We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Directional Routing Protocol in Wireless Mobile Ad Hoc Network

L.A.Latiff[1], N. Fisal[2], S.A. Arifin[3] and A. Ali Ahmed[4],
[1,2,3]Telematics Research Group, Universiti Teknologi Malaysia, Malaysia
Yemen University of Science and Technology, Yemen

1. Introduction

Advancement in wireless communication technology and portable computing devices such as wireless handhelds, Personal Digital Assistants (PDA) and other mobile information terminals have led to a revolutionary change in our information society towards the era of mobile computing. The ubiquitous access to a variety of digital devices and multimedia tools makes it possible to create, analyze, synthesize and communicate knowledge using a rich variety of media forms. Additionally, the mobile devices are getting smaller, cheaper, more convenient, and more powerful and have contributed to the explosive growth of the mobile computing equipment market. Vast interest and concerted work in developing and enhancing wireless and mobile network protocols are being driven by the ever increasing demand for an anytime and anywhere Internet access.

To date, the type of network that have been widely deployed is based on a centralized approach which requires a network point of access, commonly called the Access Point (AP) that act as a gateway for the mobile device to the Internet. Even though these infrastructure-based networks provide the path for mobile devices connectivity, time and potentially high cost are required to set up the necessary infrastructure (Xue & Ganz, 2003). Also due to the limited radio range, the devices must be in the vicinity of an AP in order to be connected. It is important to note that when a natural catastrophe, war, or geographic isolation occurs, communication may break down; thus, unavailability of the network connection (Milanovic et al.,2004). Hence, the provision of required connectivity and network services at this instance becomes a real challenge. With this scenario, ad hoc technology emerges with the aim to solve this problem.

1.1 MANET Features

Mobile Ad Hoc Network (MANET) is a large collection of mobile nodes which are equipped with wireless communication devices. These devices communicate peer-to-peer in a network with no fixed infrastructure even when moving. Nodes in MANET can serve as hosts and routers where communication between nodes beyond their transmission radius can be achieved via several hops as inherent in collaborative communication of

neighbouring nodes. Due to its portability, MANET nodes depend on batteries for their source of energy (Sun, 2001).

The wireless media used will remain to have a significantly lesser capacity compared to the wired media (Macker & Corson, 1997). Also, the communication media, which is shared with neighbours that are within communication range, warrants implementation of a multiple access protocol to support systematic and efficient sharing.

Each mobile device in MANET is an autonomous node (Macker & Corson, 1997) and this means that nodes not only perform basic processing abilities to initiate and receive data as a host, it also performs routing functions. The routing algorithm in MANET can be a single-hop or multihop which has different link layer attributes and routing protocols. Single-hop communication is simpler in terms of structure and implementations but has lesser functions and applications compared to multi-hop communication. In multi-hop communication, the destination is beyond the transmission coverage of the source and hence the packets are forwarded via one or more intermediate nodes. Figure 1 shows a MANET network consisting of nodes and their transmission ranges. As shown in Figure 1, Node 2 and Node 3 are neighbours of Node 1 whilst Node 4 and Node 5 are not. Therefore, data transmission to Node 4 and Node 5 will have to be relayed by Node 2.
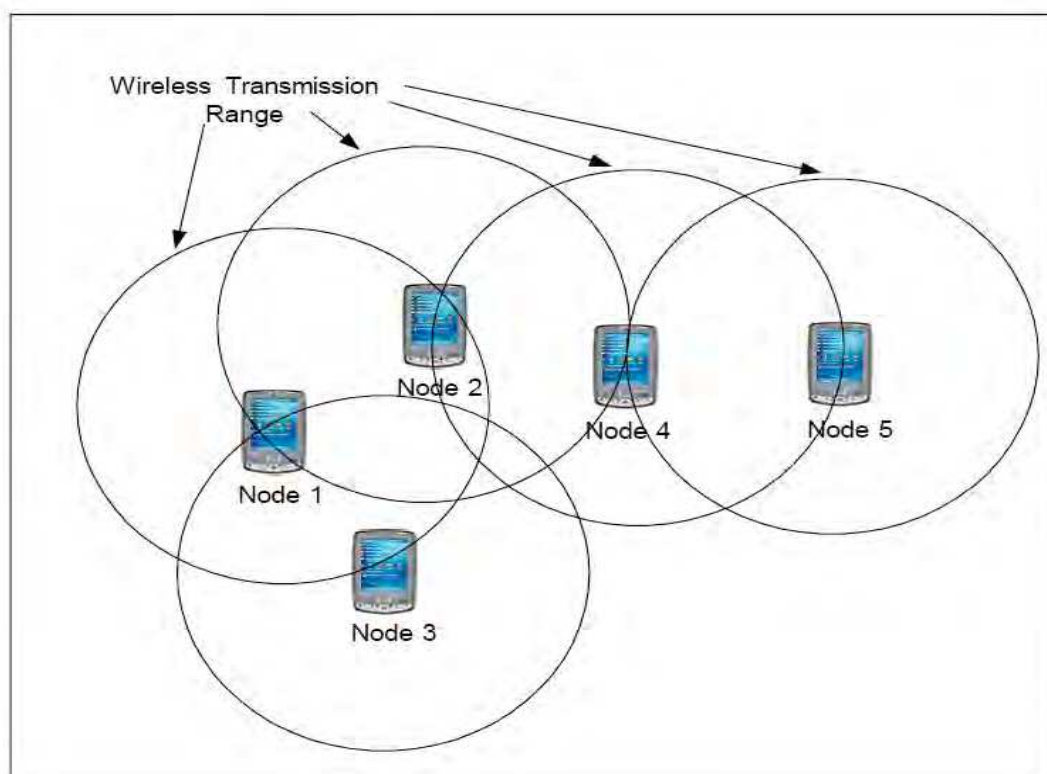

Fig. 1. MANET nodes and their transmission ranges

Since nodes in MANET use batteries as their source of energy which is easily depleted, it is required to extend the longevity of nodes by utilizing effective and energy-saving algorithm.

In MANET, nodes are free to move randomly. Hence, the topology of the network is ever changing and the communication link created between a source and destination pair may vary with time (Agrawal & Zeng, 2003). Nodes must be able to establish and maintain the routes as they move to allow applications services to operate without interruptions. Therefore, the control and management of the network must be distributed among the mobile nodes to support multi-hop communication beyond the transmission range of a node.

These characteristics do pose challenges to the roll-off of MANET which have motivated a concerted time and effort of researchers in proposing new, innovative and improved routing protocols for MANET. The important issues in routing are reaching the destination in minimum time at minimum cost.

1.2 Routing Protocols in MANET

Topology in MANET is very dynamic and ever-changing where nodes are free to join or leave arbitrarily. The goal of mobile ad hoc networking is to extend mobility into the realm of autonomous, mobile, wireless domains, where a set of nodes themselves form the network routing infrastructure in an ad hoc fashion (Macker & Corson, 1997). Traditional routing protocol used in wired network cannot be applied directly to wireless and mobile network. Several considerations are needed before we embark on the development of a routing protocol for a wireless network which is non-trivial due to nodes' high mobility.

Generally, there are two different stages in routing; they are route discovery and data forwarding. In route discovery, route to a destination will be discovered by broadcasting the query. Then, once the route has been established, data forwarding will be initiated and sent via the routes that have been determined. Through broadcasting, all nodes that receive the query will broadcast to all neighbours and hence, large number of control messages is transmitted. It will be further compounded if the nodes move and new route need to be recomputed. Frequent route discovery and in some instances, additional periodic updates will cause more bandwidth being utilized and thus more energy wastage. Hence, to conserve the power consumption, route relaying load, battery life, reduction in the frequency of sending control messages, optimization of size of control headers and efficient route reconfiguration should be considered when developing a routing protocol (Chlamtac et al., 2003).

Over the past several years, many routing protocols have been proposed and can be categorized into topology-based (Royer & Toh, 1999) and position-based protocols (Giordano et al., 2004). Topology-based routing protocols route packets based on information about the network links while position-based routing protocols uses physical information about the participating nodes to decide on how to route packets. Topology-based protocols can be further divided into proactive, reactive and hybrid routing protocols. The network links are determined long before routing process in proactive protocols, when routing in reactive protocols and a combination of before and when routing in hybrid protocols. In the position-based protocols, location information of the destination are known and used. There are two sub-divisions in position-based routing protocols, namely

greedy forwarding and restricted flooding. In greedy forwarding, nodes that have the best progress will be selected and data packet will be forwarded to these nodes. Ideally, this process is repeated until the packet arrives at the destination. Note there is no route discovery in greedy forwarding. Restricted flooding, on the other hand, will mitigate broadcast storm problem where only nodes in the direction of the destination will participate in the route discovery until the route to destination is found. The participation of nodes in routing will optimize broadcasting in MANET. Restricted flooding will broadcast messages to a selected number of nodes which is usually more than one that are located closer to the destination. It will significantly reduce not only energy but also reduce the probability of packet collisions of messages rebroadcast by neighbours using the same transmission channel (Stojmenovic, 2002; Mauve et all, 2001; Giordano et al, 2004). Figure 2 shows the categorization of routing protocols in MANET.
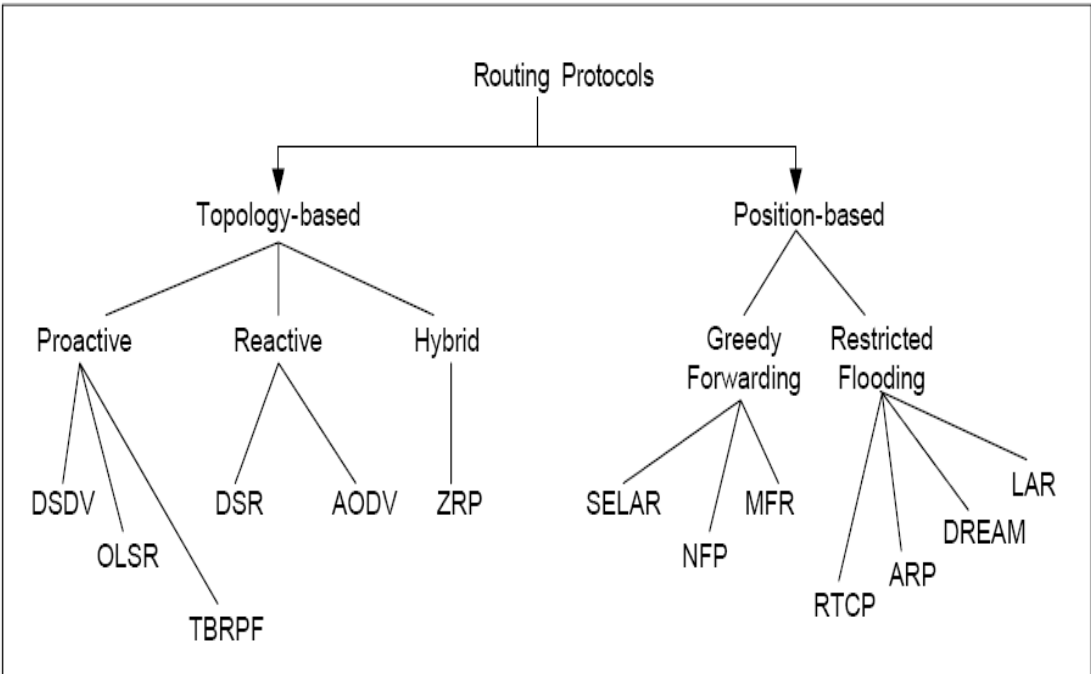

Fig. 2. Catergorization of MANET Routing Protocols

2. Literature Review

With the advent of Global Positioning System (GPS) and MANET environment-based self-positioning (Latiff et al, 2005) and remote-positioning system (Li et al, 2000; Ali et al, 2004), location information can be easily disseminated to the requesting node as required in the position-based routing protocol. Besides availability of location information, complexity of mathematical computations and issues pertaining to the implementation of the said protocols should also be considered. Complex computational iterations will result in processing delay and hence, higher latency while many routing packets transversing in the network will result in high energy consumption and high probability of packet collisions. Hence, position-based routing that restricts the broadcast region will reduce routing

packets, packet collisions and lower end-to-end delay with tolerable percentage of packet delivered.

2.1 Greedy Forwarding

Greedy forwarding requires an up-to-date local topology via periodic beaconing which eliminates route discovery and hence, only data packet forwarding are employed until it reaches the destination. There are several forwarding strategies proposed that differ in the way the node selects the next hop among its neighbours (Stojmenovic, 2002; Macker & Corson, 1999). Figure 3 illustrates the various strategies, where S and D are the source and destination nodes. The circle with radius *r* is the transmission range of S. The strategy is to select and forward the packet to the node that has the best progress towards (or closest to) destination.
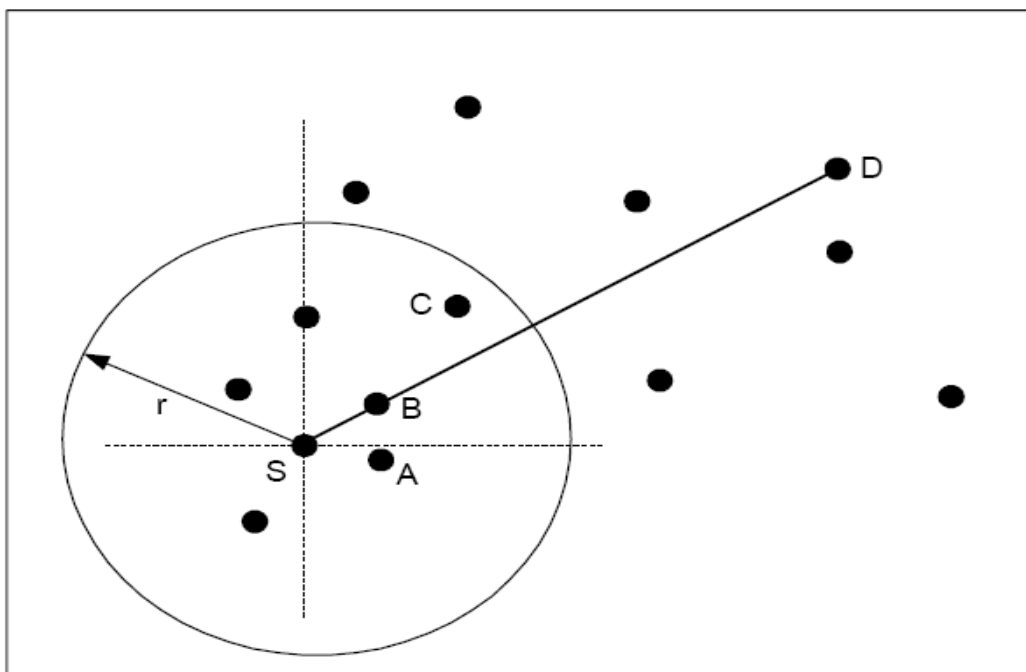

Fig. 3. Greedy forwarding strategies

The first strategy is Most Forward within *r* (MFR) (Hou & Li, 1986) which select nodes that will minimize the number of hops that a packet will traverse in order to reach D. The selected node is C. Nearest with Forward Progress (NFP) (Kranakis et al, 1999) proposed to minimize interference with other nodes and also reduce the overall power consumption by forwarding to the node that is nearest to S which is node A. In compass routing (Chang & Tassiluas, 2000), the forwarding decision will select the neighbour that is closest to the straight line between S and D. In this approach, the selected node is B which minimizes the spatial distance a packet travels. However, there are drawbacks in greedy forwarding where it can only guarantee loop freedom for a certain kind of network topology. Greedy forwarding works well in dense network but degrades in sparse networks. Hence, a path towards destination cannot be found even though the path exists (Stojmenovic, 2002). As described above, proactive information of one-hop neighbours obtained via HELLO

messages periodically transmitted that has information of the sending node location information must be implemented. Hence, knowledge of the local topology will have to be maintained in a neighbours table at each node. This will require storage of neighbour information which meant additional cost. Also, these approaches will also require complex computation at the nodes and hence,will incur delay at the intermediate nodes.

2.2 Restricted Flooding

The main approach in restricted flooding is to limit the flooding region which can be based on distance, angle and distance covered by the next intermediate node. Using distance, only nodes that are nearer to the destination will participate in the route discovery. Nodes that are further away from source will not participate. LAR (Ko & Vaidya,2000) calculates distance from the destination based on location information of the destination that will be extracted from the request packet while (Cartigny et al, 2003) uses the relative neighborhood graph (RNG) with local information of distance to neighbours and distances between neighbours to decide whether its participation has better coverage compared to other nodes. This will minimize the total energy consumption while still maintaining the whole network coverage through broadcasting. (Ali et al, 2005) calculates distances to all nodes in the network and will compare the distance information of the source to the destination extracted from the request packet to determine its participation.

On the other hand, ARP (Kumar Bankar & Xue, 2002) and DREAM [Basagni et al, 1998) uses the angle made from the straight line drawn from source to destination as the restricted region whereby all nodes in this region will participate in the route discovery. However, DDB (Heissenbutte et al, 2004) uses the location information of the destination node and also of the intermediate node which are inserted in the request packet. With this additional information, an intermediate node can calculate the estimated additional covered area that it would cover with its transmission which is based on Dynamic Forwarding Delay (DFD). The concept of DFD is to determine when to forward the packet and node with more area covered will be given a smaller delay to broadcast and hence, will broadcast it first.
All the proposed protocols require quite complex mathematical computation of the distance, angle and coverage at all intermediate nodes to determine the nodes' participation. Information of the source and destination are required and must be inserted in the incoming packet.

In MANET, route discovery is initiated by total flooding of route request (RREQ) messages that consume a large portion of the already limited bandwidth in MANET. As illustrated in Figure 4, RREQ is broadcasted to all neighbours whereby frequent broadcast causes network congestion and degrades the performance of routing protocol. This could be proved by several performance observations that the number of RREQ in the network increases linearly with the node population (Perkins, 2004). The ratio of control packet over data packet even reaches 5000 in one of the experiments.

As such, we suggest utilizing restricted flooding mechanism to optimise the route establishment phase of AODVbis. Restricted flooding is broadcasting messages to a selected number of nodes which is more than one that are located in an area in the vicinity of the destination. Position information of the destination can be obtained from any location

service while position location of the destination can be obtained with the aid of GPS or any other self-positioning system proposed for MANET. Then if these information are piggy-backed in the query packet, nodes will calculate its location with reference to the source and destination and will then decide to broadcast the query or not. Figure 5 illustrates that the same network topology shown in Figure 4 but with limited flooding. RREQ packets will be broadcast by nodes located in the request zone which is a quadrant drawn with respect to source node coordinates. Nodes participation is denoted by shaded circles with arrows indicating the direction of broadcast while lesser-toned circles indicate non-participating nodes. With this unique approach of using quadrant as the broadcast region, we proposed Quadrant-based Directional Routing Protocol or Q-DIR.
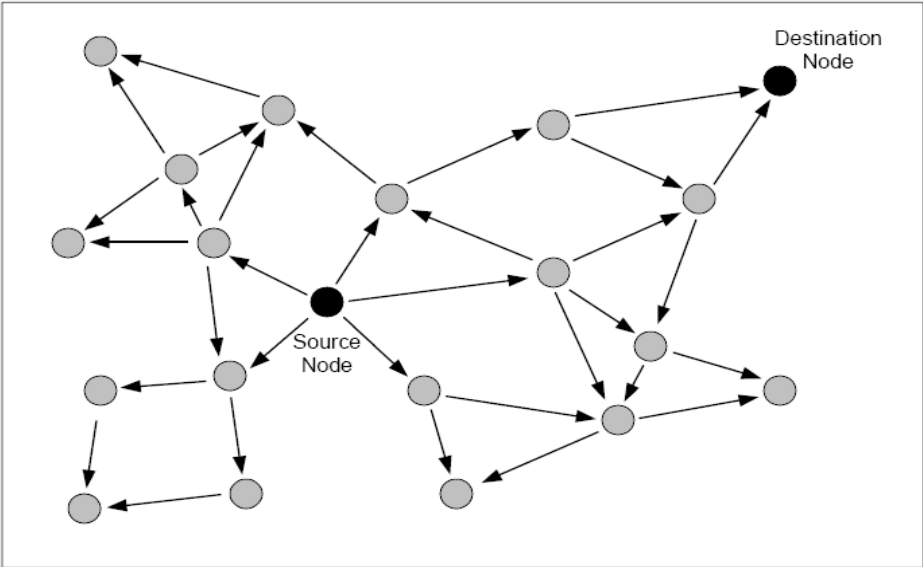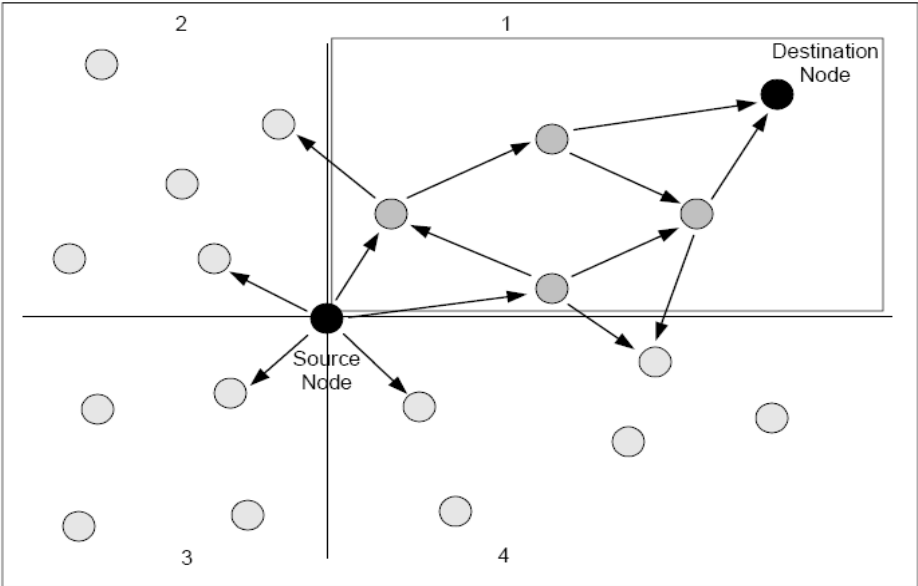

Fig. 4. RREQs broadcast based on Total Flooding.

Fig. 5. RREQ broadcast based on Restricted Flooding.

2.3 Quadrant-based Directional Routing Protocol (Q-DIR)

Q-DIR is a limited flooding routing protocol that concentrates on a specified zone using location information provided by a location service. It restricts the broadcast region to all nodes in the same quadrant as the source and destination and does not require maintenance of a separate neighbours table at each node as in (Ko & Vaidya, 2000; Kumar Banka & Xue, 2002; Cartigny et al, 2003; Heissenbutte et al, 2004). Q-DIR determines the quadrant of the current node based on the coordinates of source, destination and the current node that will direct the packet towards the destination. Even though (Cartigny et al, 2003) uses all these information to determine the distance or area covered, it requires trigonometric computations which will further incur delay if computed in kernel space.

Decision to broadcast or discard will be done as the RREQ packet is received by the node. Unlike LAR scheme 2 (Ko & Vaidya, 2000), geocast-enhanced AODVbis (Ooi, 2005; Latiff et al, 2006), nodes in Q-DIR do not keep a distance table or a neighbours table which must be updated frequently. Keeping a distance table is much like a table-driven proactive routing concept. The size of the table will increase for a larger network because nodes need to have distance information of itself to every other node in the network which will vary from node to node. This will pose a constraint to the maximum number of nodes in the network as the memory allocation to store distance information of every node in the network at each node is limited and scarce.

In Q-DIR, the RREQ packet which contains the coordinates of the source and destination will be the only information the current node needs to decide to participate in the routing or not. The decision to participate at each node is made immediately as the node receives the RREQ packet and a neighbours table is not required to make the decision.

Q-DIR will significantly reduce not only energy but also reduce the probability of packet collisions of messages rebroadcast by neighbours using the same transmission channel. This will result in reduced routing overhead especially in a dense network.

3. Development of Q-DIR in Ns-2

In Q-DIR operation, the location information of the source and destination nodes is piggy-backed in the route request (RREQ) packet and then broadcasted. Upon receiving the RREQ, intermediate nodes will compare using a simple mathematical comparison based on the coordinates of source, destination and the current node that directs the packet towards the destination and as illustrated in Figure 6. This mathematical processing will be done in the kernel environment to eliminate the cross-over from user to kernel space and vice versa. Hence, the decision to participate is made immediately.

> Quadrant of me compared to source?
> Quadrant of destination compared to source?
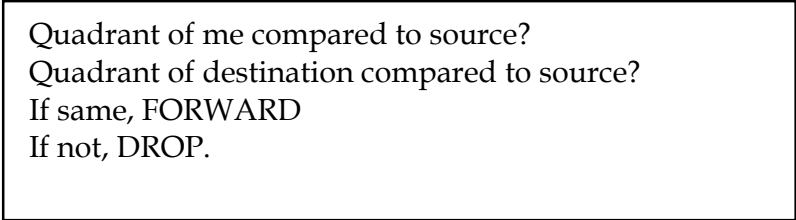> If same, FORWARD
> If not, DROP.

Fig. 6. Q-DIR decision at each intermediate node.

Once the decision to broadcast has been made, the intermediate node will insert its location by replacing the source node coordinates and append its address and sequence number at the end of the RREQ packet. It will then broadcast the packet. The process will repeat at each intermediate node until it reaches the destination. The replacement of the source node location information with the intermediate node coordinates will make the packet more directed towards the destination since the comparison now is based on the previous node. Upon receiving the RREQ, destination node will send a route reply message (RREP) back to source via the path taken to reach the destination that was appended in the RREQ as it traverses across the network. There is no need for the route discovery to the source node. Figure 7 shows the format of the RREQ packet in Q-DIR where the source and destination nodes location information are inserted are highlighted.

0 1 2 3 4 5 6 7	0 1	2 3 4 5 6 7 0 1	2 3 4 5 6 7	0 1 2 3 4 5 6 7
Type	D G	Reserved		Hop Count
RREQ ID				
X_s		Y_s	X_d	Y_d
Destination IP Address				
Destination Sequence Number				
Originator IP Address				
Originator Sequence Number				
Path Node IP Address				
Path Node Sequence Number				
(additional path node IP address and sequence number pairs) ...				

Fig. 7. RREQ format in Q-DIR.

There are several open source network simulators such as Commnet, OMNeT++ but Network Simulator-2 (Ns-2) has been found to be a widely used tool for simulating inter-network topologies and to test and evaluate various networking protocols (CMU Monarch Project, 2006). It is a discrete event simulator written in C++ and uses Massachusetts Institute of Technology (MIT) Object Tool Command Language (OTcl) as a command and configuration interface. The most important characteristic of a discrete-event approach is that the components of an actual network are represented within the software and real events are simulated by the operation of the software.

Ns-2 can be installed on both Windows and Linux platforms. For Q-DIR, the simulation work is done on a Red Hat Linux platform (Chakeres et al, 2005) The compiler used in Q-DIR is the ns-allinone-2.28 version (CMU Monarch Project, 2006). The underlying protocol is AODVbis which has the path accumulation feature (Gwalani et al, 2003). The Dynamic MANET On-demand – University of Murcia (DYMOUM) (Ros & Ruiz, 2004) is based on DYMO which is an internet draft dated June 2005. DYMO enables reactive, multi-hop routing between participating node that wish to communicate. The basic operation of DYMO is similar to AODVbis which are route discovery and route management and the

differences are in the new packet format, generic packet handling, unsupported element handling and optional path accumulation. DYMOUM has There are three types of elements that have been defined in DYMO. They are RE (Routing Element), RERR and UERR (Unsupported-element Error) and RE can be further divided into RREQ and Route Reply (RREP). From the description given, DYMO can be used as the underlying protocol in this work. DYMO is reactive and implements route discovery and path accumulation. Even though it uses a generic element structure but basically has the needed RREQ, RREP and RERR packets as in the AODVbis routing protocol. Any modification work should be done in the C++ hierarchy.

3.1 RREQ packet format

As shown in Figure 8, to modify the RREQ packet, the source and destination coordinates are declared as a double precision integer. In the DYMOUM source file, when a new RREQ is generated by the source node, the **NS_CLASS** *re_create_rreq ()* procedure will create the RREQ packet. The RREQ packet requires location information of the source node; therefore the following syntax will extract the source coordinates from the ns-2 environment which is searched by using the node address.

Node*node=Node::get_node_by_address(re_node_addr.s_addr);
((MobileNode *)node)->getLoc(&x,&y,&z);

To extract the destination coordinates, a declaration of the following were made at the beginning of the source file that permits calling for those information using Tcl hooks in the ns-2 platform. Description of the declaration for Tcl hooks will be described in the following section.

extern dst_x, dst_y;

```
/* dymo_re.c */
RE *NS_CLASS re_create_rreq(struct in_addr target_addr,u_int32_t
target_seqnum,struct in_addr re_node_addr,u_int32_t re_node_seqnum,u_int8_t prefix,
u_int8_t g,u_int8_t ttl, u_int8_t thopcnt)
{
        RE *re;
/* declare variable type for coordinates and get info of node using s_addr */
double x=0,y=0,z=0;
Node *node=Node::get_node_by_address(re_node_addr.s_addr);
((MobileNode *)node)->getLoc(&x, &y, &z);

        re              = (RE *) dymo_socket_new_element();
        re->m           = 0;
        re->h           = 0;
        re->type        = DYMO_RE_TYPE;
        re->a           = 1;
        re->i           = 0;
        re->res1        = 0;
        re->res2        = 0;
        re->ttl         = ttl;
        re->len         = RE_BASIC_SIZE + RE_BLOCK_SIZE;
        re->thopcnt     = thopcnt;
        re->target_addr         = (u_int32_t) target_addr.s_addr;
        re->target_seqnum       = htonl(target_seqnum);
/* add new field for x and y coordinates of source and destination in RREQ packet */
        re->dst_x=dst_x;  /* x-coord for destination */
        re->dst_y=dst_y;  /* y-coord for destination */
        re->src_x=x; /* x-coord for source */
        re->src_y=y; /* y-coord for source */

        re->re_blocks[0].g              = g;
        re->re_blocks[0].prefix         = prefix;
        re->re_blocks[0].res            = 0;
        re->re_blocks[0].re_hopcnt      = 0;
        re->re_blocks[0].re_node_addr           = (u_int32_t) re_node_addr.s_addr;
        re->re_blocks[0].re_node_seqnum         = htonl(re_node_seqnum);
        return re;
}
```

Fig. 8. Declaration of additional fields to the RREQ packet.

3.2 Tcl Hooks

Ns-2 consists of two hierarchies: compiled C++ hierarchy and the OTcl that make use of objects in C++ through OTcl linkages that have a one-to-one correspondence to each other. The objects that have already been linked are "*no_path_acc_*", "*reissue_rreq*", and "*s_bit*". Therefore, to link the coordinates of the destination node that will be declared in the tcl script in the OTcl environment, links for both objects have to be created and the declarations are as shown in Figure 9. This *ns-2.28/dymoum-0.1/ns/dymo_um.cc* file has other links to the C++ hierarchy that are relevant to the DYMO configuration but will not be shown here. The variables for *dst_x* and *dst_y* in the header file *dymo_um.h* to enable referencing by *dymo_um.cc* have been declared. To enable calling DYMOUM from the tcl script, the agent DYMOUM (Agent/DYMOUM), *dst_x* and *dst_y* in the *ns-default.tcl* file in ns-2 library are inserted.

```
/* dymoum-0.1/ns/dymo_um.cc */
NS_CLASS DYMOUM(nsaddr_t id) : Agent(PT_DYMOUM), qtimer_(this), initialized_(0), pq_len(0)
{
        /* Enable usage of some of the configuration variables from Tcl.
        Note: Do NOT change the values of these variables in the constructor
        after binding them! The desired default values should be set in
        ~ns/tcl/lib/ns-default.tcl instead.*/
        bind_bool("no_path_acc_", &no_path_acc);
        bind_bool("reissue_rreq_", &reissue_rreq);
        bind("dstx", &dst_x);
        bind("dsty", &dst_y);

        .
        .
        .
        .
        .
}
```

Fig. 9. Binding of Tcl objects to the C++ hierarchy

3.3 Processing RREQ

As described in Section 3, processing of RREQ consists of two events. They are Generating RREQ when the current node has data to send and initiates the route discovery for a certain destination and Receiving RREQ that is implemented at the intermediate nodes that receives the query broadcast. In the same *dymo_re.c* file previously mentioned, in the function **NS_CLASS re_process(),** variables such as temporary fields to store coordinates of current node and value of quadrant are declared. Then, the syntax to search for the current node coordinates and store these information in *mynode_x* and *mynode_y* will be made as shown in Figure 10.

```
void NS_CLASS re_process(RE *re, struct in_addr ip_src, u_int32_t ifindex)
{
        struct in_addr node_addr;
        rtable_entry_t *entry;
        int i;
/* declare variable type for x and y coordinates field in packet received */
        double mynode_x,mynode_y;
        int quaddest, quadsrc;
        double x=0,y=0,z=0;
/* get info of current node using address from ns */
        Node *node=Node::get_node_by_address(ip_src.s_addr); /* current node
address*/
        printf("process current node=%d\n",ip_src.s_addr);
        ((MobileNode *)node)->getLoc(&x, &y, &z);
        mynode_x=x;mynode_y=y; /* put the coordinates into mynode_x and
mynode_y */
/* to check for correct info extracted from ns */
        printf("srcx=%f,srcy=%f,dstx=%f,dsty=%f,myx=%f, myy=%f\n",
        re->src_x,re->src_y,re->dst_x,re->dst_y,mynode_x,mynode_y);
                              .
                              .
                              .
```

Fig. 10. Declaration of variables in *dymo_re.c.*

When receiving RREQ, nodes will compare the quadrant of destination and current node and the codes are be inserted right after the declaration of the variables in function **NS_CLASS re_process()**. The coordinates of the source are denoted by *src_x* and *src_y*, while the coordinates of destination are denoted by *dst_x* and *dst_y*. The current node coordinates are denoted by *mynode_x* and *mynode_y*. If the quadrant of the source is equal to the quadrant of destination, the current node will broadcast the request. The code for this receiving RREQ is shown in Figure 11.

```
if ((re->dst_x >= re->src_x) && (re->dst_y >= re->src_y)) quaddest = 1;
if ((re->dst_x >= re->src_x) && (re->dst_y < re->src_y)) quaddest = 4;
if ((re->dst_x < re->src_x) && (re->dst_y >= re->src_y)) quaddest = 2;
if ((re->dst_x < re->src_x) && (re->dst_y < re->src_y)) quaddest = 3;
printf("Quadrant of destination compared to previous node is %3d\n", quaddest);

if ((mynode_x >= re->src_x) && (mynode_y >= re->src_y)) quadsrc = 1;
if ((mynode_x >= re->src_x) && (mynode_y < re->src_y)) quadsrc = 4;
if ((mynode_x < re->src_x) && (mynode_y >= re->src_y)) quadsrc = 2;
if ((mynode_x < re->src_x) && (mynode_y < re->src_y)) quadsrc = 3;
printf("My quadrant compared to previous node is %3d\n",quadsrc);
if (quaddest != quadsrc )
    {
     printf("I am not in the same quadrant as destination. DROP packet.\n");
     return;
    }
else printf("I am in the same quadrant as destination.  FORWARD packet.\n");
```

Fig. 11. Code for broadcast decision at each node.

4. Verification of Q-DIR

4.1 Network Model

A network model N-1 as shown in Figure 12 is used which consist of 6 nodes and the coordinates are carefully chosen so that there will be at least 2-hops transmission to the destination. The imaginary x- and y-axis are drawn to show in which quadrant the nodes are located with reference to their immediate neighbours.
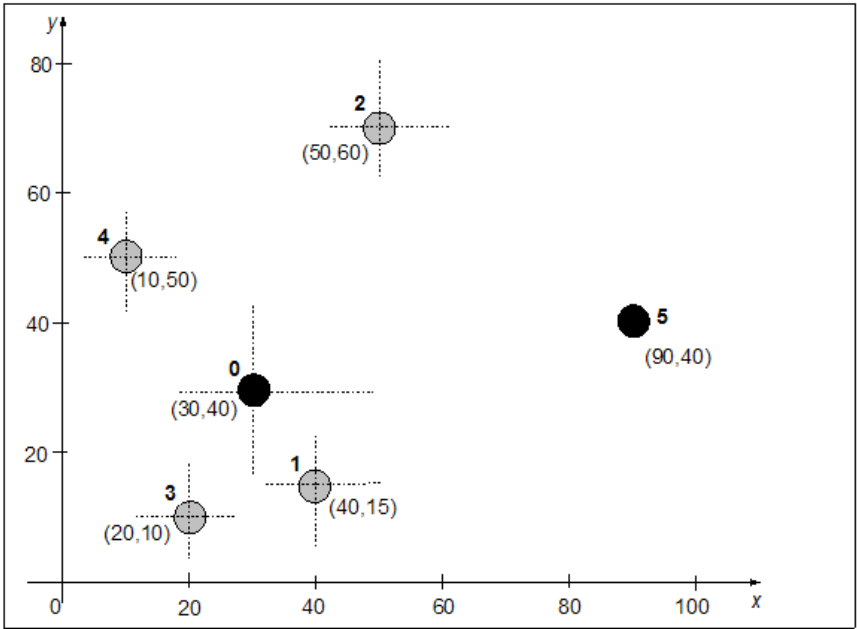

Fig. 12. Topology N-1

For topology N-1, the 1-hop neighbours of node 0 are nodes 1, 2, 3 and 4 while the neighbours of node 5 are 1 and 2. The configuration parameters used for both Q-DIR and AODVbis routing are shown in Table 1. The values can be modified depending on the network and its environment. The maximum number of hops between nodes have been set to 10 while the estimated average of one hop traversal time is set to 0.6 s. From I-D (Perkins et al, 2003), for correct operation, the route delete period must be greater than both (Allowed HELLO loss* HELLO interval) and the total traversal time.

The MAC layer protocol used is IEEE 802.11 DCF CSMA/CA. The data rate have been set to 2 Mbps and the network protocol is IP. The path loss model used is the log-normal path loss model (Rappaport, 2002) and the value for n is 2.4 and the standard deviation σ is 4. To simulate in ns-2, the receive threshold power has to be determined first in order to set the transmission range to 30 meters for all 1-hop neighbours. The default transmitted power is 0.28318 W. The receive threshold power calculated is 1.20475e-08 watts and the packet rate is set to 1 packet/sec while the packet size is set at 64 bytes and 512 bytes with a CBR (Constant Bit Rate) traffic pattern.

Configuration Parameters	Value
Maximum number of possible hops between two nodes	10
Average one hop traversal time	60 milliseconds
Route discovery time	2400 milliseconds
Route delete period	4800 milliseconds
Number of RREQ tries	3
Total traversal time	1200 milliseconds
HELLO interval	1000 milliseconds
Allowed HELLO loss	2

Table 1. Simulation Configuration Parameters.

4.2 Results

The simulation for topology N-1 was run and results show that for N-1, nodes 1, 2, 3 and 4 will all receive the RREQ packet from source node 0 destined for destination node 5. However, nodes 1, 3, and 4 will drop the packet since they are in different quadrant from the source and destination. Figure 13(a) shows the snapshot of the message displayed on the screen when running the simulation. The snapshot shows that node 1 and 3 drop the RREQ received from source node 0 while Figure 13(b) shows that node 4 drops the packet from source node 0. On the other hand, from Figure 13(c), node 2 forwards the packet to destination node 5 since it is in the same quadrant as destination compared to source.

(a)

(b)

(c)
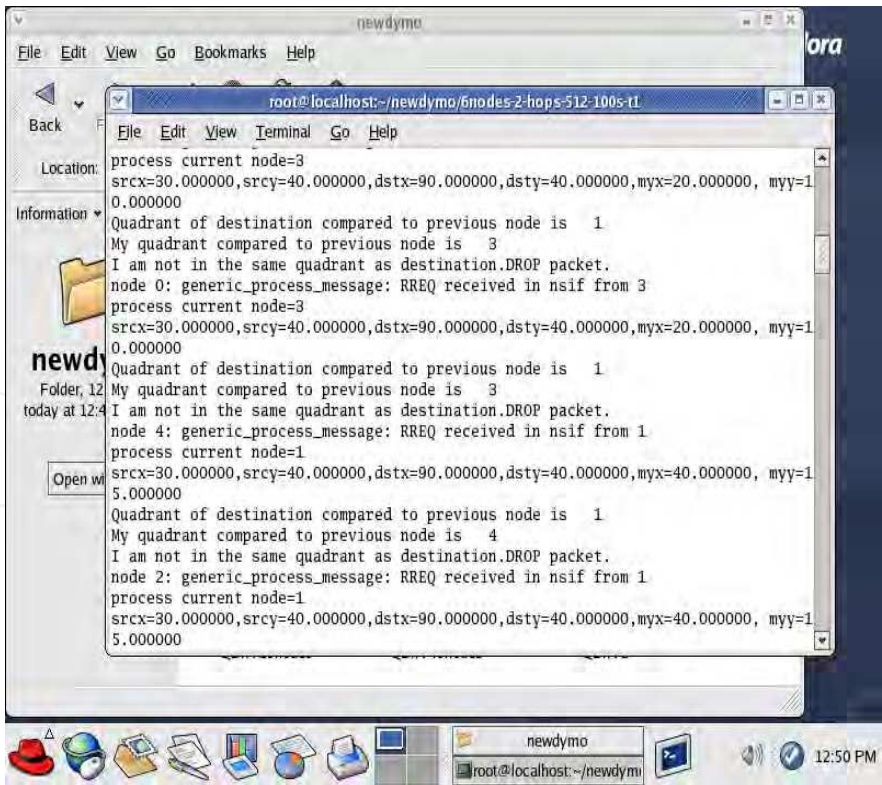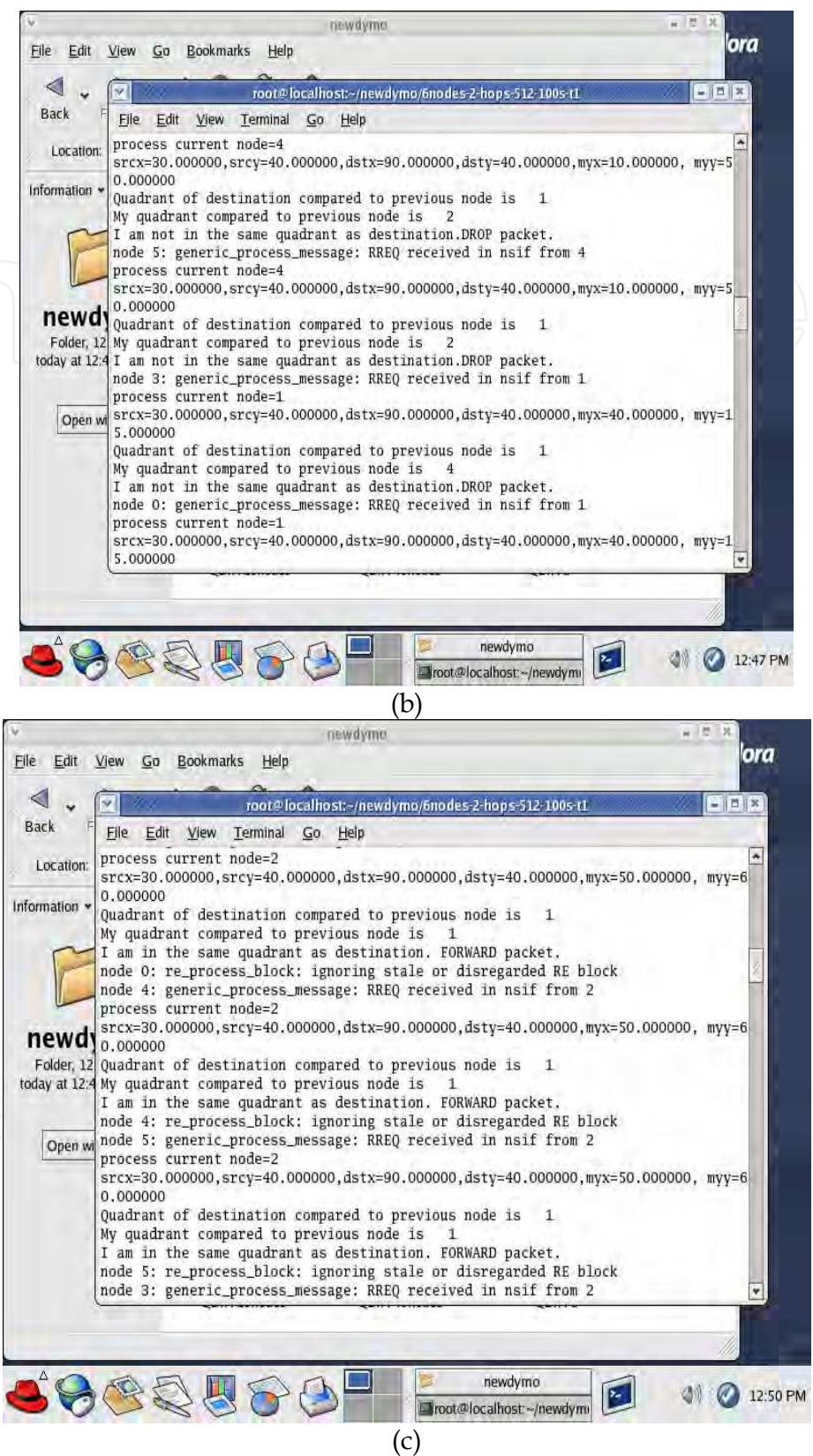
Fig. 13. RREQ packet broadcasting decision for Topology N-1 (a) Node 1 and 3 drop. (b) Node 4 drops. (c) Node 2 broadcasts.

5. Performance of Q-DIR in Dense Network

The study to evaluate the performance of Q-DIR in a large and densely populated network were conducted and it is hoped that results will show that Q-DIR with reduced collisions, and less contention of bandwidth, less routing overhead and consequently, power consumption is inherent and reflects that this new routing protocols is implementable and economical.

5.1 Dense Network Model

Figure 14 shows a network model of 49 nodes that forms a 7 by 7 grid model where the distance from adjacent nodes are 30m. Based on this grid model, the density is 1 node per 661m². In the network model, the x- and y-axis of the Cartesian coordinate system have been drawn to denote in which quadrant the nodes are located. The source and destination are denoted by the letter S and D respectively and destination node is at the top right edge of the grid. The simulation configuration parameters used in the simulation are as shown in Table 1.
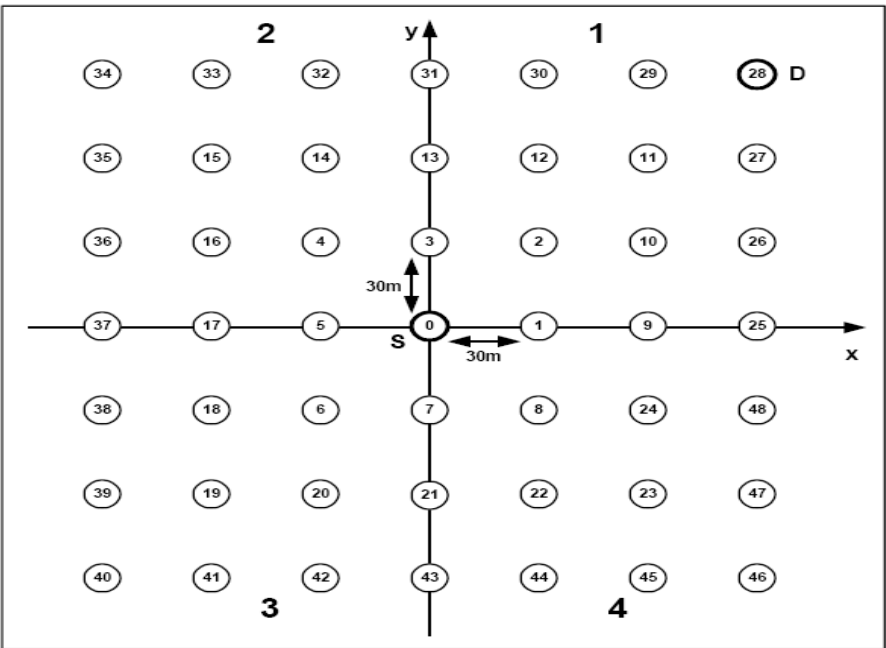
Fig. 14. Simulation Network Model of 49 nodes

5.2 Performance Metrics

Two protocols were simulated and they are AODVbis which is a total flooding protocol and Q-DIR which is based on restricted flooding. The performance metric used are as follows:

- *normalized routing overhead* - The number of routing packets transmitted per data packet received at the destination.
- *Effective energy consumption per data packet received* - The total energy consumption in the network for every data packet successfully received by the destination. This is the metric on the effectiveness of energy consumption when routing data packets.

5.3 Simulation Results

A. Effect of Varying Simulation Time

The simulation time was varied from 100s to 800s in steps of 100s. The number of routing packets that are broadcast and the corresponding data packet received at the destination in the network are counted for both AODVbis and Q-DIR routing protocol. Figure 15 shows the normalized routing overhead graphs for both protocols. As the simulation time increases to 800s, both protocols show reduced routing packets and leveled to a constant as it approaches 800s. The average normalized routing overhead in AODVbis is 338 packets while in Q-DIR, the average normalized routing overhead is 128 packets per data packet received. It is observed that 160% more routing packets are transmitted in AODVbis compared to Q-DIR due the higher number of node participations in the network in AODVbis.

Figure 16 shows graph for effective energy consumed per data packet received for both protocols. Both protocols shows a reduced energy consumption as the simulation time increases The average effective energy is 2.43 J in AODVbis and 1.48 J in Q-DIR. Q-DIR consumes 64% less energy to send packets since only a quarter of the number of nodes participated in the routing process which is a limited flooding protocol based on quadrant.
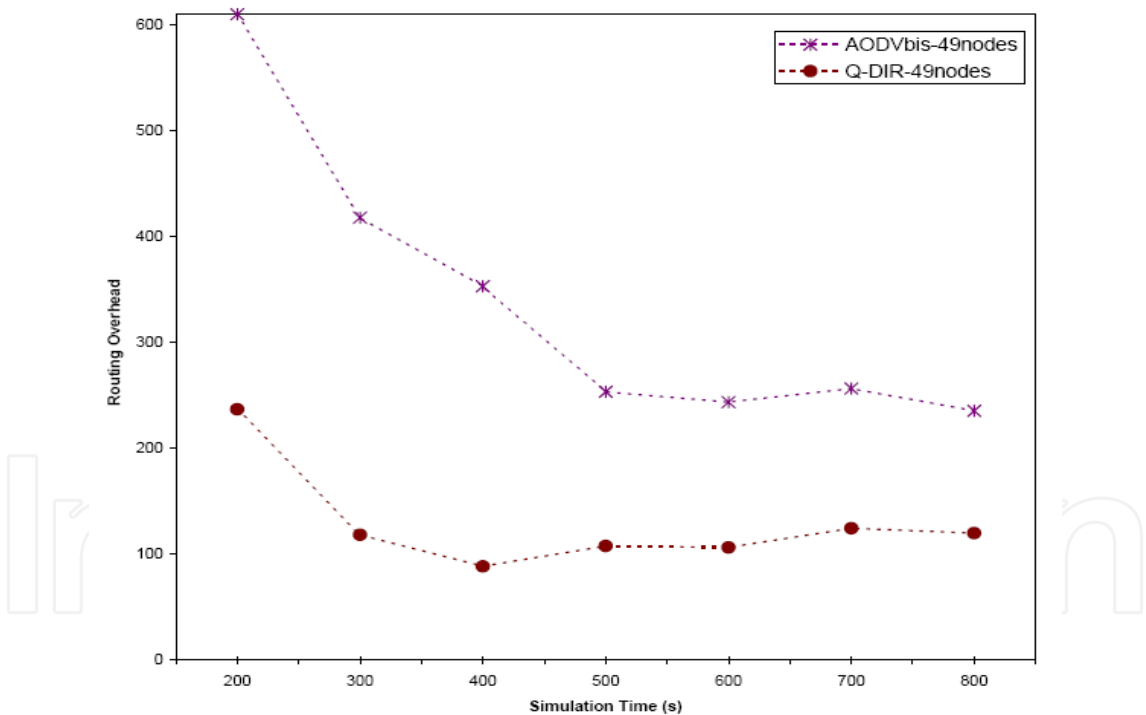
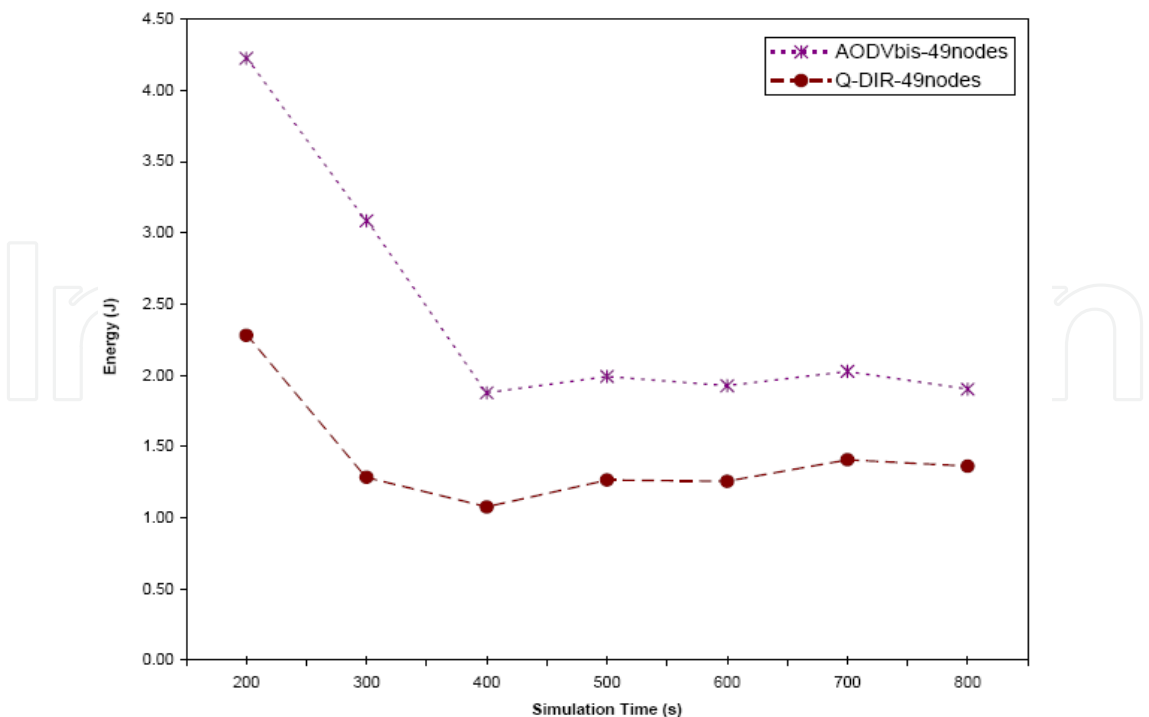Fig. 15. Normalized routing overhead with simulation time.

Fig. 16. Effective energy consumed per data packet received in Q-DIR.

B. Effect of Varying Packet Rate

Both AODVbis and Q-DIR routing protocols are simulated in the 49 nodes topology for a simulation time of 400s because the performance of both protocols remains constant. The transmission rate was varied in steps of 32 kbits/s with initial rate of 16 kbits/s to a maximum of 144 kbits/s. Figure 17 shows the average normalized routing overhead for both protocols which increases as the transmission rate increases. The graph for AODVbis shows large fluctuations as the transmission rate increases. AODVbis sends out an average of 255.664 normalized routing packets compared to Q-DIR which sends out only 108.08 packets. The large fluctuations in AODVbis are due to the total flooding algorithm of AODVbis and hence the routes taken vary for different transmission rate. However, the graphs in Q-DIR remain consistent throughout due to the directed flooding based on quadrant.

Figure 18 shows the graphs for effective energy consumed per data packet received for both AODVbis and Q-DIR protocols. The effective energy for AODVbis fluctuates as the transmission rate increases but for Q-DIR, it remains constant. Again, the fluctuation in AODVbis is due to different route taken at different transmission rate. .AODVbis consumes an average of 1.574 J of energy while Q-DIR consumes only 1.084 J of energy which 45% less energy consumed compared to AODVbis. Based on this trend in energy consumption, less power is consumed if only a section or an area of a network participates in the routing.
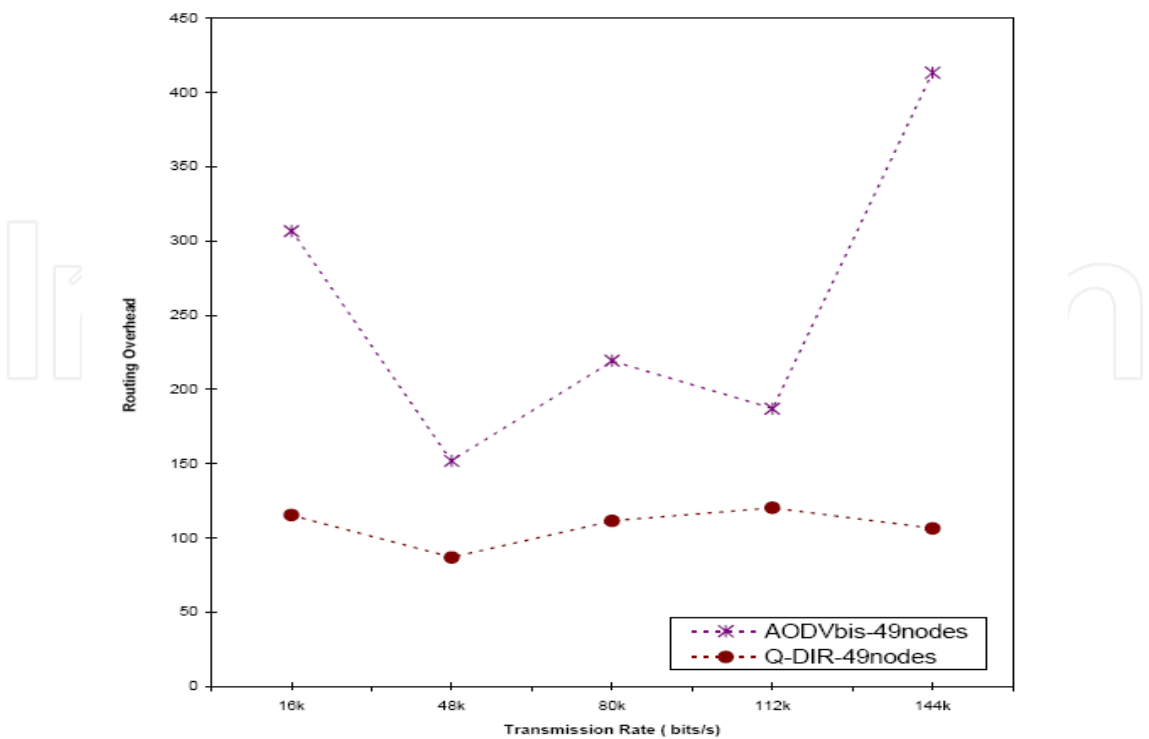
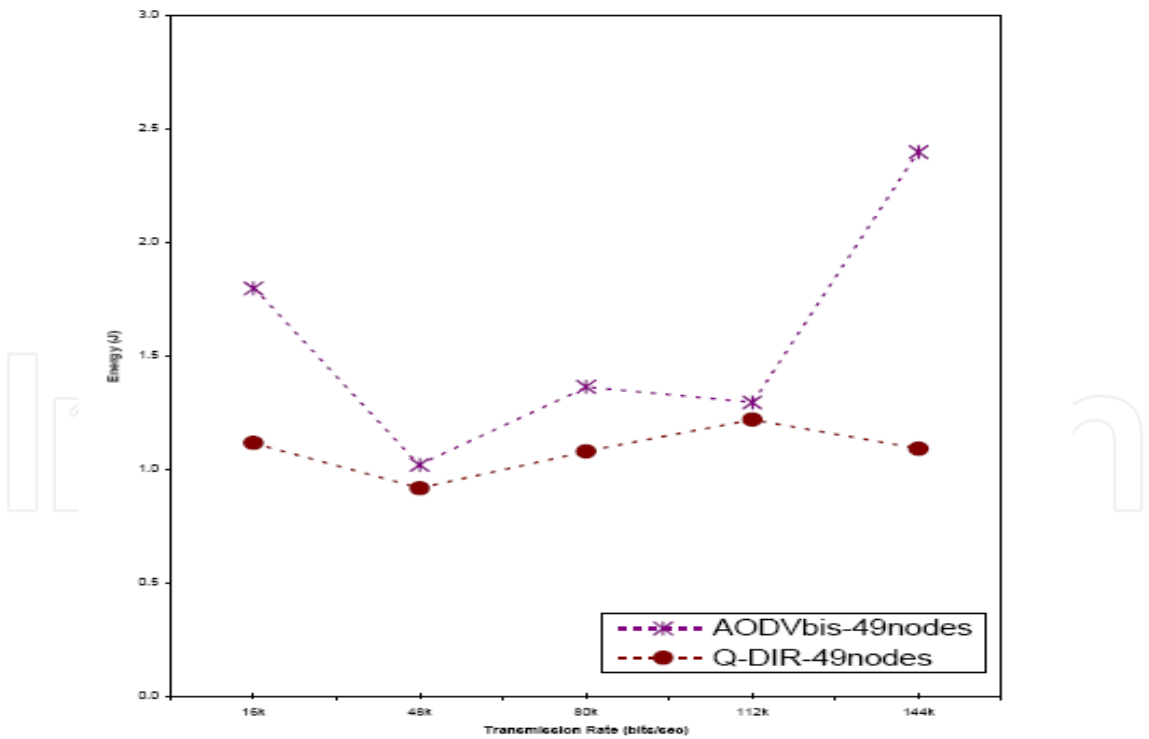Fig. 17. Normalized routing overhead for 49 nodes.

Fig. 18. Effective energy consumed per data packet received

6. Conclusion

This paper has presented the performance of Q-DIR which is a restricted flooding algorithm which uses location information of the source, destination and the intermediate node to determine the broadcasting decision. Nodes that are in the restricted broadcast region will broadcast while other nodes which are out of this region will ignore the RREQ packet. The simple mathematical comparison is implemental in the kernel environment which does not incur processing delay due the crossing from user to kernel space and vice versa. The simulation results shows that implementing Q-DIR reduces the power by 160% as the simulation time is increased and by 45% as the transmission rate increases compared to AODVbis. The restricted flooding and directional routing reduces the number of participating nodes as the RREQ traverses in the network towards the destination node and hence reduced routing overhead and power consumption are achieved in Q-DIR.

7. References

Agrawal, D.P. and Zeng, Qing-An. (2003). *Introduction to Wireless and Mobile Systems*. USA:Brooke/Cole-Thomson Learning.

Ali, A., Latiff, L.A., Fisal,N. (2004). GPS-free Indoor Location Tracking in Mobile Ad Hoc Network (MANET) using RSSI. RFM 2004, Malaysia.

Ali, A., Latiff, L.A., Ooi, Chia-Ching, and Fisal, N. (2005). Location-based Geocasting and Forwarding (LGF) Strategy in Mobile Ad Hoc Network (MANET) *ICT 2005*, Cape Town, South Africa, May 2005.

Basagni, S., Chlamtac, I. and Syrotiuk, V.R. (1998). A Distance Routing Effect Algorithm for Mobility (DREAM). *ACM/IEEE International Conference on Mobile Computing and Networking*. Oct. 1998. 76-84.

Cartigny, J., Simplot, D. and Stojemenovic, I. (2003). Localized minimum-energy broadcasting in ad hoc networks. *INFOCOM 2003*. 30 March-3 April 2003. 3:2210 – 2217.

Chakeres, I. D., Royer, E.M. and Perkins, C. E. (2005). Dynamic MANET On-demand (DYMO) Routing . *Internet Draft of IETF MANET WG*. draft-ietf-manet-dymo-02.txt. June 2005.

Chang, J. and Tassiluas, L. (2000). Energy Conserving Routing in Wireless Ad Hoc Networks. *Proceedings of IEEE INFOCOM 2000*. 22-31.

Chlamtac, I., Conti, M. and Liu, J. J. (2003). Mobile Ad-hoc Networking: Imperatives and Challenges. *Elsevier Ad Hoc Networks Journal*. 2003. Vol. 1(1): 13-64.

Department of Computer Science, Rice University. (2006). *CMU Monarch Project*. http://www.monarch.cs.cmu.edu/. Accessed June 2006.

Giordano, S., Stojmenovic, I. and Blazevic, L. (2004). Position-based Routing Algorithm for Ad Hoc Networks: A Taxanomy. In: *Ad Hoc Wireless Networking*. X. Cheng, X Huang and D.Z. Du (Eds.). 103-136. Kluwer Publishers.

Gwalani, S., Belding-Royer, E.M. and Perkins, C.E. (2003). AODV-PA:AODV with Path Accumulation. *International Conference on Communications*. May 2003. 26(1):527-531.

Heissenbuttel, M., Braun, T., Walchli, M. and Bernoulli, T. Broadcasting in Wireless Multihop Network with Dynamic Forwarding Delay Concept. *Technical Report IAM-04-010*. Institute of Computer Science and Applied Mathematics University of Berne, Switzerland, December 2004.

Hou, T.C. and Li, V.O.K. (1986). Transmission Range Control in Multihop Packet Radio Networks. *IEEE Transactions. In Communications*. 34(1):38-44.

Ko, Y.B. and Vaidya, N.H. (2000). Location-Aided Routing (LAR) in Mobile Ad Hoc Networks. *ACM/Baltzer Wireless Networks Journal.* 2000. 6:307-321.

Kranakis, E., Singh, H. and Urrutia, J. (1999). Compass Routing on Geometric Networks. *Proceeding of the 11th Canadian Conference on. Computational Geometry.* Vancouver, Canada.

Kumar Banka, R. and Xue, G. (2002). Angle routing protocol: location aided routing for mobile ad-hoc networks using dynamic angle selection. *MILCOM 2002 Proceedings.* 7-10 Oct 2002. 1:501-506.

Latiff, L.A., Ali, A., Ooi, C.-C. and Fisal, N. (2005). Development of an Indoor GPS-free Self-Positioning System for Mobile Ad Hoc Network (MANET). MICC-ICON 2005, Malaysia.

Latiff, L.A., Ali, A., Ooi, C.-C. and Fisal, N. (2006). Implementation of a Quadrant-based Directional Routing Protocol (Q-DIR) in Wireless Mobile Ad Hoc Network. *IASTED NCS 2006.* 29-31 Mac 2006. Chiang Mai, Thailand. IASTED. 2006.

Li, J., Jannotti, J., De Couto, D.S.J. , Karger, D.R. , Morris, R. (2000). A Scalable Location Service for Geographic Ad Hoc Routing. Mobicom 2000.

Macker, J. and Corson, S. (1997). Mobile Ad Hoc Networks (MANET). 1997. http://www.ietf.org/html.charters/manet-charter.html

Mauve, M., Widmer, A. and Hartenestein, H. (2001). A Survey on Position-based Routing in Mobile Ad Hoc Network. *IEEE Network.* 15(6):30-39.

Milanovic, N., Malek, M., Davidson, A. and Milutinovic, V. (2004). Routing and Security in Mobile Ad hoc Network. *IEEE Computer Journal.* 37(2). 61-65.

Ooi, C.-C. (2005). *Implementation of Geocast-Enhanced AODVbis Routing Protocol in MANET.* M.Sc. Thesis, UTM;2005.

Perkins, C.E., Royer, E. M and Chakeres, I. (2003). Ad hoc On Demand Distance Vector-bis (AODVbis) routing. *Internet Draft of IETF MANET WG.* October 2003.

Perkins, C.E. (2004). Convergence in Ad hoc Networking Protocol. *The IASTED International Conference on WNET 2004.* July 2004.

Rappaport, T.S. (2002). *Wireless Communications: Principles and Practice.* 2nd Edition. USA. Prentice Hall. 2002.

Ros, F. J. and Ruiz, P. M. Implementing a New Manet Unicast Routing Protocol in NS2. December 2004. http://www.masimum.dif.um.es/nsrt-howto/html/nsrt-howto .html.

Royer, E.M and Toh, C.K. (1999). A Review of Current Routing Protocol for Ad-Hoc Mobile Wireless Network. *IEEE Personal Communications.* pp. 46-55, April 1999.

Stojmenovic, I. (2002). Position-based Routing in Ad Hoc Networks. *IEEE Communications Magazine.* 40(7). 128-134.

Sun, J.Z. (2001). Mobile Ad Hoc Networking An Essential Technology for Pervasive Computing, *Intl. Conference on Info-tech & Info-net.* Beijing, China.

Xue, Q. and Ganz, A. (2003). Adaptive Routing in Ubiquitous Mobile Access Networks. *Proceedings of IEEE VTC.* 6-9 October, 2003. Orlando, Florida: IEEE. 3070-6074.

Trends in Telecommunications Technologies

Edited by Christos J Bouras

The main focus of the book is the advances in telecommunications modeling, policy, and technology. In particular, several chapters of the book deal with low-level network layers and present issues in optical communication technology and optical networks, including the deployment of optical hardware devices and the design of optical network architecture. Wireless networking is also covered, with a focus on WiFi and WiMAX technologies. The book also contains chapters that deal with transport issues, and namely protocols and policies for efficient and guaranteed transmission characteristics while transferring demanding data applications such as video. Finally, the book includes chapters that focus on the delivery of applications through common telecommunication channels such as the earth atmosphere. This book is useful for researchers working in the telecommunications field, in order to read a compact gathering of some of the latest efforts in related areas. It is also useful for educators that wish to get an up-to-date glimpse of telecommunications research and present it in an easily understandable and concise way. It is finally suitable for the engineers and other interested people that would benefit from an overview of ideas, experiments, algorithms and techniques that are presented throughout the book.

INTECH
open science | open minds