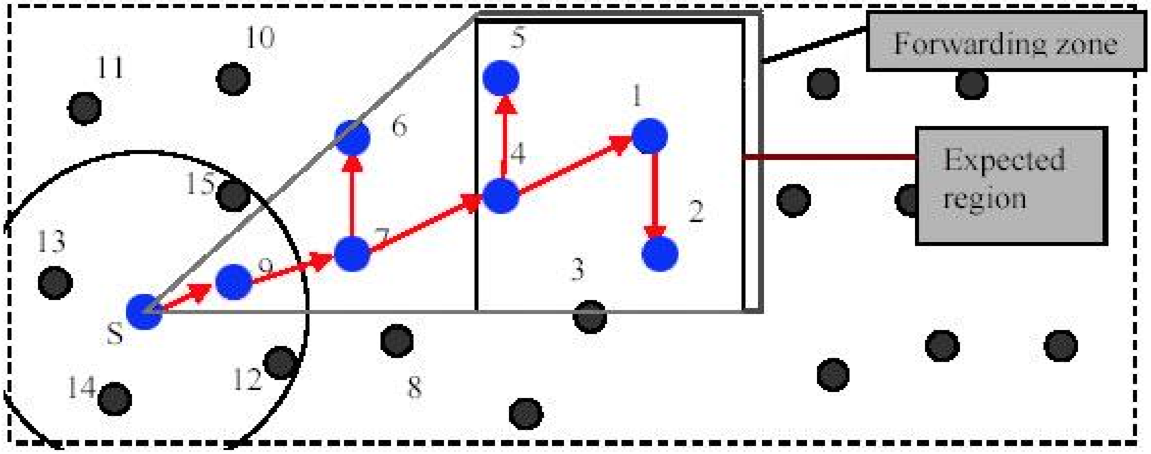

Figure 12: scenario of directional flooding while forwarding query packet

Step 3: *Receiving by the destination node-*
The intermediate nodes on the way to the destination will receive the flooded packet. Firstly, they will check up their own geographical locations whether they are in the forwarding zone or not. If there are not in the forwarding zone they will drop the packets otherwise they will go through another check that they are in the expected region or not. If they are not in the expected zone then they will forward the packet to their neighbors by directional flooding otherwise they will use plain flooding to flood the packets to send it to their neighbors.

Any packet will not be forwarded or received by the same node twice. To avoid this kind of repetition, each packet will have an identification number and this number will be encountered and stored by each node when it will face the packet for the first time. Each time a node will receive a packet, it will check up its identification number into its database and then continue its next action. If the packet identification number is found into its database, the packet will be discarded, otherwise forwarded or received satisfying next procedure mentioned above.

Step 4: *Reply by the destination node-*
The destination node will send the reply packet, which will contain location information of the source node and the queried information, i.e. identity and location information of the replying node or additional information. The replying routing approach will be **GPSR** as the position of the source node is known. Before sending the reply, the destination node may check the location information of the source node into its database to get the latest updated location information of the source node. Here **GPSR** is using because it has a great successful packet delivery ratio as well as the power conservation of the node are very higher than the other protocols.

Step 5: *Checking and storing by the source node-*
Last of all the source node will receive the reply packets and unpack them to retrieve queried information. It will compare the ID and position of each node with the ID and position of the same node contained in the location table (LT) (if necessary) and then store the queried information into its database.

Basically it is a proposed idea for this thesis to find out the destination for realizing range query in MANET.

13. Future work

In this thesis all the comparisons among different protocols are done based on only two criteria or performance metrics. There are also many other criteria exists that can be done in future work. More over here all the comparisons are made based on usual situation. But there are lots of situations even some are extreme situations that also can be done in future work. Last of all the theoretical concept of our proposed idea can be implemented in the future as well.

14. Conclusion

The thesis work has come to an end with a successful comparison and the analysis of the different routing protocols in MANET as well as the range query (RQ) which is the most fascinating topics in MANET obtained from the theoretical concept of the different routing protocols and by using the network simulator 2. The overall report has been organized concentrating on two major areas. The first one is the comparison among the different routing protocols that was help to analysis of the algorithms supposed to be suitable for the realization of RQ. The outcome of this analysis has been used in the 2nd and main point of attention where the strategies of RQ have been proposed with the accumulation of suitable algorithms. The proposals of RQ have been analyzed theoretically in using some metrics of measurement.

15. References

[1] Master's thesis on "Routing Protocols in Wireless Ad Hoc Networks A Simulation Study", Tony Larsson and Nicklas Hedman, Lulea University of Technology.

[2] S. Corson and J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", The Internet Society, Request For Comments (RFC) 2501, Jan. 1999.
Available at, http://www.ietf.org/frc/rfc2501.txt

[3] Tahmid Al-Mumit Quazi, "Design and Implementation of an On-demand Ad-hoc Routing Algorithm for a Positional Communication System", M. Sc. thesis, School of Electrical, University of Natal, Durban, South Africa, September 2003.

[4] Abu Raihan Mostofa Kamal, "Adaptive Secure Routing in Ad Hoc Mobile Network", Master of Science Thesis, Department of Computer and Systems Science (DSV), Royal Institute of Technology (KTH), Stockholm, Sweden, November 01, 2004.

[5] Mobile Ad-hoc Networks (MANET).
Available at, http://www.ietf.org/html.charters/manet-charter.html

[6] D. Bertsekas and R Gallager, "Data Networks", Prentice Hall, 1987, ISBN: 0131968254.

[7] C.S.R. Murthy and B.S. Manoj. Ad Hoc Wireless Networks: Architectures and Protocols. Pearson Education, pp. 207-208,304, 2004.

[8] C. E. Perkins and E. M. Royer, "Ad hoc On-demand Distance Vector Routing," Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications, 1999, pp 90–100.

[9] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," Internet Engineering Task Force (IETF) draft, November 2002.
Available at, http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-12.txt.

[10] V. D. Park and M. S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks", Proceedings of INFOCOM '97, April 1997, pp. 1405-1413.

[11] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", Computer Communications Review, October 1994, pp. 234-244.

[12] T. Clausen, P. Jacquet, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L. Viennot, “Optimized Link State Routing Protocol,” Internet Engineering Task Force (IETF) draft,March, 2002.
Available at, http://www.ietf.org/internet-drafts/draft-ietfmanet-olsr-06.txt.

[13] M. Mauve, J. Widmer, and H. Hartenstein, “A survey on position-based routing in mobile ad-hoc networks,” IEEE Network Magazine, vol. 15, no. 6, pp. 30–39, November 2001.

[14] T. Lin, S. F. Midkiff, and J. S. Park, "Minimal Connected Dominating Set Algorithms and Application for a MANET Routing Protocol," Proceedings of IEEE International Performance, Computing, and Communications Conference, 2003, pp. 157-164.

[15] T. Lin, “Mobile Ad-hoc Network Routing Protocols: Methodologies and Applications,” unpublished doctoral dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, March 2004.

[16] J. Broch, D. Maltz, D. Johnson, Y-C. Hu, and J. Jetcheva, “A Performance Comparison of Multi-hop Wireless Ad hoc Network Routing Protocols,” Proceedings of IEEE/ACM MOBICOM, 1998, pp 85–97.

[17] S. Das, C. Perkins, and E. Royer, “Performance Comparison of Two On-demand Routing Protocols for Ad hoc Networks,” Proceedings of IEEE INFOCOM, 2000, pp. 3-12.

[18] T. Lin, “Mobile Ad-hoc Network Routing Protocols: Methodologies and Applications,” unpublished doctoral dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, March 2004.

[19] Y. B. Ko and N. H. Vaidya, “Using Location Information in Wireless Ad Hoc Networks”, Vehicular Technology Conference, 1999 IEEE 49th, Volume: 3. pp 1952-1956

[20] S. R. Das, C. E. Perkins, E. M. Royer, and M. K. Marina, “Performance comparison of two on-demand routing protocols for ad hoc networks,” IEEE Personal Communications Magazine, special issue on Mobile Ad Hoc Networks, vol. 8, no. 1, pp. 16–29, February 2001.

[21] Samba Sesay, Zongkai Yang, Biao Qi and Jianhua He, “Simulation Comparison of Four Wireless Ad hoc Routing Protocols”, Information Technology Journal 3 (3): 219-226, 2004, ISSN 1682-6027, Asian Network for Scientific Information, 2004.

[22] Kazi Atiqur Rahman “Strategies for Range Queries in Mobile Ad hoc Networks Based on a Geometric Location Model”, INFOTECH-University of Stuttgart, Germany

[23] B. Karp and H. T. Kung, “Greedy Perimeter Stateless Routing for Wireless Networks,” *Proc.6th Annual ACM/IEEE Int’l.Conf. Mobile Comp. Net.,* Boston, MA, Aug. 2000, pp. 243–54.

[24] I. Stojmenovic, “Position-Based Routing in Ad hoc Networks”, IEEE Communication Magazine, July 2002.

[25] J. Li *et al.*, “A Scalable Location Service for Geographic Ad Hoc Routing,” *Proc. 6th Annual ACM/IEEE Int’l. Conf. Mobile Comp. Net.*, Boston, MA, 2000, pp. 120–30.

[26] Y.-B. Ko and N.H. Vaidya, Location-aided routing (LAR) in mobile ad hoc networks, in: *Proc. of ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Dallas, TX (October 1998) pp. 66–75.

[27] J.C. Navas and T. Imielinski, Geocast – geographic addressing and routing, in: *Proc. of ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Budapest, Hungary (September 1997) pp. 66–76.

[28] S. Basagni, I. Chlamtac and V.R. Syrotiuk, Geographic messaging in wireless ad hoc networks, in: *Proc. of IEEE Vehicular Technology Conference (VTC)*, Houston, TX (May 1999) pp. 1957–1961.

[29] X. Li, Y. Jin Kim, R. Govindan, W. Hong, “Multi-dimensional Range Queries in Sensor Networks”, Computer Science Department, University of Southern California, {xinli, youngjki, ramesh}@usc.edu.