Master degree of ICNS

B213120

WSN routing protocol evaluation in special network patterns

By Shengjie Zhou

Supervisor: Dr. Huanjia Yang

Master of Science in

Internet Computing & Network security (ICNS)

of Loughborough University

September 2013

Acknowledgement

I would like to offer my sincerest gratitude to my supervisor, Dr Huanjia Yang, who has supported me throughout my degree in all aspects of project design and NS2 programming. In particular I would like to show my appreciation for his commitment in his role as supervisor throughout the life of this project. Also he has taught me lots of routing protocol concepts in the whole project. Without his guidance, support and patience through my long rambling emails and meetings I would not be handing in this report.

My thanks also go out to all those lecturers who have helped me throughout my degree, particularly Prof Shuanghua Yang who has taught me valuable knowledge of wireless sensor network and understanding in network routing algorithm, which has contributed greatly to my success in this project. In similar vein my thanks to Dr Ling Guan whose guidance during my introduction to network modeling and NS2 programming, in particular pave the way for everything that has followed.

Away from computing I would like to thank to my school mate Weibin Ye, who has help me to research the background of WSN routing protocol. And also taught me some NS2 programming skills. Finally I would like to thank everyone who has contributed in any way to help make this project succeed, including everyone who participated in research, and who has helped me find the webpage and books.

Abstract

Time Is Ticking Out, with the technological growth in recent years, the popularity of wireless networks increased. Wireless networks offer many advantages in the form of availability, mobility and adaptability to the users.

This project will research on wireless sensor network routing protocols, which was an important part in the wireless network. In order to show the performance of each protocol in different network patterns, this project will research use current Linux system and NS2 simulator in an attempt to combine the two and come up with a solution to simulate the protocols. It is hoped that through this research a result of WSN protocols performance comparison can be found.

The solution will be programmed using NS2 for wireless sensor network protocols simulation, and compare the difference of several protocols in varies categories for different network patterns. Learn to Use the NS2 simulation to simulate the wireless network is a great value of learning research to us. The main objective of this thesis: to design a simple wireless simulation scripts and demonstrate the NAM animation on the NS2 simulation. While generating trace files, and then analyzes the trace file, access to the network of the three technical indicators: packet loss rate, throughput , end to end delay. Finally using the Gnuplot tools to draw picture according to the three indicators. Simulation program will be simulated on the NS2 simulation software based on Ubuntu Linux operating system.

Content

1. Introduction

With the rapid development on communication technology, microelectronic technology and multi-functional sensor technology, wireless sensor network (WSN) has become one of the research focuses in computer science. Routing is one of the most important topics in WSN research. This project will focus on the evaluation of several popular routing protocols in various special patterns in WSN.

WSN is usually deployed with a large number of cheap micro sensor nodes in the detection area to form a multi-hop and self-organizing network system, which performs information collection, data processing and wireless communication. The aim is often to process the sensing target information in the network coverage area and supplied to the end user.

WSN is widely used in the military, detection and prediction of the environment, health care, smart home, building condition monitoring, complex machinery monitoring, urban transportation, space exploration, large workshop and warehouse management, as well as airports and large industrial park safety testing and other business fields. IT is expanding into all areas of our everyday life.

Nowadays, Ad-hoc network research is worldwide developed, with important applications in the field of military and emergency relief. It has already become one of the popular research topics of the field of communication technology. All countries around it to carry out a variety of research work; they focus on the network protocols for wireless ad-hoc network. The key issues in networking research are hierarchy design and routing protocol select. A WSN is usually an Ad hoc multi-hop network in which only adjacent nodes can communicate. If the distance between two nodes goes out of the transmission range (typically 150 - 250 m),the communication must go via one or more intermediate nodes to reach the destination. Therefore, one of the key issues of the Ad hoc network is the design of effective and adaptive routing protocol, to achieve high-quality communication even after changes in the network topology. Most research considers the WSN as a random mesh network while design and evaluating their protocol. However, in the real world this is not always the case and a

WSN deployment can contain many special patterns in it depending on the requirement of its applications. Therefore, this project will emphasize research on the protocol performance in different special network patterns and use the NS-2 simulator to evaluate several popular WSN routing protocols. Based on the research result, we hope to provide recommendations or even new design of routing protocols for various patterns we identified. [4][5]

Therefore, Part 2 presents a literature review of WSN routing protocol and introduces protocol that will be used in the simulation.

2. Aims and objectives

This project aims to evaluate several routing protocols of wireless sensor network (WSN) in a few special network patterns, and compare and analysis performance of the protocols. NS2 simulator will be used to simulate the different protocols' performance in varies patterns. The specific aims are displayed as followed:

- Investigate the background of WSN routing protocol
- Introduce the WSN routing protocol category
- Identify the performance features of each protocol.
- Learn how to simulate network routing protocol in NS2
- Design the simulations by introduce data analysis and comparison, such as delay, data integrity, energy consumption, node organization.
- Refine the simulation by considering the different patterns
- Improve the protocols by final result if necessary
- Create the comparison diagram to display in the demonstration

Based on network simulation, the final result is expected to be presenting the difference of several protocols in varies categories for different network patterns.

3. Literature review

As we mentioned in Chapter 1, there are many routing protocols in the WSN, the following protocols was done before, based on their finding and experience to further simulate the network more authenticity.

3.1 Routing Protocols in WSN

Wireless sensor network (WSN) node energy resources, computing power and bandwidth are limited, and wireless sensor networks are usually constituted by a large number of intensive sensor node, which determines the design of the wireless sensor network protocol stack layers must be energy efficiency original design elements. An original design target of the network layer is the establish energy efficiency path, reliable data forwarding mechanism, longest network lifecycle implementation.

There are many different between WSN routing protocol with traditional wireless ad hoc networks (MANET) routing protocol. Wireless sensor network routing design target is to reduce the node energy loss, improve network life cycle. However, traditional wireless ad hoc network routing protocol design is a high quality of service providers in the mobile conditions. This different result in the traditional wireless ad hoc network routing protocols cannot be directly used in wireless sensor networks, many new routing protocol for wireless sensor networks is proposed, and its research has become a hot spot in the wireless sensor network research.

Wireless sensor network routing protocol is different from the traditional network protocol, it has three characteristics with applications: the energy priority, topology information based on the local, data-centric. Thus, depending on the design of application routing mechanism, there are four aspects to measure the pros and cons of routing protocols:

- Energy efficient

Traditional routing protocols have less consider the node's energy problems when they choose the optimal path. Due to the limited energy of nodes in wireless sensor networks, WSN routing protocol not only to select the minimum energy consumption of message transmission path, but also to consider the balance of energy consumption, simple and efficient transmission, to extend the entire lifetime of network as much as possible.

- Scalability

Wireless sensor network application determines the size of its network is not static, and it can easily make the topology changes dynamically. Thus, WSN requires routing protocol scalability and able to adapt the structure change. It embodies in the quantity of the sensor, the network coverage area, the network life cycle, the network latency and network-aware accuracy.

- Robustness

In wireless sensor networks, due to the pattern change and node runs out of energy, lead to the sensor failure and the communication quality decrease. Therefore, the design process of routing protocols must consider protecting the network robustness.

- Fast convergence

Due to the dynamic changes in the network topology, the network requires routing protocol can fast convergence to adapt to the dynamic changes in the topology, increase bandwidth and node energy, also increase the message transmission efficiency.

Nowadays, there are varieties of classifications for different standards routing protocols, still no uniform classification standards. In the next chapters, we will detailed description of the classification on wireless sensor network routing protocols and MANET routing protocols.

3.2 Classification of routing protocols in wireless sensor networks

There are many classification methods on wireless sensor network routing protocols, researchers classification according to different criteria of WSN routing protocols. There is no uniform standard, the following of these classification summarizes and lists some of the classic and characteristic routing protocols.

According to the logical structure of the network, The WSN routing protocol can divide into Data centric protocol and Hierarchical protocol

3.2.1 Data centric protocol

The data centric routing nodes will collect data transmitted to the sink node, all nodes have the same status and functions, they come together to complete data processing. Typical data centric routing protocol as flooding, SPIN, Directed Diffusion (DD) and Rumor routing, etc.

In many applications of sensor networks, it is not feasible to assign global identifiers to each node due to the sheer number of nodes deployed. Such as lack of global identification along with random deployment of sensor nodes make it hard to select a specific set of sensor nodes to be queried. Therefore, data is usually transmitted from every sensor node within the deployment region with significant redundancy. Since this is very inefficient in terms of energy consumption, routing protocols that will be able to select a set of sensor nodes and utilize data aggregation during the relaying of data have been considered. This consideration has led to data-centric routing, which is different from traditional address-based routing where routes are created between addressable nodes managed in the network layer of the communication stack. [9]

In data-centric routing, the sink sends queries to certain regions and waits for data from the sensors located in the selected regions. Since data is being requested through queries, attribute-based naming is necessary to specify the properties of data. SPIN [10] is the first data-centric protocol, which considers data negotiation between nodes in order to eliminate redundant data and save energy. Later, Directed Diffusion [11] has been developed and has become a breakthrough in data-centric routing. Then, many other protocols have been proposed either based on Directed Diffusion [12] and [13] or following a similar concept [14], and so on. In this section, we will describe these protocols in details and highlight the key ideas.

- **Flooding**

Flooding is a traditional flood routing technology, wireless sensor network routing protocols earliest began with Flooding. In flooding, each sensor receiving a data packet broadcasts it to all of its neighbours and this process continues until the packet arrives at the destination or the maximum number of hops for the packet is reached. [9]

Flooding routing protocol is simple to implement, do not need to maintain the network topology and achieve the routing algorithm by consume computing resources, and it suitable for high robustness requirements in any kind of patterns. However, it also has some drawbacks: see Fig.1 and Fig.2 redrawn from [10]. The drawbacks include implosion caused by duplicated messages sent to same node, overlap when two nodes sensing the same region send similar packets to the same neighbour and resource blindness by consuming large amount of energy without consideration for the energy constraints [10].

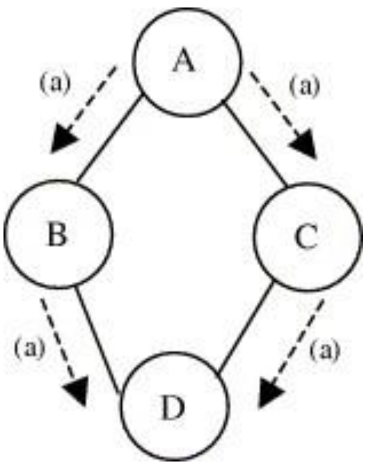


Fig.1. the implosion problem: Node A starts by flooding its data to all its neighbours. D gets two same copies of data eventually, which is not necessary.

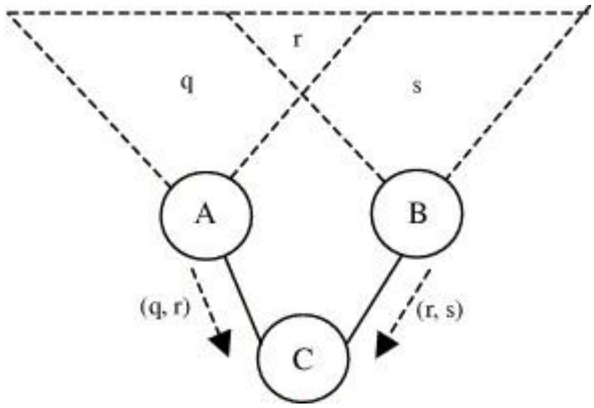


Fig.2 The overlap problem: Two sensors cover an overlapping geographic region and C gets same copy of data from these sensors.

- **SPIN (Sensor protocols for information via negotiation)**

One of the earliest data centric routing protocols is SPIN (Sensor Protocols for Information via Negotiation). Nodes running a SPIN communication protocol name their data using high-level data descriptors, called meta-data [6]. They use meta-data negotiations to eliminate the transmission of redundant data. When a node receives a new data, it broadcast the message and other nodes may retrieve the data by sending a message to that node.

In SPIN, there are three kinds of message type defined to transmission data between nodes. ADV message is to allow a sensor to advertise a particular meta-data, REQ message to request the specific data and DATA message that carry the actual data. Fig. 3, redrawn from [10], summarizes the steps of the SPIN protocol.[9]

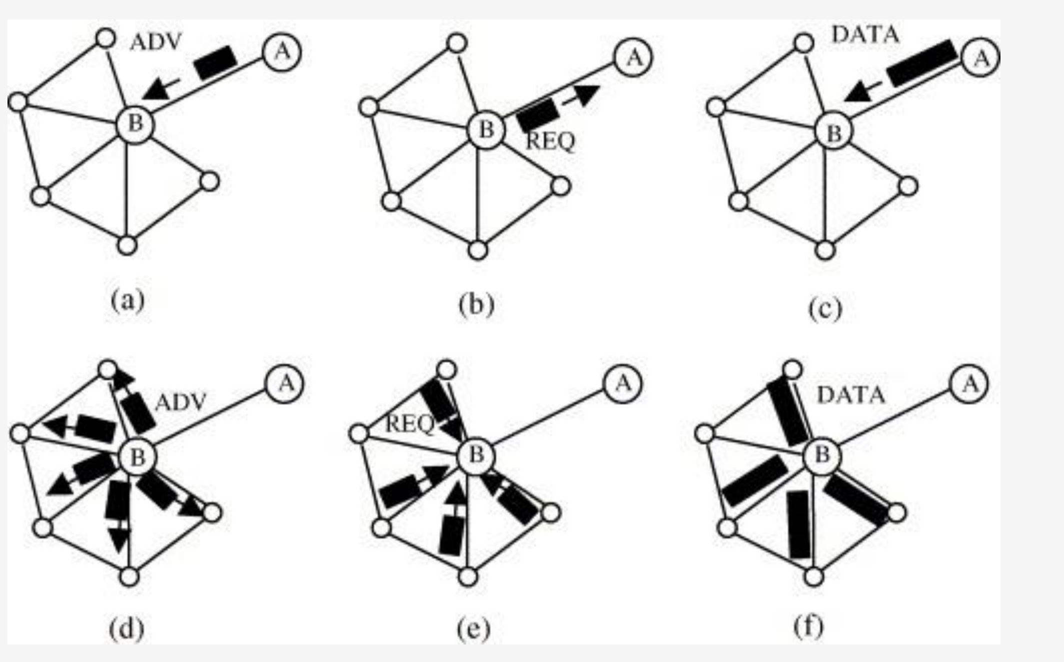


Fig.3 SPIN protocol: Node A starts to send its data to node B (a). Node B responds by sending a request to node a (b). After receiving the requested data (c), node B then sends out advertisements to its neighbours (d), who in turn send requests back to B (e–f).Redrawn from [10].

SPIN have one advantage is that topological changes are localized when each node needs to know only its single-hop neighbours. The SPIN protocol provides 3.5 times less than flooding in terms of energy dissipation and meta-data negotiation almost half of the redundant data. However, SPIN's data advertisement mechanism cannot guarantee the data transmission. For example, if the nodes that are interested in the data are far away from the source node and the nodes between source and destination are not interested in that data, such data will not be delivered to the destination at all. Therefore, SPIN is not a good choice for applications such as intrusion detection, which require reliable delivery of data packets over regular intervals. [9]

- **Directed Diffusion(DD)**

Directed diffusion [15, 16] is a data-centric routing protocol for sensor query dissemination and processing. It meets the main requirements of WSNs such as energy efficiency, scalability, and robustness.DD protocol aims at diffusing data through sensor nodes by using a naming scheme for the data. The main reason behind using such a scheme is to get rid of unnecessary operations of network layer routing in order to save energy. Direct Diffusion suggests the use of attribute-value pairs for the data and queries the sensors in an on demand basis by using those pairs. In order to create a query, an interest is defined using a list of attribute-value pairs such as name of objects, interval, duration, geographical area, etc. The interest is broadcast by a sink through its neighbours. Each node receiving the interest can do caching for later use. The nodes also have the ability to do in-network data aggregation, which is modelled as a minimum Steiner tree problem [17].

At the beginning of the directed diffusion process, the sink specifies a low data rate for incoming events. After that, the sink can reinforce one particular sensor to send events with a higher data rate by resending the original interest message with a smaller interval. Likewise, if a neighbouring sensor receives this interest message and finds that the sender's interest has a higher data rate than before, and this data rate is higher than that of any existing gradient, it will reinforce one or more of its neighbours. Therefore, by using gradients, paths are established between sink and sources. Several paths can be established so that one of them is selected by reinforcement. The sink resends the original interest message through the selected path with a smaller interval hence reinforces the source node on that path to send data more frequently. Fig. 4, redrawn from [11], summarizes the Directed Diffusion protocol.

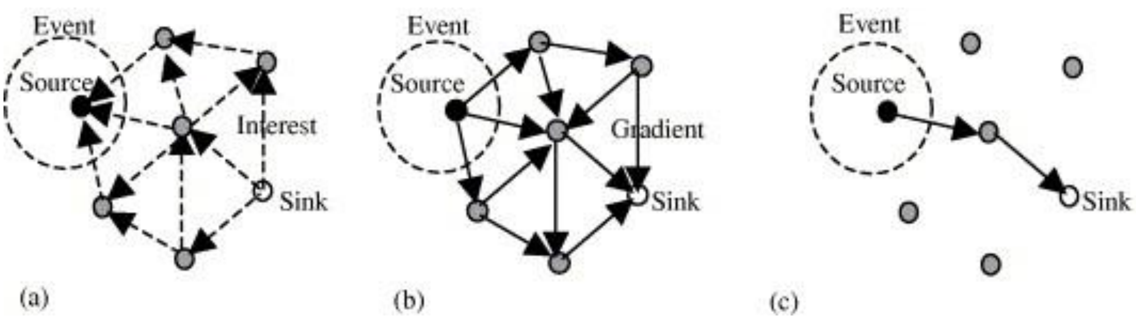


Fig.4 Directed Diffusion protocol phases: (a) Interest propagation, (b) initial gradients setup, (c) data delivery along reinforced.

In Directed Diffusion, Path repairs are also possible. When a path fails between a source and the sink, a new or alternative path should be identified. For this, Directed Diffusion basically restarts reinforcement by searching among other paths, which are sending data in lower rates. Ganesan et al. [18] suggest employing multiple paths in advance so that in case of a failure of a path, one of the alternative paths is chosen without any cost for searching for another one. There is extra overhead of keeping these alternative paths alive by using low data rate, which will clearly use extra energy but more energy can be saved when a path fails and a new path should be chosen. [9]

- **Rumor protocol**

In Rumor protocol, Sink point of a query only once reported, as we mentioned that Directed Diffusion protocol cost is too much, Rumor is designed to solve this problem and this protocol builds on the idea of the Euclidean chance that any two curves has large chance to cross. When the node monitor got the event, it will save and create a longer life cycle called Agent packet. This packet includes information of events and the source node. Then the packet will forward to the network using one or more random path. Agent node received the packet to establish a reverse path, and the Agent again randomly sent the packet to adjacent nodes, moreover it will add its known event information before sending again in the Agent. The request from sink point query also forwarded along a random path, route established when the two paths cross. If the two paths are not crossing, sink point will flooding the request. When in a multi-sink point, the number of queries is large, Rumor protocol is more effective. However, if the event is very large, maintenance event table and transceiver Agent will have a huge cost.

Figure 5 shows in the Rumor protocol, the Agent Communication and Agent path and query path crossing situation. [19]

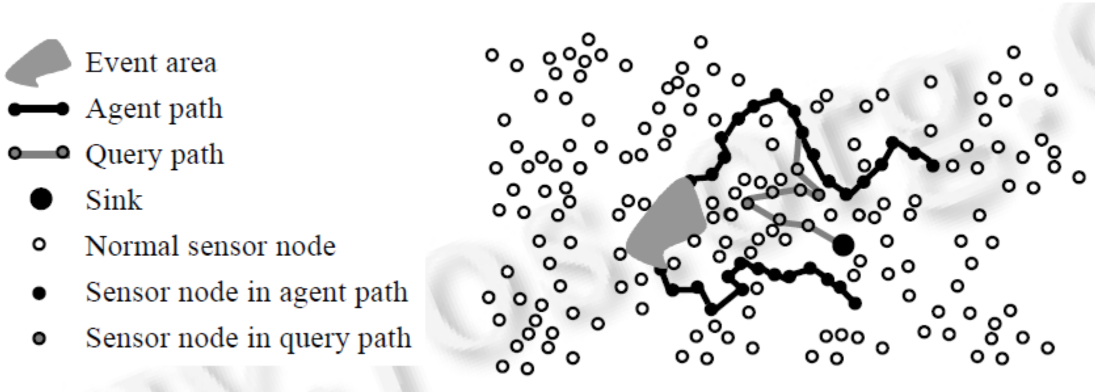


Fig.5. Agent propagation and cross of Agent path and query path in Rumor protocol [19].

3.2.2 Hierarchical Protocol

In the hierarchical structure of the network, cluster composed by associate network node. Clustering is an energy-efficient communication protocol that can be used by the sensors to report their sensed data to the sink [20].In this chapter, we will describe several different hierarchical protocols in which a network is composed of several clusters of sensors. There is a cluster head which manage each of the clusters. The head coordinate the data transmission activates of all sensor in its cluster.

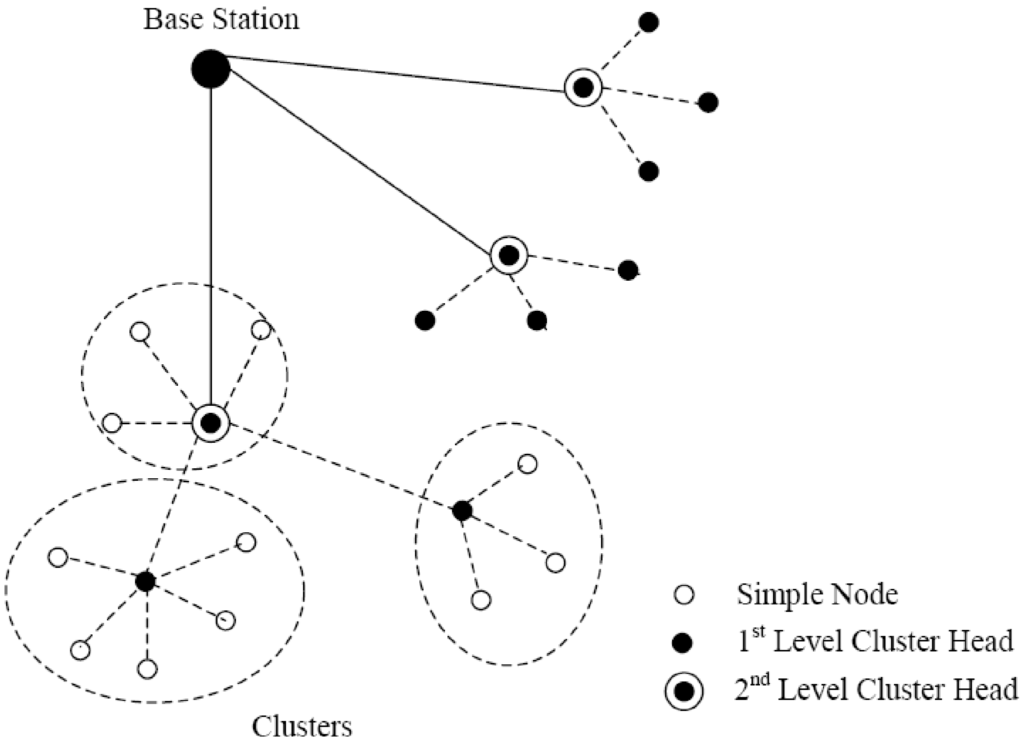


Fig.5. Cluster-based Hierarchical Model [20].

As shown in Figure 5, a hierarchical approach breaks the network into clustered layers [21]. The sensor nodes are grouped into clusters with a head which can transmission data from lower layer cluster head to higher other heads or base stations. Although, it hops from one node to another, but as it hops covers a large distances. This moves the data faster to the base station. In this section, we review a sample of hierarchical based routing protocols for WSN.

- **LEACH**

Low-energy adaptive clustering hierarchy (LEACH) is the first and most popular energy-efficient hierarchical clustering algorithm for WSN [22]. LEACH protocol basic idea is to choose a random cluster head in loop to assigned the whole network energy load average for each sensor node, the purpose is to reduce the network energy consumption and improve the network overall survival time.

The LEACH defined the concept of a "Round", Round is composed by two parts: the establish stage and stable operation. In the initialization phase, LEACH protocol randomly selects a sensor node as the cluster head. The selected principles are sensor nodes randomly generated random number between 0 and 1, if it is greater than the threshold T, the node is elected as the cluster head. After the cluster head chosen, it preferred to broadcast the message "he is the cluster head "to the surrounding nodes, nodes will decide which cluster node he join based on the strength of the received message. Finally, it will broadcast to the appropriate cluster head and establish the communication.

In order to reduce the energy consumption, the stable phase duration is longer than the duration of the establishment phase. Compare with the data centric routing protocols or static routing protocol based on multi-cluster structure, LEACH protocol network of overall survival time is longer.

The disadvantage of LEACH protocol is that each round must be selected clusters and sub-clusters; therefore the establishment phase of the protocol energy cost is larger. Secondly, it requires sensor nodes between with sensor nodes and sink point must have directly communication, so the network scalability is not strong, does not apply to large networks. Based on LEACH algorithm, it improved many new algorithms, such as PEGASIS (Power Efficient Gathering in Sensor Information Systems), etc. [23]

- **TEEN and APTEEN**

TEEN protocol [14], which is a hierarchical routing protocol, it used the filter to reduce the amount of data transfer. This protocol used cluster same as LEACH protocol, but the cluster head according to the sink point distance difference to establish different hierarchical structure. After the clustering completed, the sink point will through the cluster head node to broadcast the two threshold (called the hard threshold and soft threshold) to filter the data transmission.

The initial stage, the cluster head send to all the members the value of a hard threshold (measured value) and a soft threshold value (measured changed value). When monitoring data exceed the set of hard threshold for the first time, the node will use it as a new hard threshold, and sends the data back to base station. In the next process, if the monitoring data compared with the hard threshold, the difference is greater than the soft threshold range, the node will send the latest data to base station, and set it to the new hard threshold. By adjusting the size of the soft threshold, we can obtain a reasonable balance between monitoring accuracy and energy consumption.

The main disadvantage of this protocol is if you do not receive the threshold, the node will not communicate, so that users cannot get data. However, using ARTEEN [25] to a certain extent overcome this defect. Adaptive Periodic Threshold Sensitive Energy Efficient Sensor Network Protocol (APTEEN) [25] is a hybrid routing protocol, use the same pattern as TEEN protocol, but if more than a certain amount of time, the node does not send any data, it will force the node to transmit the data once. Its main disadvantage is increased by the complexity of the protocol requires a soft threshold, hard threshold and timing.

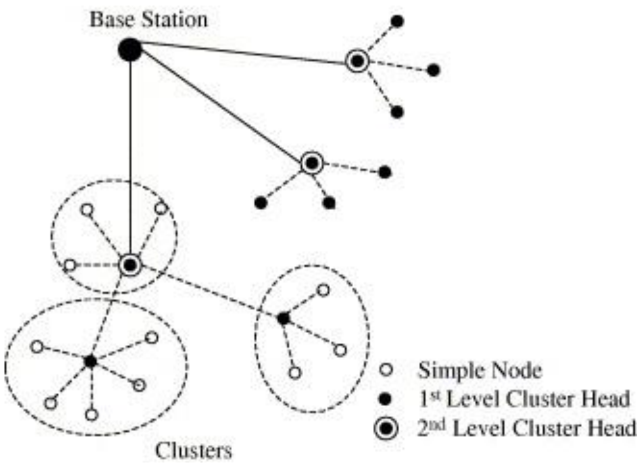


Fig.6. Hierarchical clustering in TEEN and APTEEN. Drawn from[24]

To summarize the TEEN and APTEEN, the simulation results show that these two protocols is better than LEACH [26].In the aspects of the life cycle of the network, APTEEN is between TEEN and LEACH. TEEN gives the best performance since it decreases the number of transmissions. The two protocols in soft threshold and hard threshold update and cluster formation process, the consumption is too large.

3.2.3 Location-based routing protocol

Location based routing protocol is a kind of routing protocols, which use of nodes' location information, instead of links' information for routing. In recent years, many location based routing protocols have been developed for WSN and MANET. In this part we will present the concept of location-based routing protocol, also its advantages and disadvantages. There are two popular location-based protocols will be displayed: Geographic Adaptive Fidelity (GAF) and Geographic and Energy Aware Routing (GEAR).

- **GAF**

Geographic Adaptive Fidelity[27] routing protocol is an energy-aware routing protocol based on the location information, its initial application is in the Ad hoc network, but also applies for many wireless sensor networks. The basic idea of the protocol is the network area divided into many cells, each cell within the nodes cooperates with each part of the node to keep a normal working condition, and then it can complete tasks such as data collection and forwarding. Another part of the nodes is to save energy and extend the network whole life cycle in a sleep state. The worked node is the head of nodes in this area. The disadvantage of this protocol is required each node must through the GPS to get their own geographic information, which greatly increased the cost and complexity, and the node does not apply to many occasions.

- **GEAR**

Geographic Energy Aware Routing [28] similar as GAF protocol, but also the overall network by geographical area is divided into several small areas, it is also an energy-aware routing based on location information. The core idea of this protocol is to limit the directed diffusion protocol messages to a certain area, rather than the broadcast the message to whole network.

In GEAR protocol, the node according the energy and distance information to estimate the cost of the path. When one node to the destination node cost more than to all neighbour nodes, this node is the routing void point; by adjusting the path cost can solve the problem of routing void. Whenever a packet arrives to the destination node, it will adjust the next data packet transmission path by their feedback of the cost information. The data transmission can be divided into two parts: first is passed to the destination area from other areas, second is in the transmission area, similar as transmission between the group and the transmission inside the group.

GEAR routing protocol with GPSR (Greedy Perimeter Stateless Routing) routing protocol are similar, both of them is greedy routing algorithms based on geographic information. The difference is GPSR does not consider the problem of energy consumption. Compare with GEAR, the GPSR network have better energy consumption and performance. GEAR only applies to less node moves network occasions.

- **MECN**

Minimum energy communication network (MECN) [29] protocol is based on the node positioning routing protocols; the basic idea is to use the low-power GPS to build minimum energy properties to reduce the energy consumption in data transmission. In MECN protocol, each node defines a relay area, the relay area is constituted by a series of nodes, and the sensor through the relay area forwards the data to the Sink point, rather than directly sent to the Sink point, because forward can save more energy, so that each node can balance the energy consumption. Use local search algorithm to determine the relay area. The MECN can adapt the network applications with small dynamic changes.

3.2.4 MANET Routing protocols

Existing ad hoc network routing protocols can be divided into two categories:

Proactive Routing Protocol (table-driven) and Reactive Routing Protocols (source-initiated on-demand routing).

- **Proactive Routing Protocol**

Proactive protocols send and receive topological information through the network nodes to learn the topology of the network. Each node in these routing protocols need to keep one or more routing table update. These tables contain the network routing information about which node reach the other nodes. Therefore, when detecting the network topology changes, the nodes will send the update messages. Once other nodes received update messages will update their routing table, it can accurate the network consistent and routing information. According to the change of the routing table, Proactive routing protocol constantly detects the change of the network topology and link quality. Thus, the routing table can accurately reflect the network topology. Once the source node to transmit the packet to its exist destination node in routing table, it can immediately get the massage and the delay is very small. However, it takes a larger cost, because it needs to update the routing table to reflect the current topology changes immediately. Proactive routing protocol include: DSDV (Destination-Sequenced Distance Vector), CGSR (Cluster-Head Gateway Switch Routing) and WRP (Wireless Routing Protocol), etc.

Proactive routing protocols can quickly arrive to all other nodes in the routing table, but the disadvantage is inevitably signal congestion and excessive energy consumption, and the Proactive routing protocols gradually are Reactive routing protocol instead.

- **Reactive Routing Protocol**

Reactive routing protocols do not like the Proactive routing protocols to keep update routing information for each node in every time, but only when the source node require to send, which reduces the network bandwidth and energy consumption. When a source node needs a routing table to reach to a destination node in the network, it will use a finding process to establish a routing table and routing maintenance procedures will keep update, until each path is broken or no longer needs to be routed. Reactive routing protocol include: AODV (Ad hoc on-demand vector routing), DSR (dynamic source routing), TORA (temporally ordered routing algorithm).

3.3 The design key issues of WSN routing protocol compare with MANET.

Compared to MANET, WSN has its own characteristics, which makes routing protocol must be designed according to the characteristics of WSN to adapt the requirement as large-scale network, topology easy to change, energy limited and to extend the network lifetime. The following describes the characteristics of WSN routing protocol design key issues:

WSN compared to the number of nodes in MANET is much larger (thousands or even tens of thousands), node distribution is denser, which makes it suitable for large-scale network. Therefore, the routing protocols must be designed for WSN; Due to the patterns and energy effects, WSN node may easily break down, probably because the network topology changed, which makes the WSN routing protocols need to have a stronger self-adaptability and robustness.

WSN nodes of energy, processing power, and storage capacity and communications capabilities are limited, which makes the routing protocol must be designed simple and energy-saving for WSN.

MANET use point-to-point transmission mode. However, the WSN data transmission use many-to-one or one-to-many mode, which makes the WSN routing protocol design must consider load balancing; Usually the MANET node has highly mobile, while in WSN most of the sensor nodes is fixed; MANET is address-centric routing; WSN is data-centric routing. Moreover they have similar data collection between neighbouring nodes, both of them need data fusion.

MANET original design target is to provide high quality of service and efficient bandwidth utilization, and secondly to consider energy. However, WSN’s original design aim is the efficient use of energy, which makes energy-efficient routing protocol, must be designed for WSN.

WSN routing protocol is designed based on the specific application, it is difficult to design a versatility routing protocol for WSN.

4. MANET protocols concept

In this chapter, different concepts of routing in MANET are shown. Further the mobile ad-hoc routing protocol AODV (Ad-hoc On-Demand Distance Vector), DSR (Dynamic Source Routing), DODV(Destination-Sequenced Distance Vector), AODV+(Modified from original AODV) is discussed in more detail, because it is used in the comparison and improvement.

4.1 DSR

The purpose of the design is to create DSR very low overhead, while changing rapidly respond to a network routing protocol, in order to ensure highly reactive type of service data packets in the mobile node changing network conditions or other conditions can still be properly submitted. DSR use source routing algorithm, each given route data packets are in the header with a complete, orderly grouping must pass through this list of nodes. Using source routing can guarantee no loop, packet forwarding or listening node can cache the packet routing information for later use, but also due to the transmission of data packets already contains the necessary routing information, routing information without saving the intermediate nodes.

In DSR, Route Discovery and Route Maintenance each operate entirely "on demand". In particular, unlike other protocols, DSR requires no periodic packets of any kind at any layer within the network.[42] This entirely on-demand behaviour and lack of periodic activity allows the number of overhead packets caused by DSR to scale all the way down to zero, when all nodes are approximately stationary with respect to each other and all routes needed for current communication have already been discovered. As nodes begin to move more or as communication patterns change, the routing packet overhead of DSR automatically scales to only what is needed to track the routes currently in use. Network topology changes not affecting routes currently in use are ignored and do not cause reaction from the protocol.[42]

4.2 DSDV

Destination sequenced distance vector routing (DSDV) is adapted from the conventional Routing Information Protocol (RIP) to ad hoc networks routing. It adds a new attribute, sequence number, to each route table entry of the conventional RIP. Using the newly added sequence number, the mobile nodes can distinguish stale route information from the new and thus prevent the formation of routing loops.

In DSDV, each mobile node of an ad hoc network maintains a routing table, which lists all available destinations, the metric and next hop to each destination and a sequence number generated by the destination node. Using such routing table stored in each mobile node, the packets are transmitted between the nodes of an ad hoc network. Each node of the ad hoc network updates the routing table with advertisement periodically or when significant new information is available to maintain the consistency of the routing table with the dynamically changing topology of the ad-hoc network.

Periodically or immediately when network topology changes are detected, each mobile node advertises routing information using broadcasting or multicasting a routing table update packet. The update packet starts out with a metric of one to direct connected nodes. This indicates that each receiving neighbour is one metric (hop) away from the node. It is different from that of the conventional routing algorithms. After receiving the update packet, the neighbours update their routing table with incrementing the metric by one and retransmit the update packet to the corresponding neighbours of each of them. The process will be repeated until all the nodes in the ad hoc network have received a copy of the update packet with a corresponding metric. The update data is also kept for a while to wait for the arrival of the best route for each particular destination node in each node before updating its routing table and retransmitting the update packet. If a node receives multiple update packets for a same destination during the waiting time period, the routes with more recent sequence numbers are always preferred as the basis for packet forwarding decisions, but the routing information is not necessarily advertised immediately, if only the sequence numbers have been changed. If the update packets have the same sequence number with the same node, the update packet with the smallest metric will be used and the existing route will be discarded or stored as a less preferable route. In this case, the update packet will be propagated with the sequence number to all mobile nodes in the ad hoc network. The advertisement of routes that are about to change may be delayed until the best routes have been found. Delaying the advertisement of possibly unstable route can damp the fluctuations of the routing table and reduce the number of rebroadcasts of possible route entries that arrive with the same sequence number.[38]

The elements in the routing table of each mobile node change dynamically to keep consistency with dynamically changing topology of an ad hoc network. To reach this consistency, the routing information advertisement must be frequent or quick enough to ensure that each mobile node can almost always locate all the other mobile nodes in the dynamic ad hoc network. Upon the updated routing information, each node has to relay data packet to other nodes upon request in the dynamically created ad hoc network.

4.3 AODV

AODV is a kind of reactive mobile ad-hoc routing protocol. It joins the mechanisms of DSDV and DSR, which uses DSDV protocol "destination sequence number" prevent caching routing information expired and loops, routing is based on DSR protocol to establish the methods used, the difference lies in AODV by-hop routing protocol is not the source routing.

AODV protocol is on-demand routing protocol. When the communication network nodes relatively low, the control overhead and storage overhead than the first node routing protocol should be small, the link interrupt response more quickly and with a certain scalability. The disadvantage is that large routing setup delay. AODV protocol uses the following methods: Using the route cache technology, any intermediate node can return to the source node routing information; timely reporting link 'breaking information quickly rebuild route, to some extent reduce the routing setup delay. In addition, AODV only supports two-way link.

Here also will introduce the AODV message format to facilitate improve the next chapter AODV+ protocol.

Message Format

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

Type	J|R|G|D|U	RESERVED	Hop Count
RREQ ID			
Destination IP Address			
Destination Sequence Number			
Originator IP Address			
Originator Sequence Number			

Figure 4.1: AODV: Route Request (RREQ), Figure redrew from [43]

AODV has four types of messages: Route Requests (RREQ), Route Replies (RREP), Route Errors (RERR), and Route Replies Acknowledgment (RREP − ACK). All these messages are received via UDP using normal IP header processing. AODV uses the IP limited broadcast address (255.255.255.255) to broadcast messages.

0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

Type	R|A	RESERVED	Prefix Sz	Hop Count
Destination IP Address				
Destination Sequence Number				
Originator IP Address				
Lifetime				

Figure 4.2: Route Reply (RREP), Figure redrew from [43]

0 1 2 3 4 5 6 7	8	9 0 1 2 3 4 5 6 7 8 9 0 1 2 3	4 5 6 7 8 9 0 1
Type	N	RESERVED	Destination Count
Unreachable Destination IP Address			
Unreachable Destination Sequence Number			
Additional Unreachable Destination IP Addresses			
Additional Unreachable Destination Sequence Numbers			

Figure 4.3: AODV: Route Error (RERR), Figure redrew from [43]

0 1 2 3 4 5 6 7	8 9 0 1 2 3 4 5
Type	RESERVED

Figure 4.4: AODV: Route Reply Acknowledgment (RREP – ACK), Figure redrew from [43]

Type 1 for RREQ

2 for RREP

3 for RERR

4 for RREP − ACK

[J] Join Flag; reserved for multicast.

[R] Repair Flag; reserved for multicast.

[G] Gratuitous RREP flag; indicates whether a gratuitous RREP should be unicast to the destination node.

[D] Destination only flag; only the destination node may answer to this RREQ, no intermediate node is allowed of answering with a RREP.

[U] Unknown sequence number.

[A] Acknowledgment required; used, if there is a danger of unidirectional links. It causes the receiver of the RREP message to send back a RREP −ACK message. The reception of such an acknowledgment provides assurance that the link is currently bidirectional.

[N] No delete flag; set if upstream nodes should not delete the route, although a node has performed a local repair of a link. [34]

4.4 AODV+

AODV+ is an implementation to use AODV as an ad hoc routing protocol for simulations of wired-cum-wireless scenarios in ns-2. It is currently being implemented for the network simulator ns2 [41]. There are no guarantees that it implements all features correctly.

The used concepts are kept general and could be adapted to a real implementation.

The work is focused on gateway discovery methods in mobile ad-hoc network. Gateways are needed to send packets outside the mobile ad-hoc network. These gateways are able to route messages in both networks. Therefore they support mobile ad-hoc routing and routing in the wired domain. [40] explains the operation of the mobile nodes and the gateways. The application of these concepts for a reactive routing protocol heads in adaptation of the route discovery messages (RREP, RREQ), so that it is possible to detect gateways.

There are different ways to catch information about the gateways:

- the gateway broadcasts periodically messages (proactive gateway discovery)
- the mobile node requests a default route to a gateway by sending a RREQI message (reactive gateway discovery)
- the gateways replies on received RREQ with a RREPI

The mobile node adds this information about the gateway to its routing table as the default route entry. If the mobile node has no route to a certain node, it broadcasts a RREQ. If it does not receive a RREP, the node is supposed to be outside the current mobile ad-hoc network. The packet is sent to this node by using the default route to the gateway.

With AODV+ a mobile node can find a route to a gateway in three different ways:

1. proactive gateway discovery method
2. hybrid gateway discovery method

3. reactive gateway discovery method (default)

- **Proactive Gateway Discovery**

The gateway discovery is initiated by the gateways themselves. Gateway broadcast regularly a gateway advertisement (GWADV) message to the whole mobile ad-hoc network. It is important to carefully choose the broadcasting interval to not flood the network unnecessarily.

The mobile nodes receiving this GWADV message complete their routing tables with the default route to the gateway. Although the problem of duplicated broadcast messages is solved with the same mechanism as in AODV (RREQ ID), the periodical flooding of the whole network remains a big disadvantage of this approach. The cost of the flooding cannot be ignored because of the limited resources in a mobile ad-hoc network.

- **Hybrid Gateway Discovery**

The two approaches are combined. The gateway broadcasts periodically RREPI messages which are forwarded within the advertisement zone. Proactive gateway discovery is used for this area. The nodes outside of it have to start a reactive gateway discovery by sending a RREQI message.

- **Reactive Gateway Discovery**

The mobile node that needs a route to a gateway broadcasts a RREQI . Only gateways react on this message by unicasting a RREPI back to the mobile node. Intermediate nodes only forward the RREQI . They prevent the duplicated forwarding with the known RREQ ID mechanism. The advantage of this approach is that it is purely reactive. Only if routes to the gateway are needed, a request is started.

Disadvantages are that the load on the forwarding node near the gateway is increased and the response time is higher. [39]

5. Implement simulation in NS2

5.1 NS2

NS is originated by REAL network simulator, then DARPA started to support the development of NS. Now NS was developed by DARPA and supported by NSF's SAMAN the CONSER project. NS has an open structure and good distensibility, absorbing the rich set of modules ranging from UCB Daedelus and CMU Monarch working groups, also get SUN's wireless code. [45]

NS is a discrete event simulator. Its core is a discrete event simulation engine. NS has a "Scheduler" category, it is responsible for record the current time, the scheduler network events lined up, and provides functions to generate new events specified event occurs. With discrete event simulation engine, users can be simulated any system not restrict by the communication network system. Users work through the establishment of the system model, the preparation of each event-handling code, then use this discrete event simulator to complete the simulation.[45]

NS is a split object model. Its library is constructed using two object-oriented languages: one is C++, the other is OTcl. OTcl is developed at MIT ObjectTcl, namely Tcl (Toolkit command language) object-oriented extensions. Tcl is a flexible, interactive scripting language, OTcl is added in Tcl classes, instances, inheritance and other object-oriented concepts. In the NS component is implemented as a C++ class, while there are a OTcl class correspond. This split object model using C++ to achieve functional simulation, can make the execution of the simulation get better performance. Tcl simulation configuration, without having to recompile the case freely modify simulation parameters and simulation process, so it improved the efficiency of simulation. Tcl simulated user configuration, just need to understand the use and configuration of the interface elements, then it can be achieved without having to understand the function of these components how to implemented.

5.2 NS2 Simulation procedure

NS simulation process is divided into two levels: one is based on the Tcl programming, using the existing network elements implement NS simulation, without modifying the NS itself, only to write Tcl scripts; another level is based on C + + and Tcl programming, if NS network elements is not required, it is necessary to extend the NS, which means you need to add the network elements [45]. NS simulation process shown in Figure 8

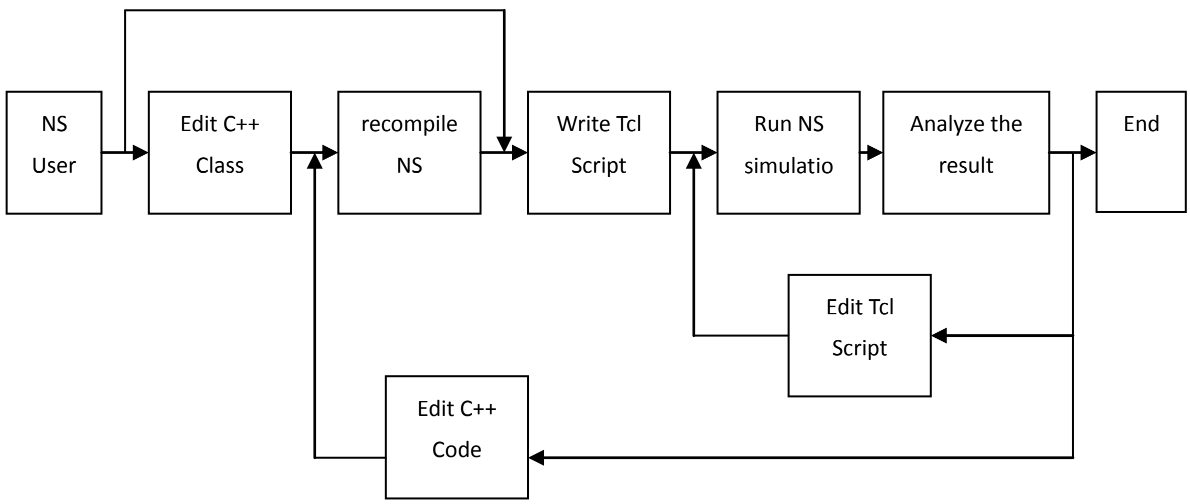


Fig.5. NS simulation procedure

NS network simulation carried out the main steps are as follows:

(1) Write Tcl scripts, this step is the main configuration of network topology, determine the basic characteristics of the link.

(2) Establishment protocol agents, including terminal protocol bindings and communication traffic model build.

(3) Set up model configuration parameters to determine the distribution of traffic in the network.

(4) Set Trace object. Trace object able to simulate the network occurs during certain events happened and it recorded in tr file.

(5) Preparation of write other auxiliary processes, set up the simulation end time.

(6) To use the gawk software for analysis tr file and use tools such as nam viewing network simulation for process.

5.3 NS2 simulation of wireless networks

5.3.1 NS2 wireless network model

The NS2 wireless module is CMU's Monarch working group introduced, the CMU/Monarch model can be introduced in pure wireless networks simulation, further extended to support wireless and wired networks mixed simulation. Wireless module is based on the MobileNode, through some additional features to support multi-hop Ad hoc networks and wireless local area network simulations. MobileNode of C++ class is a derived class of the Node class. MobileNode have basic node, also have wireless and mobile node functions [45].

The mobile node is constituted by a series of network components, and which included: a link layer (LL), connected to the LL of the ARP module, the interface queue (Ifq), MAC layer (Mac), Internet Interface (Network Interface). Mobile node through the network interface to connect to the wireless channel.

5.3.2 cbrgen

cbrgen can be used to create CBR and TCP traffics connections between wireless mobile nodes. In order to create a traffic-connection file, we need to define the type of traffic connection (CBR or TCP), the number of nodes and maximum number of connections to be setup between them, a random seed and in case of CBR connections, a rate whose inverse value is used to compute the interval time between the CBR pkts. So the command line looks like the following:

ns cbrgen [-type cbr|tcp] [-nn nodes] [-seed seed] [-mc connections] [-rate rate] [47]

-type: Specifies the traffics connection is CBR or TCP.

-nn: Specifies the number of the mobile nodes.

-seed: Specifies the number of the seeds in Random Number Generator.

-mc: Specifies the maximum number of the online nodes, that is the number of data streams generated will be smaller or equal to the maximum number of the online nodes.

-rate: If you are using CBR, then the packet is fixed at 512B, while the rate value specifies the number of packets sent out per second

For example, let us try to create a CBR connection file between 10 nodes, having maximum of 8 connections, with a seed value of 1.0 and a rate of 4.0. So at the prompt type:

ns cbrgen.tcl -type cbr -nn 10 -seed 1.0 -mc 8 -rate 4.0 > cbr-10-test

5.3.3 establish a wireless scene (Setdest)

Setdest is used to randomly generate the required node moving scenes, the scene file in the random number of nodes in a fixed size in each of the rectangular area to a random direction. Firstly, we need to generate the Makefile for make. Setdest command takes two forms, because in the back of the simulation using only one of these formats, this paper describes in detail here used setdest command format:

. / setdest-n <num_of_nodes>-p <pausetime>-M <maxspeed>-t <sumtime>

-x <maxx>-y <maxy>> <outdir> / <scenario-file>[44]

Command parameter definitions are mainly:

-n num_of_nodes Specifies the number of nodes in scene file.

-p pausetime Specified node in the movement to a temporary destination residence time.

-M maxspeed Given node is the maximum delivery random movement, the speed of the node [0, maxspeed] randomly selected.

-t simtime Specify the duration of the simulation scene

-x maxx Specified node movement length of the area

-y maxy Specified node movement width of the region

Use setdest generated scene files, you can use the script file NS2 command "source" to call. [46]

5.4 Nam animation

Nam is a kind of Network Animation simulator, it is based on Tcl / Tk animation demo tool for demonstration network running animation. Nam's function is based on network simulation or real environment specific format for the output file nam animation.

Nam's use is: First open an output nam animation format output files, and Tcl script file was written to the corresponding command to set the output file xxx.nam by namtrace-all member functions of the file and animation output linked.

5.5 Performance analysis tools

5.5.1 gawk

gawk is a programming language, the data has a strong analytical skills, you can use a short code to easily complete text of the document modification, analysis, extraction and comparisons.

gawk main function is to search the file for each line of the specified pattern. When in a row matches the specified mode, gawk will be in this line executes the specified requirements. gawk sequentially processed each line of input to the document until the end of the input.

gawk program constituted by a lot of pattern or action, action written in curly braces {} inside, a pattern followed by an action. Program structure like this:

pattern {action}

pattern {action}

pattern {action}

....

The program provides in gawk, pattern or action can be omitted, but the two parts cannot be omitted.

5.5.2 Gnuplot

gnuplot is a free drawing tools, it can be ported to all major platforms. It can operate in one of two modes: You can operate the tool when you need to make adjustments and modifications chart displayed, through gnuplot commands entry into interactive mode. Or gnuplot can read commands from a file, generate charts in batch mode. Through the shell command prompt, type gnuplot command to start the program. gnuplot has two modes, interactive mode or batch mode. At the command line, the direct input gnuplot is an interactive way. When you want to exit gnuplot, simply use the quit command or the exit command. [45]

After the command "gnuplot" with the file name argument, gnuplot enters batch mode. In batch mode, gnuplot function is used in order to load all the files transferred to process and execute the command, and then exit automatically. The same commands in the file as use interactive mode command. For example, gnuplot will be executed in order of command file1 and file2:

gnuplot file1 file2

Gnuplot plot command is the main plot and splot, which plot is to draw two-dimensional graphics, splot is paqiinted 3D graphics. You can display a variety of settings in the graph, including graphic titles, horizontal and vertical coordinate range, coordinate scale, tagging shows and axis labels.

5.6 AODV+ implementation

The AODV+ implementation based on the AODV routing protocol structure. Assume that NS2 is already in our Ubuntu system. We need to move out all the original AODV protocol file and put the AODV+ file in. To achieve this we need to use the Linux command "sudo nautilus" to get the root authority which can modify any Linux system file. So the original AODV file will not be overwritten by your AODV+ file.

Secondly, we need to add some line for the NS2 agent, which make the agent understand our AODV+ protocol.

```
#My
modification***************************************************************//
#Modifications needed for aodv+!
#Reactive gateway discovery method is used by default.
Agent/AODV set gw_discovery 2
#******************************************************************************
*//
```

Finally just type command "make" for compile the protocol in the NS2.

With AODV+ a mobile node can find a route to a gateway in three different ways:

1. proactive gateway discovery method
2. hybrid gateway discovery method
3. reactive gateway discovery method (default)

If you want to use:

1. PROACTIVE GATEWAY DISCOVERY METHOD

    * Type "Agent/AODV set gw_discovery 0" before the creation of the nodes in
    * your Tcl file.

    Configurable parameters: ADVERTISEMENT_INTERVAL (aodv.h)

2. HYBRID GATEWAY DISCOVERY METHOD

    * Type "Agent/AODV set gw_discovery 1" before the creation of the nodes in
    * your Tcl file.

    Configurable parameters: ADVERTISEMENT_INTERVAL & ADVERTISEMENT_ZONE (aodv.h)

3. REACTIVE GATEWAY DISCOVERY METHOD

* Type "Agent/AODV set gw_discovery 2" before the creation of the nodes in

* your Tcl file.

Every time we change the file, we need to type command "make" once, to make the NS2 compile the protocol.

6. Evaluation

6.1 Simulation pattern setting

The main purpose of the simulation environment setting is to research the four protocols mobile performance and load bearing properties. There are two conditions for compare: change the number of the source node and the nodes pause time. Each node according to the speed of a randomly selected starting from a random position moves towards a random destination node. Destination location, pause for some time, and continues to move another random destination, were investigated 20,30 a source node load conditions.

Platform: NS2.35, MAC layer protocol IEEE802.11

Wireless transmission model: TwoRayGround

Queue types are: AODV as Queue / DropTail / PriQueue

DSR is CMUPriQueue

Simulation environment settings: Using NS2 own scenario generation script generation and CBR streams, such as the state of motion

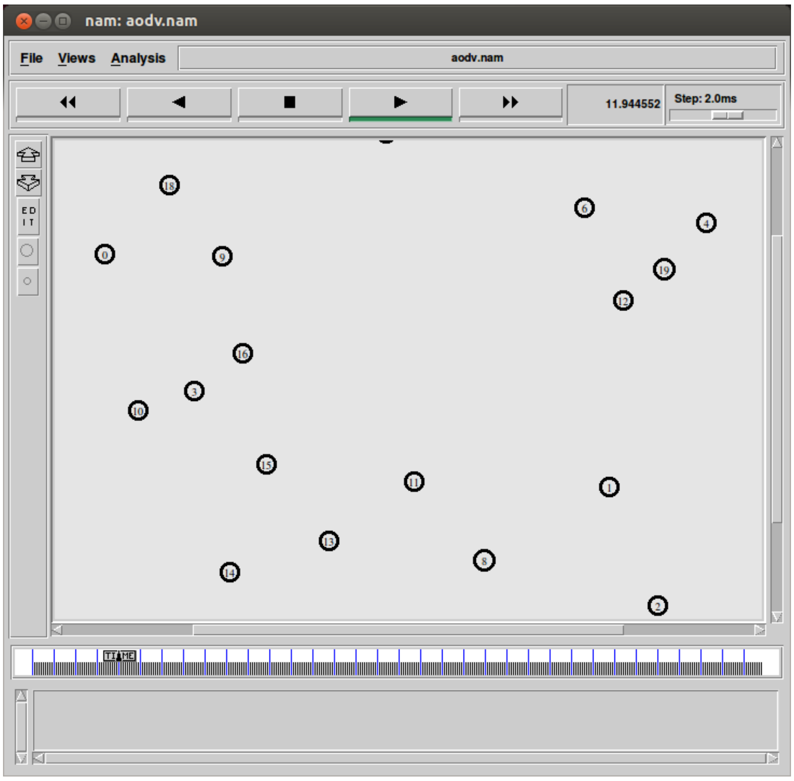
Example: ns cbrgen.tcl -type cbr -nn 20 -mc 3

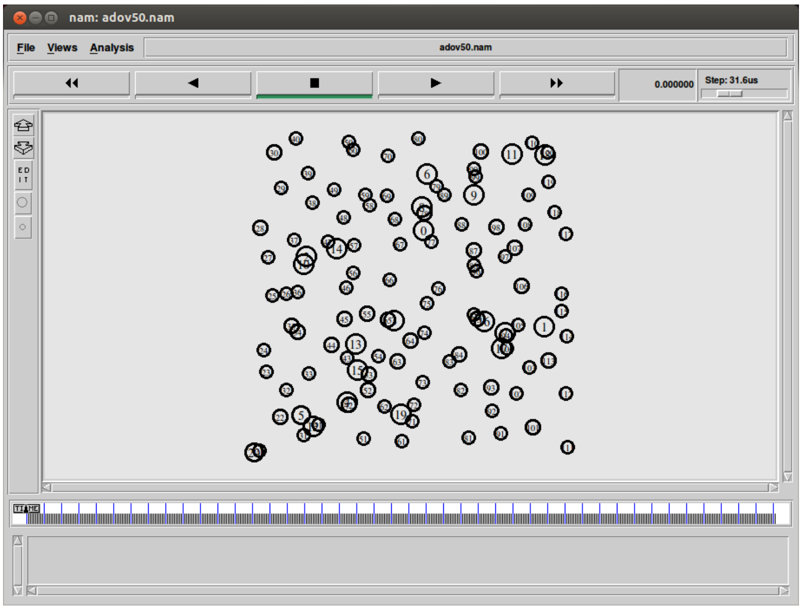
. / setdest-v1-n50-p $j-M6.0-tu900-x400-y400;

In setdest $i, $j is variable parameters, sports scenes of 400m × 400m square region, the node transmission distance of 250m, the maximum moving speed of 6m / s constant bit stream data is cbr, two data packets sent per second, packet length is 512 bytes, the simulation time is 900s.

Simulation process: As the volume of data, the simulation for a long time, in order to ensure accurate data, this project individually for each data input code comparison. Simulation results represents model is exactly same as each data point, but different randomly generated simulation results under the program moving. Use exactly the same for each routing protocol test solutions and test program.

Picture below is the screenshot of 20 and 100 nodes in aodv mode.





6.2 Simulation results and analysis

6.2.1 Data packet transmission delay

Shown in Figure 1, with the increase of the source node, AODV and DSR delay has increased. Mobility of the DSR greater impact than the AODV, AODV reflected in the whole process of moving relatively stable; DSR in the process delays remain at the top of AODV. DSR is also same as AODV and DSDV when 20 source node, but at 50 source node pause time from 60s to 100s delay increased, we can see that this value is already quite large. Performance of DSR first increases and then decreases with increasing number of nodes. Delay of DSDV network increased with increasing nodes same as AODV, it varies with increasing number of nodes, but compared to three figures AODV is most appropriate.

Figure 1 (20 nodes of the delay in AODV, DSDV, DSR)

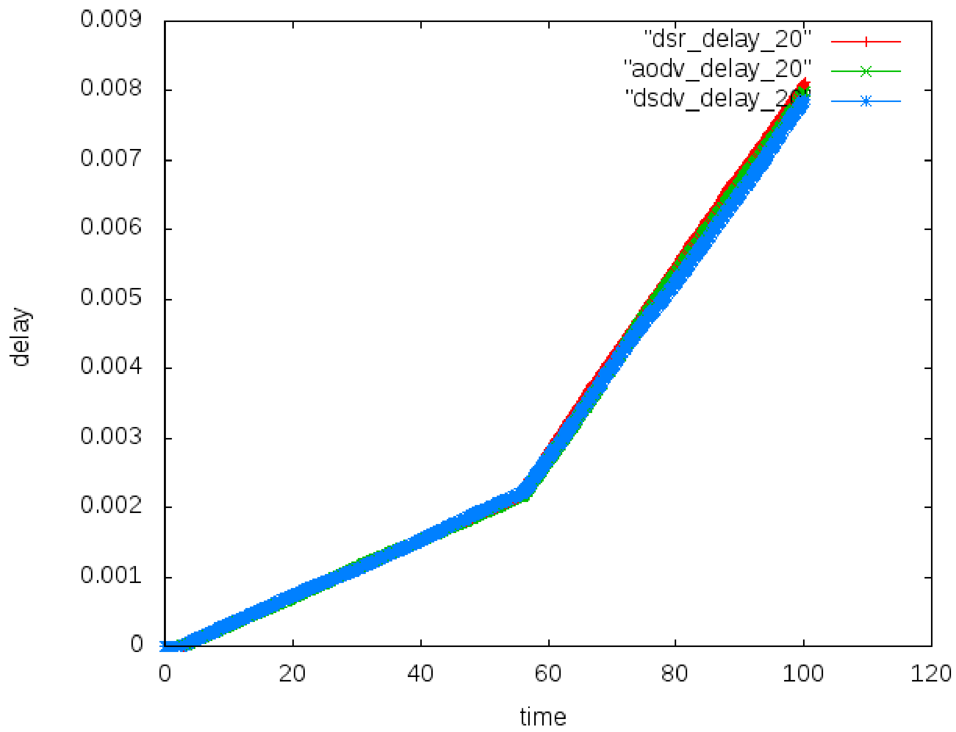


Figure 2 (50 nodes of the delay in AODV, DSDV, DSR)

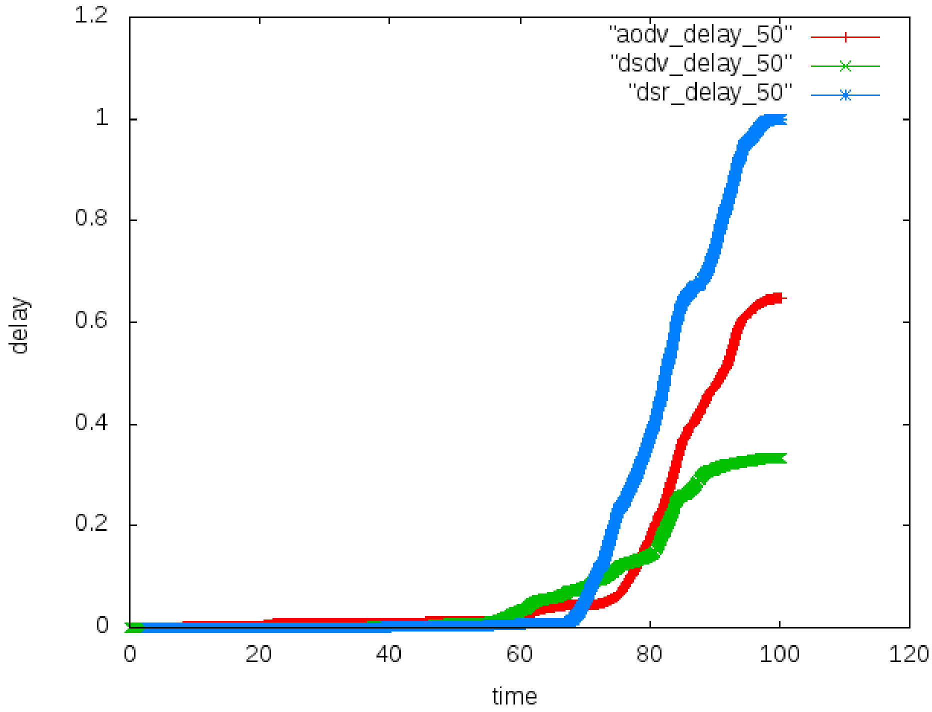
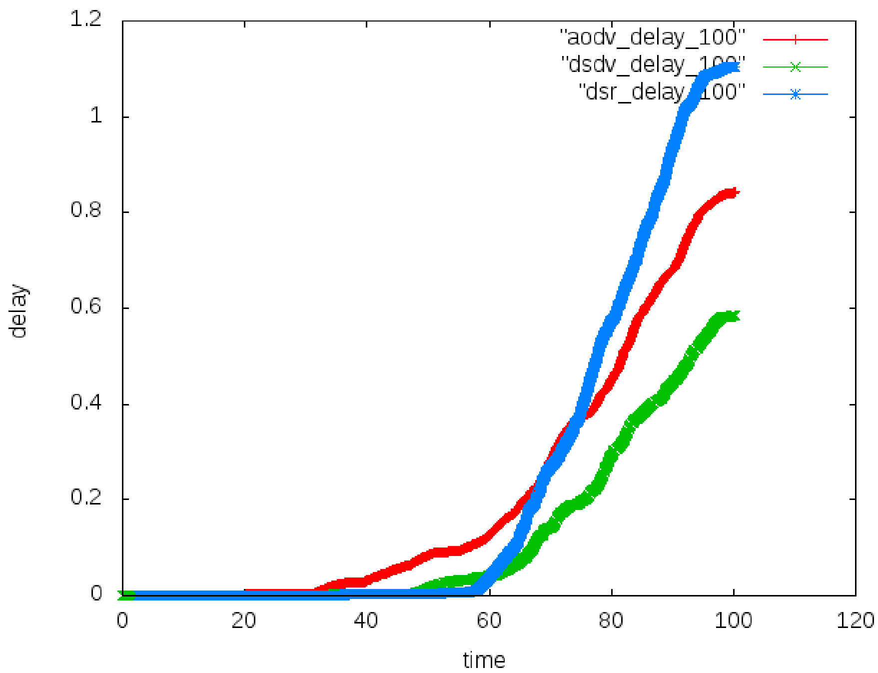


Figure 3 (100 nodes of the delay in AODV, DSDV, DSR)



6.2.2 Packet Loss

Shown as figure 4 and figure 5, With increasing number of nodes, the packet loss of AODV varies with increasing number of nodes, AODV show worst-performance, DSDV performs consistently and packet loss increases for DSR. When we use 20 nodes to test the packet loss on the link, we found the effect is not obvious, thus we choose 50 nodes and 100 nodes to compared.

Figure 4 (50 nodes of the Packet Loss in AODV, DSDV, DSR)

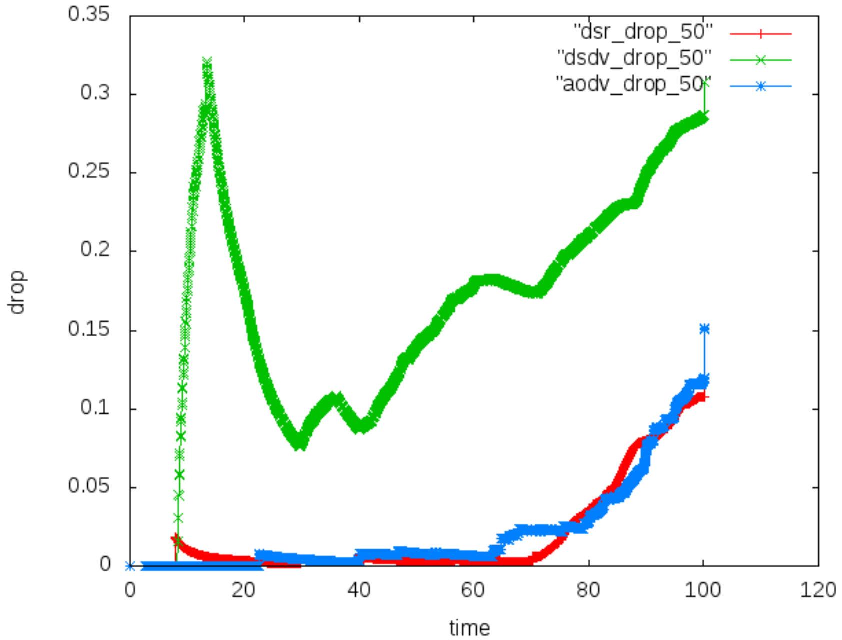
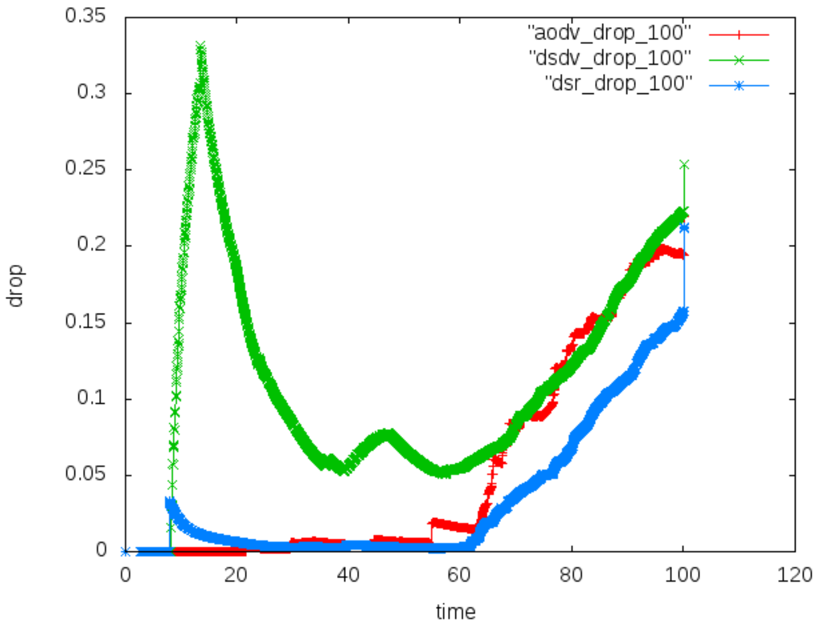


Figure 5 (100 nodes of the Packet Loss in AODV, DSDV, DSR)



6.2.3 Throughput

AODV and DSR perform same for less number of nodes, which decreases with increasing nodes for DSR network. Overall, DSDV shows highest throughput and outperforms the other protocols.

Figure 6 (20 nodes of the throughput in AODV, DSDV, DSR)

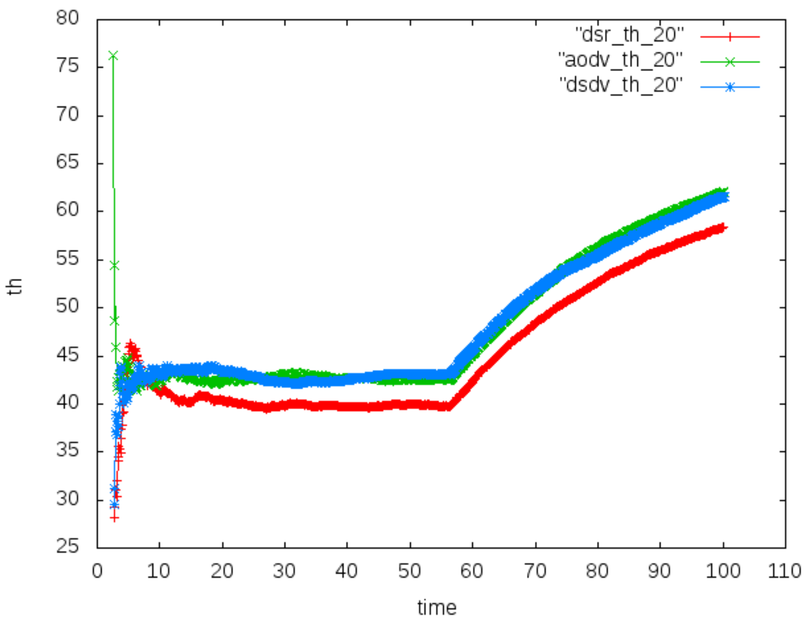


Figure 7 (50 nodes of the throughput in AODV, DSDV, DSR)

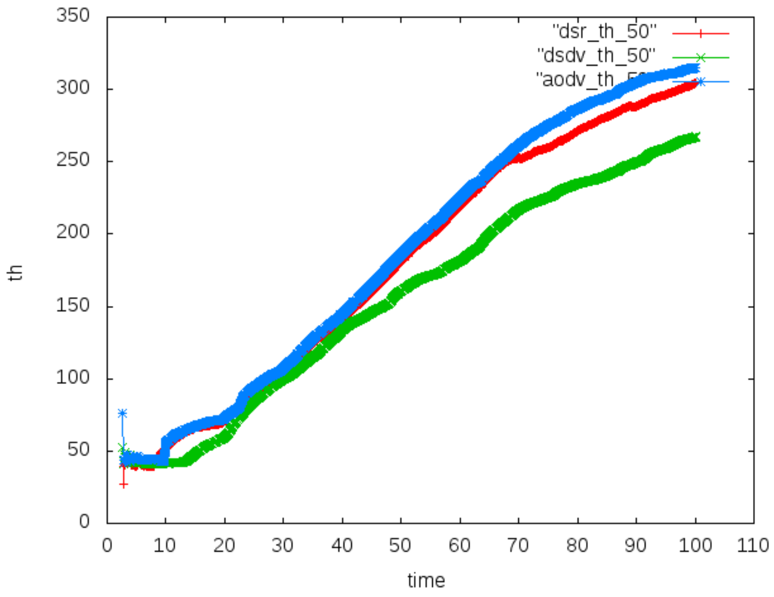
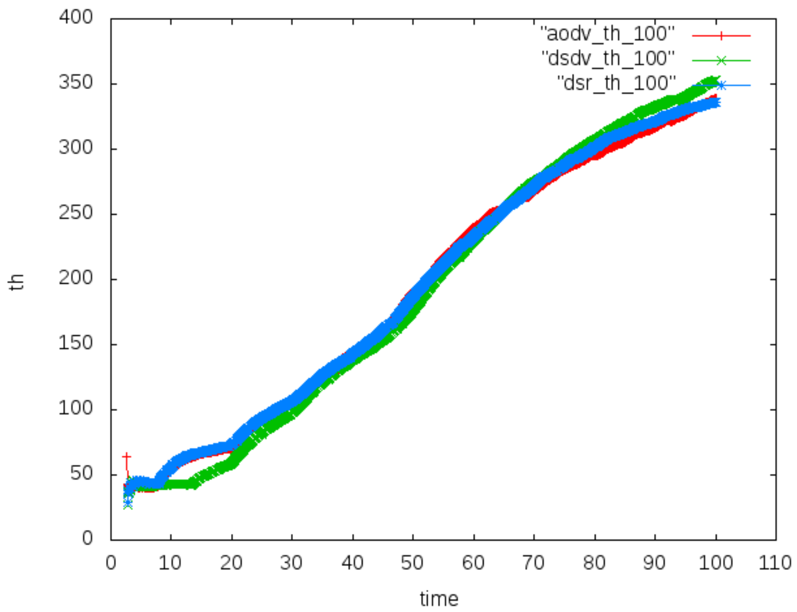


Figure 8 (100 nodes of the throughput in AODV, DSDV, DSR)

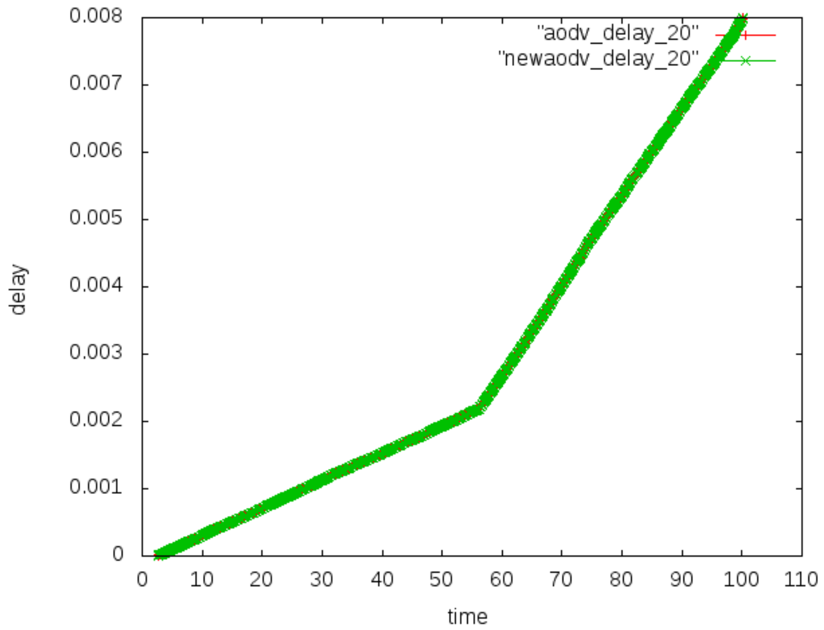


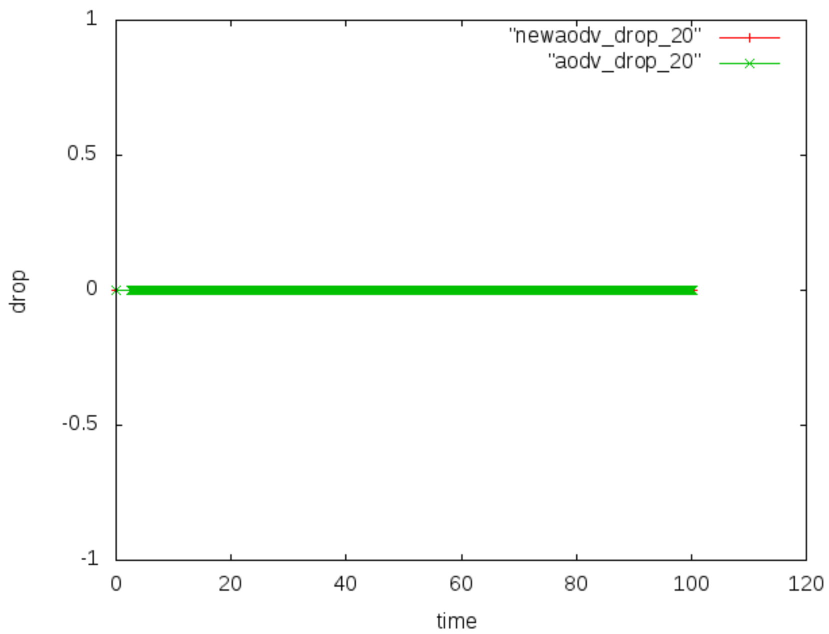
6.2.4 For AODV+

The performance of the AODV+ compare with AODV at 20 nodes are almost same, shown in figure 9 below. However, as nodes increased, AODV+ in the three performance test with AODV gradually widen the gap, easy to find from the figure 10, The performance of AODV still better than AODV+, AODV is more stable. While from

AODV+ research and improvement did not shown as expected, but we still gained a lot of valuable knowledge from research and programming.

Figure 9 (20 nodes of the delay, packet loss, throughput in AODV+)





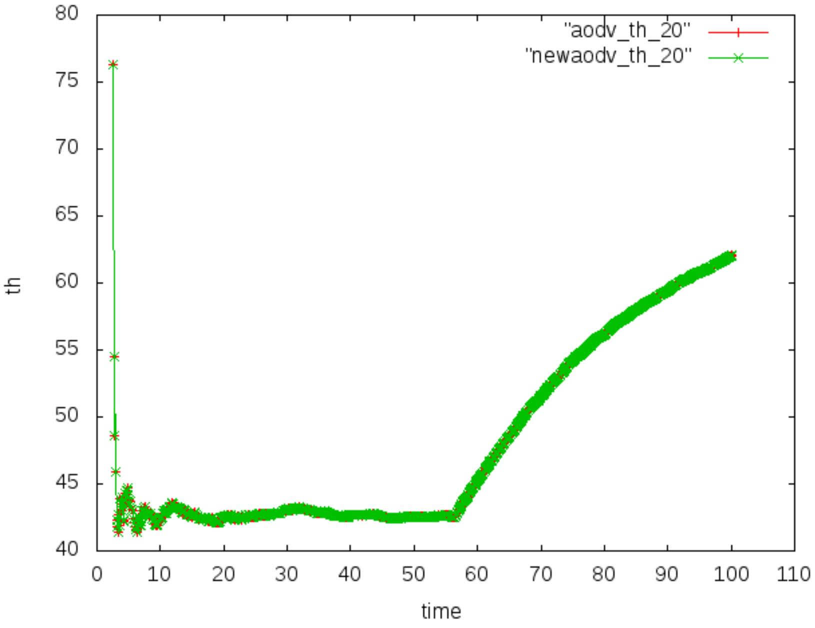
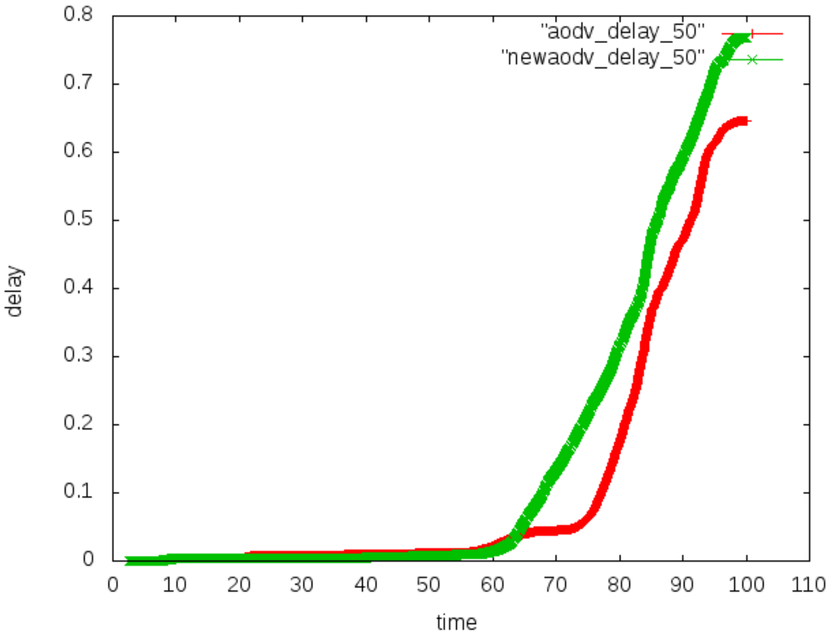
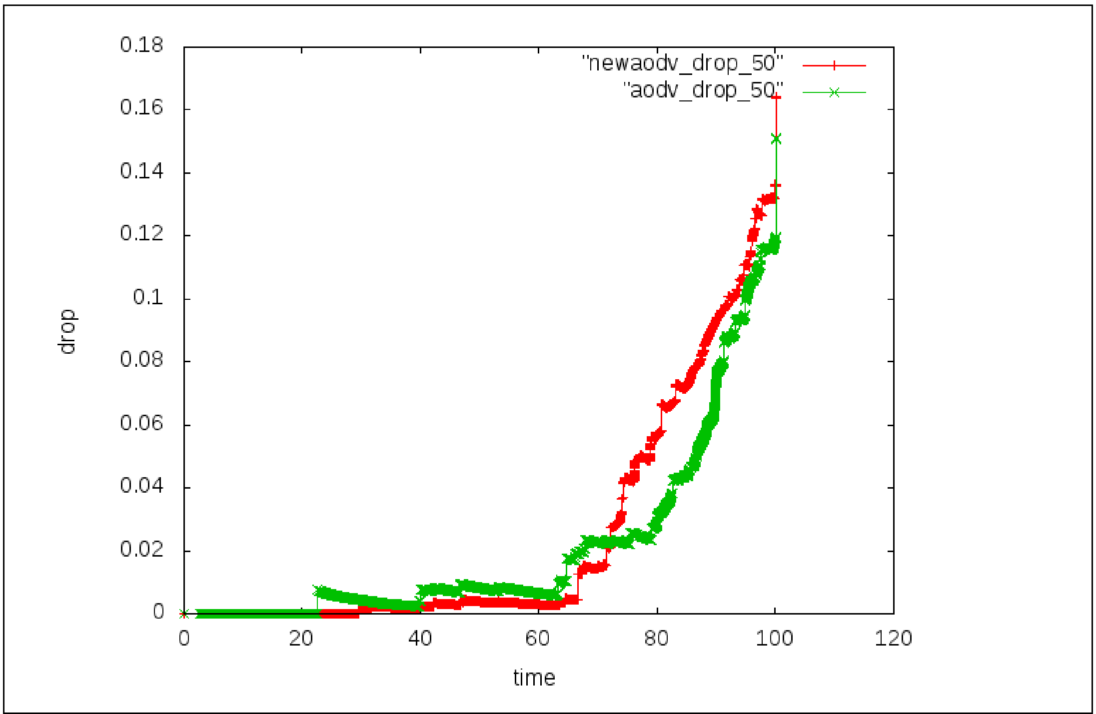
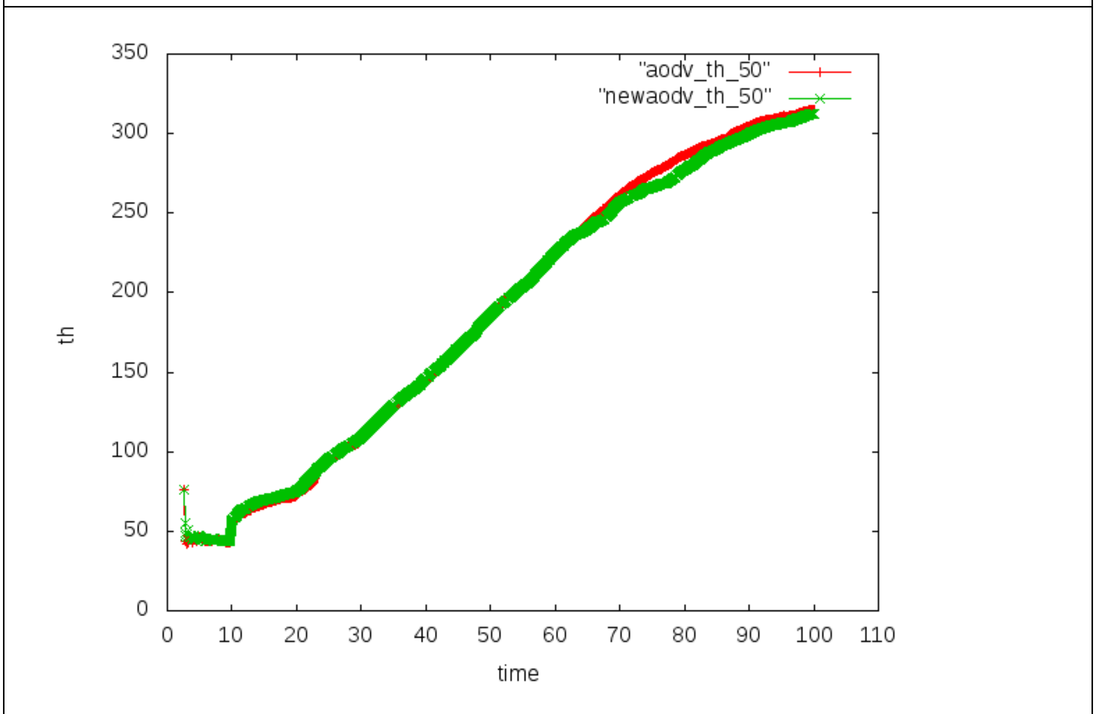


Figure 9 (50 nodes of the delay, packet loss, throughput in AODV+)







6.3 Discussion

This project shows the realistic comparisons of MANET protocols which are both reactive and proactive, also present the simulation results based on theoretical analysis.

Follow by the number of the nodes varied, from the scenario we can see all three protocols show same delay for small number of nodes, but the delay decreases with increasing nodes for DSDV network. For packet Loss, 3 protocols remain same for less number of nodes, but with increasing nodes DSR show maximum packet loss. Performance of AODV packet delivery ratio remains constant for increasing number of nodes, whereas for DSDV the packet delivery ration is more than DSR. The performance of AODV, DSDV and DSR remains almost constant for increasing number of nodes but AODV and DSR shows better than DSDV.

Through ad-hoc networks simulation analysis can be seen even the number of the links increased, but the network link performance does not appear significantly decreased. Each link the through output packet rate remain at 99%, the number of the packet loss is minimal. Because DSR is demand routing protocols, only send data when it is set up the link, moreover each node contains a retransmission buffer to store the data which need to send to next node, if next node does not receive the transmitted response packet and ACK packet, the node will retransmission the data until the number exceeds a certain amount, then discarded. Another reason is the TCP protocol is a connection-oriented , reliable transport protocol , to ensure a high rate of investment package.

Analysis of network end to end delay can see some of the links appear to have a longer portion of the data packet delay , which is mainly caused by two reasons , the first one is the source node to the destination node without routing , you need to initiate route discovery process cause a longer delay , another case of network nodes in constant motion so that the source node to the destination node's link appears broken, need to re-initiate route discovery process caused long delays. Calculating the average delay of the link can be seen in the average delay of the link is very small, no larger jitter. End three cases the overall delay is very short.

Some nodes in the network is very high number of packets forwarded by the nam animation observation can be found. These nodes are intermediate node links, the link for forwarding the data, thus the number of these nodes loss more than other nodes. While some nodes in the network is far from the source node and the destination node, so there is no access to the link to transmit packets, so the number of packets forwarded from these nodes is zero and no packet loss occurs.

With the increase in the number of links, some nodes of packet loss and number of forwarding packets also follow the increase, which was mainly due to some nodes in several different links, they forwarded packet data to several links, therefore packet loss number and the number of packets transmitted along with the increase of the number of links. Compare and analysis with these cases, we found when the number of links increased, some link send packet less than before, but the newly added link still keep high amount of data sent.

7. Conclusion and Future Work

7.1 Conclusions

From the last chapter analysis, we can found overall AODV shows the best performance with its ability to maintain connection by periodic exchange of information required for TCP network. AODV performs best in case of packet delivery ratio and DSDV outperform others in case of throughput. AODV better than DSDV and DSR as in high mobility topology change rapidly and AODV can adapt to the changes , but compare everything into account, DSDV is better than others. At higher node mobility, AODV is worst in case of packet loss and throughput but performs best for packet delivery ratio. DSDV performs better than AODV for higher node mobility, in case of end-to-end and throughput but DSR performs best in case of packet loss.

Hence, for real time traffic DSDV is preferred over DSR and AODV. Finally, from the above research work performance of AODV is considered best for Real-time and TCP network. Also the performance of AODV still better than AODV+, AODV is more stable. While from AODV+ research and improvement did not shown as expected, but we still gained a lot of valuable knowledge from research and programming.

7.2 Future Work

Until today, WSN routing protocol classification still have no uniform standards, therefore there are a large number of proposed routing protocols. The researchers used a variety of strategies to design routing protocols, some of the good protocols with the following characteristics: efficient use of energy almost is the first rule of WSN protocol design. Through the research in WSNs routing protocol, we can analysis and summarize the result that WSN strategies and development trend will be seen:

1) Reduce the traffic in order to save energy. Data communication in WSN is the

main area of energy consumption; therefore the WSN protocols should minimize the amount of data communication. For example, query data or data reporting using a filtering mechanism, inhibition of node upload unnecessary data; Using data combine mechanism to complete the calculate before the data arrive to the sink.

2) To keep the transmission node energy balancing through more flexible use of routing strategies, Balance the residual energy of the nodes to improve the survival time of the entire network. For example, in the hierarchical routing using dynamic cluster head; Choose to use random routing rather than stable routing; consider the remaining energy of the node in the path selection.

3) 3. Routing protocols should have fault tolerance. Due to the WSNs node prone always have problems, so we should let most of the node to get available network information to calculate routing, to ensure that the routing failure can be restored as soon as possible; we also need to improve data transmission reliability using multi-path transmission.

4) 4. Routing protocols should have security mechanisms. Because the inherent characteristics of WSNs routing protocols vulnerable to security threats especially in military applications. Current routing protocols less consider the security issues. Therefore, when we design the application must consider the safety mechanism routing protocol.

5) 5. Support for mobility. Compared to the Ad hoc network routing, due to resource constraints the current WSN routing protocols topology awareness and mobility are relatively poor. Many routing protocols adapt to a few topology change occasions. How to control protocol consumption to support rapid topology-aware, is an important challenge.

6) 6. IPv6 combined with WSN. The core of the next-generation network is IPv6 protocol, WSN will work closely together with IPv6 in the future. There are two tasks need to achieve, on the one hand to achieve the WSN and IPv6 protocol network interconnection; On the other hand, WSN network nodes work in IPv6 applications.

8. Reference

[1] Opnet is now a part of Riverbed Technology. [Online] Available: http://www.opnet.com/

[2] nsnam web pages. [Online] Available: http://www.isi.edu/nsnam/

[3] QualNet [Online] Available: http://www.qualnet.com/

[4] Basagni, S., Conti, M., Giordano, S. &Stojmenovic, I. 2013, Mobile Ad Hoc Networking: The Cutting Edge Directions, Wiley-IEEE Press

[5]Anand, D.K. &Prakash, S. 2010, "A Short Survey of Energy-Efficient Routing Protocols for Mobile Ad-Hoc Networks", Advances in Recent Technologies in Communication and Computing (ARTCom), 2010 International Conference on IEEE, pp. 327.

[6] JoannaKulik, Wendi Heinzelman,HariBalakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks",

From : http://nms.csail.mit.edu/papers/index.php?detail=85

[7] W.R. Heinzelman et al., Energy-scalable algorithms and protocols for wireless sensor networks, in: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00), Istanbul, Turkey, June 2000

[8] R. Min et al. architecture for a power aware distributed microsensor node, in: Proceedings of the IEEE Workshop on signal processing systems (SIPS'00), October 2000

[9] Kemal Akkaya, Mohamed Younis "A survey on routing protocols for wireless sensor networks", from http://www.sciencedirect.com/science/article/pii/S1570870503000738?np=y

[10] W. Heinzelman, J. Kulik, H. Balakrishnan, Adaptive protocols for information dissemination in wireless sensor networks, in: Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, August 1999

[11] C. Intanagonwiwat, R. Govindan, D. Estrin, Directed diffusion: a scalable and robust communication paradigm for sensor networks, in: Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'00), Boston, MA, August 2000

[12] D. Braginsky, D. Estrin, Rumor routing algorithm for sensor networks, in: Proceedings of the First Workshop on Sensor Networks and Applications (WSNA), Atlanta, GA, October

2002

[13] C. Schurgers, M.B. Srivastava, Energy efficient routing in wireless sensor networks, in: The MILCOM Proceedings on Communications for Network-Centric Operations: Creating the Information Force, McLean, VA, 2001

[14] A. Manjeshwar, D.P. Agrawal, TEEN: a protocol for enhanced efficiency in wireless sensor networks, in: Proceedings of the 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001

[15] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robustcommunication paradigm for sensor networks", Proceedings ACM MobiCom'00, Boston, MA, Aug. 2000, pp. 56-67.

[16] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed diffusion forwireless sensor networking", IEEE/ACM Transactions on Networking, vol. 11., no. 1, Feb. 2003, pp. 2-16

[17] B. Krishnamachari, D. Estrin, S. Wicker, Modeling data centric routing in wireless sensor networks, in: Proceedings of IEEE INFOCOM, New York, June 2002

[18]D. Ganesan et al.Highly resilient, energy efficient multipath routing in wireless sensor networks, Mobile Computing and Communications Review, 5 (4) (2002), pp. 11–25

[19] Braginsky D, Estrin D. Rumor routing algorithm for sensor networks. In: Proc. of the 1st workshop on sensor networks and applications. Atlanta: ACM Press, 2002. 22−31.

[20] S.K. Singh, M.P. Singh, and D.K. Singh, “A survey of Energy-Efficient Hierarchical Cluster-basedRouting in Wireless Sensor Networks”, International Journal of Advanced Networking andApplication (IJANA), Sept.–Oct. 2010, vol. 02, issue 02, pp. 570–580.

[21] D. B Johnson et al., “Dynamic Source Routing in Ad Hoc Wireless Networks”, in Mobile Computing,edited by Tomas Imielinski and Hank Korth, Kluwer Academic Publishers, ISBN: 0792396979, 1996,Chapter 5, pp. 153-181.

[22] W.R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient Communication Protocolfor Wireless Microsensor Networks”, in IEEE Computer Society Proceedings of the Thirty ThirdHawaii International Conference on System Sciences (HICSS '00), Washington, DC, USA, Jan. 2000,vol. 8, pp. 8020.

[23] S. Lindsey, C. Raghavendra, PEGASIS: Power-Efficient Gathering in Sensor Information Systems, IEEE Aerospace Conference Proceedings, (2002), Vol. 3, 9-16 ISSN:1790-5117,pp. 1125-1130.

[24] A. Manjeshwar, D.P. Agrawal, TEEN: a protocol for enhanced efficiency in wireless sensor networks, in: Proceedings of the 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001

[25] A. Manjeshwar, D.P. Agrawal, APTEEN: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks, in: Proceedings of the 2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile computing, Ft. Lauderdale, FL, April 2002

[26] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for wireless sensor networks, in: Proceeding of the Hawaii International Conference System Sciences, Hawaii, and January 2000

[27] Y Xu, J Heidemann, D Estrin. Geography Informed Energy Conservationfor Ad hoc Routing[C] .Proceedings of the 7 th Annual ACM/IEEE International Conference on Mobile Computing and Networking, 2001. 70 - 84.

[28] Y Yu, D Estrin, R Govindan. Geographical and Energy Aware Routing:A Recursive Data Dissemination Protocol for Wireless SensorNetworks[R].UCLA Computer Science Department, 2001, 1-11.

[29]V Rodoplu, T H Meng. Minimum Energy Mobile Wireless Networks [J], IEEE Journal Selected Areas in Communications, 1999, 17(8): 1333 -1344.

[30] Fasee Ullah, Muhammad Amin and Hamid ul Ghaffar, “Simulating AODV and DSDV For Adynamic Wireless Sensor Networks” IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.7, July 2010.

[31] Asma Tuteja , Rajneesh Gujrl and Sunil Thalia, “Comparative Performance Analysis of DSDV, AODV and DSR Routing Protocols in MANET using NS2”, International Conference on Advances in Computer Engineering, 2010.

[32] Sapna S. Kaushik1 & P.R.Deshmukh2, “Comparison of effectiveness of AODV, DSDV and DSR routing protocols in Mobile Ad-Hoc Network” International Journal of Information

Technology and Knowledge Management , Volume 2, No. 2, July-December 2009.

[33] P. Udayakumar and Asha Ambhaikar, “Experimenatal comparison of AODV and DSR protocols”, International J. of Engg. Research & Indu. Appls. (IJERIA) ISSN 0974- 1518, Vol.2, No. III , 2009

[34] Nor Surayati Mohamad Usop, Azizol Abdullah and Ahmad Faisal Amri Abidin, “Performance Evaluation of AODV, DSDV & DSR Routing Protocol in Grid Environment”, IJCSNS International Journal of Computer Science and Network Security, VOL.9 NO.7, July 2009.

[35] David Oliver Jörg, “Performance Comparison Of MANET Routing Protocols In Different Network Sizes”, University of Berne, Switzerland, 2003.

[36] Shikhar Kr. Sarma, Sumita Dhar Choudhury, Fakhar Uddin Ahmed, & Abdul Hannan, “Efficiency Analysis of Routing Protocols for Mobile Ad Hoc Networks for Heterogeneity”, International

[37] Marc Greis, “Tutorial for the network simulator NS”, http://www.scribd.com/doc/13072517/Tutorial-Ns- Full-by-MARC-GREIS

[38] COMPARATIVE ANALYSIS OF AOMDV, AODV, DSR AND DSDV ROUTINGPROTOCOLS FOR COGNITIVE RADIO, http://www.academia.edu/3311606/Comparative_Analysis_of_AOMDV_AODV_DSR_and_DSDV_Routing_Protocols_for_Cognitive_Radio

[39] A. Hamidian, “A study of internet connectivity for mobile ad hoc networks in ns 2,” Master’s thesis, Departement of Communication Systems, Lund Institute of Technology, Lund University, Sweden, January 2003.

[40] R. Wakikawa, J. Malinen, C. Perkins, A. Nilsson, and A. Tuominen, “Global connectivity for ipv6 mobile ad hoc networks,” IETF Internet Draft, November 2001.

[41] S. McCanne and S. Floyd. (2004, September) ns2 network simulator 2. [Online]. Available: http://www.isi.edu/nsnam/ns

[42] The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4, http://www.ietf.org/rfc/rfc4728.txt

[43] Ad hoc On-Demand Distance Vector (AODV) Routing, http://www.ietf.org/rfc/rfc3561.txt

[44] Tutorial for the NS by Marc Greis. Currently the tutorial is maintained and being expanded by the VINT group, http://www.isi.edu/nsnam/ns/tutorial/

[45]Leiming Xu，Bo Pan，Yao Zhao. NS and network simulation.

[46] [6] Kevin Fall,Kannan Varadhan.The ns Manual.The VINT Project. [EB/OL]. http://www.mash.Cs. berkeley.edu/ns,2007-2-1/2007-4-3.

[47] XI. Generating node-movement and traffic-connection files for large wireless scenarios, http://www.isi.edu/nsnam/ns/tutorial/nsscript7.html.

[48] S. Corson and J. Macker , “Mobile Ad hoc Networking (MANET):Routing Protocol Performance Issues and Evaluation Considerations” , University of Maryland, January 1999.

9. Appendices

9.1 Appendix 1 (TCL Script)

```
set val(chan)        Channel/WirelessChannel ;# channel type

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

set val(netif)       Phy/WirelessPhy ;

set val(mac)         Mac/802_11 ;

set val(ifq)         Queue/DropTail/PriQueue ;

set val(ll)     LL ;

set val(ant)         Antenna/OmniAntenna ;

set val(ifqlen)      50 ;

set val(nn)          20 ;

set val(x)                400

set val(y)                400

set val(rp)          DSDV;# routing protocol

set val(cp)          "cbr20" ;

set val(sc)          "scence20" ;


set ns_ [new Simulator]

set tracefd [open dsdv20.tr w]

$ns_ trace-all $tracefd


set namtracefd [open dsdv50.nam w]

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


set topo [new Topography]

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


set god_ [new God]
```

```
create-god $val(nn)

set chan_1_ [new $val(chan)]

$ns_ node-config -adhocRouting $val(rp) \
        -llType $val(ll) \
        -macType $val(mac) \
        -ifqType $val(ifq) \
        -ifqLen $val(ifqlen) \
        -antType $val(ant) \
        -propType $val(prop) \
        -phyType $val(netif) \
                -channel $chan_1_    \
        -topoInstance $topo \
        -agentTrace ON \
        -routerTrace ON \
        -macTrace OFF \
        -movementTrace OFF \

for {set i 0} {$i < $val(nn)} {incr i} {
set node_($i) [$ns_ node]
$node_($i) random-motion 0
}

source $val(cp)
source $val(sc)

for {set i 0} {$i < $val(nn)} {incr i} {
$ns_ at 100.1 "$node_($i) reset";
}
```

```
$ns_ at 100.2 "stop"

$ns_ at 100.3 "puts \"NS exiting...-\"; $ns_ halt"

proc stop {} {

global ns_ tracefd namtracefd

$ns_ flush-trace


close $tracefd

close $namtracefd


#exec nam aodv.nam &

exit 0

}

$ns_ run
```

9.2 Appendix 2 (CBR scenario)

```
#
# nodes: 20, max conn: 3, send rate: 0.10000000000000001, seed: 1
#
#
# 1 connecting to 2 at time 2.5568388786897245
#
set udp_(0) [new Agent/UDP]
$ns_ attach-agent $node_(1) $udp_(0)
set null_(0) [new Agent/Null]
$ns_ attach-agent $node_(2) $null_(0)
set cbr_(0) [new Application/Traffic/CBR]
$cbr_(0) set packetSize_ 512
$cbr_(0) set interval_ 0.10000000000000001
$cbr_(0) set random_ 1
$cbr_(0) set maxpkts_ 10000
$cbr_(0) attach-agent $udp_(0)
$ns_ connect $udp_(0) $null_(0)
$ns_ at 2.5568388786897245 "$cbr_(0) start"
#
# 4 connecting to 5 at time 56.333118917575632
#
set udp_(1) [new Agent/UDP]
$ns_ attach-agent $node_(4) $udp_(1)
set null_(1) [new Agent/Null]
$ns_ attach-agent $node_(5) $null_(1)
set cbr_(1) [new Application/Traffic/CBR]
$cbr_(1) set packetSize_ 512
$cbr_(1) set interval_ 0.10000000000000001
```

```
$cbr_(1) set random_ 1
$cbr_(1) set maxpkts_ 10000
$cbr_(1) attach-agent $udp_(1)
$ns_ connect $udp_(1) $null_(1)
$ns_ at 56.333118917575632 "$cbr_(1) start"
#
# 4 connecting to 6 at time 146.96568928983328
#
set udp_(2) [new Agent/UDP]
$ns_ attach-agent $node_(4) $udp_(2)
set null_(2) [new Agent/Null]
$ns_ attach-agent $node_(6) $null_(2)
set cbr_(2) [new Application/Traffic/CBR]
$cbr_(2) set packetSize_ 512
$cbr_(2) set interval_ 0.10000000000000001
$cbr_(2) set random_ 1
$cbr_(2) set maxpkts_ 10000
$cbr_(2) attach-agent $udp_(2)
$ns_ connect $udp_(2) $null_(2)
$ns_ at 146.96568928983328 "$cbr_(2) start"
#
#Total sources/connections: 2/3
#
```

9.3 Appendix 3 (User Manual)

This project include Tcl script，Awk，CBR，scenario and gnuplot all program.

When you in to the Linux system,
First step: you need to make the Tcl script to process the trace file.
Here just some sample:
ns aodv50.tcl
ns dsdv50.tcl
ns dsr50.tcl

It will work out 3 result: aodv50.tr, dsdv50.tr, dsr50.tr

Second step: though the Awk program get the data from trace file.
Command:

```
awk -f delay.awk aodv50.tr > aodv_delay_50
awk -f th.awk aodv50.tr > aodv_th_50
awk -f deliver.awk aodv50.tr > aodv_deliver_50
awk -f drop.awk aodv50.tr > aodv_drop_50

awk -f delay.awk dsdv50.tr > dsdv_delay_50
awk -f th.awk dsdv50.tr > dsdv_th_50
awk -f deliver.awk dsdv50.tr > dsdv_deliver_50
awk -f drop.awk dsdv50.tr > dsdv_drop_50

awk -f delay.awk dsr50.tr > dsr_delay_50
awk -f th.awk dsr50.tr > dsr_th_50
awk -f deliver.awk dsr50.tr > dsr_deliver_50
awk -f drop.awk dsr50.tr > dsr_drop_50
```

Here get the data: Delay, Through output, Delivery (packet delivery ratio), Drop (packet drop ratio)

(1) **Delay** ： Packet receive total delay / Number of the receive packet
(2) **Through output**： Number of the data / Simulation time
(3) **Packet drop ratio**： Number of the packet lost / Number of the packet sent
(4) **Delivery**: Number of the packets that destination node received / Number of the packets that Source node sent, reflects the reliability of the network transmission, higher ratio means higher reliability.

Third step: Use the data from awk to draw the graph.
Command:
set terminal gif
set output "dsr_delay.gif"

set xlabel "time"
set ylabel "delay"
plot "dsr_delay_50"with linespoint

set terminal Represents the output file suffix
set output Represents the name of the output image, the example used is *dsr_delay.gif*
set xlabel Set the name of the x-axis
set ylabel Set the name of the y-axis
plot "dsr_delay_50"with linespoint

Last sentence is based on information obtained from awk to draw the graph. In the example we use dsr_delay_50， the information is from *awk -f delay.awk dsr50.tr > dsr_delay_50*