
Multipath Distance Vector Zone Routing Protocol for Asymmetric Mobile Ad-Hoc Networks MDVZRPA

Idris Skloul Ibrahim
isi3@macs.hw.ac.uk

A. Etorban
etorban @macs.hw.ac.uk

Peter J.B King
pjbk@macs.hw.ac.uk

School of Mathematical and Computer Sciences (MACS)
Heriot Watt University at Edinburgh UK

Abstract— Most of the ad hoc routing protocols assume that all wireless networks are symmetric (bidirectional links). In reality any practical network has some links may be unidirectional and hence the network is asymmetrical rather than symmetrical. The presence of such links can reduce the performance of the existing protocol and could lead to network clogging. In this paper we introduce a Multipath Distance Vector Zone Routing Protocol for Asymmetric mobile ad-hoc networks (MDVZRPA), which is a modification to MDVZRP. It is a hybrid routing protocol assumes that all routes in the routing table are active and usable, unless a broken link has been reported or discovered for reducing control traffic.

In addition to adopting MDVZRP technique, MDVZRPA is designed to deal with both bidirectional and unidirectional links by adding a new field called Symmetric-link in each route to distinguish between the two link types.

Keywords: MDVZRP, Asymmetrical Networks, Unidirectional and Bidirectional links.

I. INTRODUCTION

In recent years, mobile computing has enjoyed a tremendous rise in popularity. The continued minimization of mobile computing devices and the extraordinary rise of processing power available in mobile laptop computers combine to put more and better computer-based applications into the hands of a growing segment of the population. Mobile devices, such as laptop computers, Pocket PCs, cellular phones, etc., are now easily affordable, and are becoming more popular in everyday life [14]. At the same time, network connectivity options for mobile hosts have grown tremendously. The markets for wireless telephones and communication devices are experiencing rapid growth. Projections have been made that, in nowadays there are more than billion wireless devices in use. With the availability of mobile computing devices, users often have a natural tendency to share information between them, even though it is not planned in advance and there is no infra structure available for connection, for example, workers at rescue scenes

and employees in a meeting room, or conference or business. Therefore, the wireless mobile ad hoc networks become the practical and conventional solution in such like situations, without requiring each user to connect to the internet or to a wide-area network to communicate with each other because of cost and time. This type of network is easy and fast of deployment, where the nodes are communicate with each other through wireless medium without any fixed infrastructure. Mobile ad-hoc network was also being named as MANET [1] by IETF The Internet Engineering Task Force (IETF) is a large open international community of network designers, operators, vendors, and researchers.

A wireless ad hoc network as a decentralizing network offers an easy and fast connection between collection of autonomous nodes or terminals by forming a multi hop radio network. Since the nodes communicate over wireless links, they have to contend with the effects of radio communication, such as noise, fading, and interference. In addition, the links typically have less bandwidth than in a wired network. Each node in a wireless ad hoc network functions as both a host and a router, and the control of the network is distributed among the nodes [11] [12].

In general, MANET topology is dynamic, because of nodes departure and new nodes arrival during the connectivity time among the nodes, and asymmetrical, because the nodes communicate over wireless links which forms a different transmission range. Hence, there is a need for efficient routing protocols to offer optimum routes during the network establishing time to allow the network nodes to communicate over multi hop paths. Some of MNET features are characteristic of the type of packet radio networks that were studied extensively in the 1970s and 1980s. In general, a multi-hop routing protocol is needed in a mobile ad hoc network, because two hosts wishing to exchange packets may not be able to communicate directly with each other because they are out of radio range [14]. Figure (1) shows a simple ad hoc network of four mobile nodes using different wireless transmission range interfaces. Node A and D are not included within the wireless transmission range of node C. Only node A is included within the wireless transmission range of node D, and node D is not included within the transmission range of node B, as indicated by the circle around A, B and C. Nodes B, C and D are all included within the wireless transmission range of node A. If B and D want to communicate with each other by exchanging packets, they may ask node A to forward packets for them because node A is within the overlapped wireless transmission range between node B and node D.

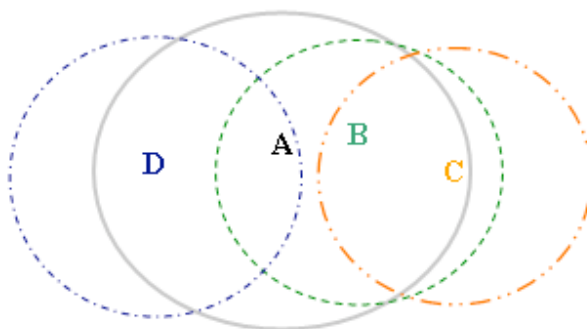


Fig (1): MANET with four wireless Mobile Nodes

In any practical MANET, packets are travel over one or more hops from one node to another node as demonstrated in Figure (1). In reality, the routing problem may be even more complex than this example, because of the nodes different wireless transmission range, and the network topology features which is dynamic because any or all of the nodes associated with the network may move at any time [14]. To provide routing the conventional way in MANET is to make each mobile node take the role as a router, and apply an existing routing protocol between them [16][17]. The fundamental difference between MANETs and the traditional wired networks that is the wired networks topology is stationary and static. This made the traditional protocols such as TCP/IP are not suitable for MANE and leads us for a specific requirement and constraints to provide routing protocols in such dynamic environments.

Over the few last years, many routing protocols have been proposed, where most of these protocols are based on distance vector or link state algorithms. In distance-vector routing protocols (i.e. DSDV) [4], routing information are periodically advertised to all nodes to get an up-to-date view over the entire network. Each node during the network establishing time sends to and receives from, all its neighbor nodes periodic messages and routing information to build and update its routing table, which contains the distance from itself to all possible destinations. Each node can decide whether to keep or update the next hop as the best and shortest path from itself to the specified destination based on comparison of the distances obtained from its neighbors. When each node has a packet to send to some destination, it simply forwards the packet to the decided next hop router. The advantage of this approach is that routes between arbitrary source - destination pairs are readily available, all the time, while the disadvantages are that the routing tables will occupy a large amount of space if the network is large, and that the updates may lead to inefficient usage of network resources if they occur too frequently.

Since ad-hoc networks are bandwidth limited and their topology changes often, an Optimized Link-State Protocol (OLSR) [5] has been proposed. While being suitable for small networks, some scalability problems can be seen on larger networks. The need to improve convergence and reduce control traffic has led to algorithms that combine features of distance-vector and link-state schemes. Such a protocol is the wireless routing protocol (WRP) [10], which eliminates the counting-to-infinity problem and avoids temporary loop without increasing the amount of control traffic. [11, 12]

In addition to the view point categorizing routing protocols in terms of either distance vector or link state routing, routing protocols for MANET also can be classified as uniform, non uniform or reactive routing protocols versus proactive routing protocols. In the reactive routing approach, a node initiates a route discovery (Route requisite) only when want to communicate with a destination which has no available route to it in its routing table, in other words, a routing protocol does not initiate route request until it is needed (Route On Demand). AODV [2], DSR [14], and TORA [13] are the most famous reactive routing protocols for MANET. The disadvantages of such algorithms are high latency time in route finding and excessive flooding can lead to network clogging (Blocking). On the contrary, the proactive routing approach is based on the exchange of knowledge of network topology periodically [9]. The proactive protocols provide a

needed route instantly at the expense of bandwidth because of transmitting periodic updates of topology frequently.

Hybrid routing protocols also exist and they try to achieve an efficient balance between both categories of protocols, where combining both the proactive and the reactive approach. ZRP is an example of hybrid routing protocols, was introduced in 1997 by Haas and Pearlman [6][7]. A more fine grained classification of ad-hoc routing protocols and taxonomy for comparing them can be found in [15].

I. SYMMTRIC AND ASYMMTRIC NETWORKS

In a symmetric computer network, all nodes can transmit and receive data at equal rates. Asymmetric networks, on the other hand, support disproportionately more bandwidth in one direction than the other. This can be a problem in wireless networks which adopt a TCP technique where TCP relies on ACKs for reliable delivery and for congestion control. If ACKs are not reliably returned the smooth of packets will be disrupted by retransmissions. Most of ad hoc networks protocols have been designed assuming that the underlying technology was bidirectional (*Symmetrical Network*). As an example, a set of nodes which are connected through a single physical network assume they can exchange routing information with each other as shown in figure (2). Exchanging routing information enables the discovery of the underlying network topology, and the routing traffic via discovered networks.

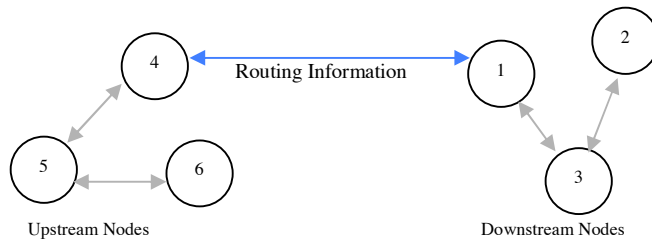


Fig (2): A Symmetric Network

However, if the link connecting these nodes is unidirectional (*Asymmetrical Network*), we can say that all downstream nodes have received only capabilities and therefore cannot send routing information to upstream nodes as shown in figure (3). As a result, upstream nodes cannot discover downstream network topologies dynamically and will therefore never forward information towards them.

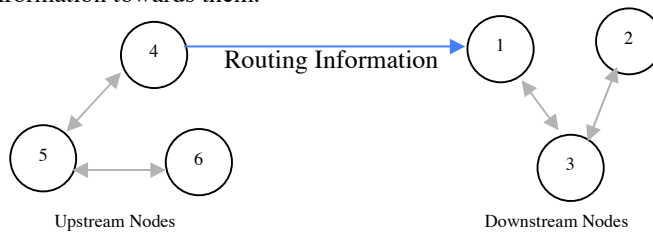


Fig (3): An Asymmetric Network

Generally, in the presence of a unidirectional link, many routing protocols, will fail to operate and lose to send data therefore, to provide full network connectivity we need to make the node to discover if the link is a bidirectional or unidirectional link before sending over any data.

II. MDVZRPA: MOTIVATION

A node in MDVZRPA has a flat view over the entire network when it joins the network and broadcasts a beacon message for the first time. It is easy to get ready multipath to each destination in the entire network by unicasting and receiving routing information (*full dump*) from all its bidirectional neighbours to build its own routing table. Since all nodes proactively store local routing information, route requests can be more efficiently performed without querying all the network nodes. In case of receiving an error message regarding to a broken active link or non reachable node, MDVZRPA uses an alternative path getting technique to get a suitable alternative path (*Best metric*) among the multipath that were stored into the node routing table, instead of wasting time in route repair or route request every time. Also MDVZRPA uses a field called *Symmetric* to distinguish between unidirectional and bidirectional links. Once a node receives the *Hello* message from a new node, it adds an entry in its routing table to this destination (new node) assuming the link between them as an unidirectional link by resetting the *Symmetric* field=0. The *Symmetric* field is set to 1 when a routing information is received from the new node, then the link is considered as a bidirectional link.

III. MDVZRPA: ZONE RADIUS IN AN ASYMMETRICAL NETWORK

The zone radius is the distance in number of hops from the specific node to the last node in its zone. A routing zone is defined for each node separately, and the zones of neighboring nodes overlap. The routing zone has a radius R expressed in hops. The zone thus includes the nodes whose distance from the node in equation is at most R hops. Figure (4) shows a new node (i.e. node 7), and its routing zone when it joined the network. Each node has only one hop from the new node we call it a 1st hop neighbour where radius $R=1$ (i.e. 4, 5 and 6), while any node has 2 hops from the new node, we call it 2nd hop neighbour where $R=2$ and so on. If a node has number of hops (*Distance*) from source node = zone radius, then we call it a peripheral node (i.e. 2, 3). All the rest nodes which have distance $> R$ (i.e. 1) are called *out of zone* nodes.

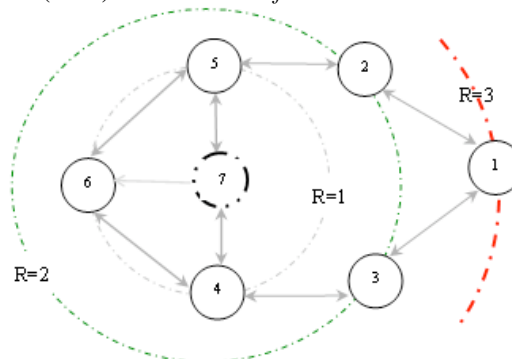


Fig (4): Zone Radius, where $R=1, 2$ or 3 in an Asymmetrical Network

IV. MDVZRPA: ROUTING INITIALIZATION

During the initialization stage as each node joins the network, it adds an entry to itself in its routing table, and broadcasts a periodic beacon (*Hello message*). In figure (5) we assume node 6 is a new node joined an asymmetrical network, where the node zone radius is 2.

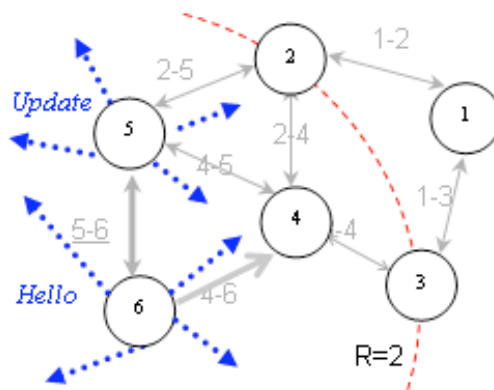


Fig (5): A Hello Message along an Asymmetrical Network

A node which receives the *Hello* message (i.e. 4, 5) checks if it has a direct route (*hop=1*) to the *Hello* message sender. If so, it updates the entry regarding to the next *Hello* message expecting time (*time-out field*), and discards the *Hello* message (*the neighbor is still there*). Otherwise, it adds a new route entry where the route *destination* and 1st *hop* fields are the address of the node that sent the *Hello* message as shown in Table (1), while the *link-id* field is the *Hello* message receiver - *Hello* message sender addresses (4-6, 5-6), sets number of hops field (*Metric*) to 1, sets time-out field, and resets the *Symmetric* field assuming the link between them as an unidirectional link (unusable) at the beginning. Then, it unicasts its routing information to the new node (*Full dump*), if a full dump back message is received from the new node, the node sets the *Symmetric* field considering the link between them as a bidirectional (usable) and broadcasts an *Update* route message to its neighbors regarding to the new node. Table (1) shows new routes to the new node (6) added by node 4 and 5 in their routing table respectively.

Node 4:

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric
6	6	-	1	4-6	0

Node 5:

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric
6	6	-	1	5-6	0

Table (1): Routes are obtained after receiving a *Hello* message

In case of node 4, the link 4-6 is unidirectional as shown in figure (5). Therefore, node 6 will not receive routing information (*full dump*) message from node 4, it is still unknown for node 6. Hence, node 6 will not send back its routing information to node 4.

In case of node 5, the link 5-6 is bidirectional as shown in figure (5). Therefore, node 6 will receive the routing information (*full dump*) from node 5, and unicasts its routing information (*full dump*) to node 5. Node 5 sets the *Symmetric* field, considering the link between them as a bidirectional and broadcasts an *Update* message regarding to the new node to its 1ST hop neighbors (node 2, 4) as shown in figure (5), where these nodes are the 2nd hop neighbors of the new node using node 5 as a 1st hop.

The new node 2nd hop neighbours add an entry in their routing tables and discard the *Update* message, where the *destination* and 2nd hop fields of the entry are the address of the node that sent the *Hello* message (node 6), the 1st hop field is the address of the node that sent the *Update* message (node 5), while the *link-id* is the same as the link id included in the *Update* message 5-6, and the metric is incremented by 1 as shown in table (2). The *Update* message is discarded and not propagated when the metric equals the zone radius (**R**).

Node 2:

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric
6	5	6	2	5-6	1

Node 4:

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric
6	6	-	1	4-6	0
6	5	6	2	5-6	1

Table (2): Routes are obtained from the *Update* message sent by node 5

Further more, Node 6 continues to broadcast a periodical beacon (*Hello*) from time to time. Each time node 5 receives this beacon, it finds a direct route (*hop=1*) in its routing table to node 6 where the link is a bidirectional (usable), therefore, it only updates the entry belongs to node 6 regarding to the next *Hello* message expecting time (*time-out field*) and discards the message. Also, node 4 finds a direct route (*hop=1*) to node 6 where the link is unidirectional (unusable), therefore, it checks if it has indirect route to node 6. If indirect route is found, such as in this case where node 4 has a route to node 6 through node 5 as shown in table (2), it updates the entry belongs to node 6 regarding to time-out field and unicasts a routing information message to node 6 using that route. Otherwise, it only updates the time-out field and discards the message.

Once node 6 received the routing information of node 4, it adds a direct route where *Symmetric* field =1(usable), without paying attention to the next *Hello* message expecting time (*time-out field*) of node 4. In other words, node 6 doesn't consider node 4 as a neighbor.

The new node immediately starts to build its routing table entry by entry, excluding any similar, long and joint paths, once received a routing information message. Tables 3-8 are the routing tables of the entire network nodes after the new node joined the network.

Destination	1 st hop	2 nd hop	Metric	link-id	Symmetric	Notes
1	1	-	0	1-1	1	Initialization
2	2	-	1	1-2	1	...
3	3	-	1	1-3	1	...
4	2	4	2	2-4	1	Multipath
4	3	4	2	3-4	1	Multipath
5	2	4	3	4-5	1	Update

Table (3): Routing table of node 1

Destination	1 st hop	2 nd hop	Metric	link-id	Symmetric	Notes
1	1	-	1	1-2	1	...
2	2	-	0	2-2	1	Initialization
3	1	3	2	1-3	1	...
3	4	3	2	3-4	1	...
4	4	-	1	2-4	1	...
4	5	4	2	4-5	1	...
5	4	5	2	4-5		...
6	5	6	2	5-6		Update

Table (4): Routing table of node 2 after receiving the *Update* messages

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric	Notes
1	1	-	1	1-3	1	...
2	1	2	2	1-2	1	...
2	4	2	2	2-4	1	...
3	3	-	0	3-3	1	Initialization
4	4	-	1	3-4	1	...
5	1	2	3	2-5	1	...
5	4	5	2	4-5	1	...

Table (5): Routing table of node 3 after receiving the *Update* messages

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric	Notes
1	2	1	2	1-2	1
1	3	1	2	1-3	1
2	2	-	1	2-4	1
3	3	-	1	3-4	1
4	4	-	0	4-4	1	Initialization
5	5	-	1	4-5	1
6	5	6	2	5-6	1	Update
6	6	6	1	4-6	0	Hello

Table (6): Routing table of node 4 after receiving the *Update* messages

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric	Notes
1	2	1	2	1-2	1	...
2	2	-	1	2-5	1
3	4	3	2	3-4	1
4	4	-	1	4-5	1
4	6	4	2	4-6	1	Update 6
5	5	-	0	5-5	1	Initialization
6	6	-	1	5-6	1	Hello

Table (7): Routing table of node 5 after receiving the *Update* messages

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric	Notes
1	5	2	3	1-2	1	Full dump 5
2	4	2	2	2-4	1	Full dump 4
2	5	2	2	2-5	1	Full dump 5
3	4	3	2	3-4	1	Full dump 4
4	4	-	1	4-6	1	Full dump 4
4	5	4	2	4-5	1	Full dump 5
5	5	-	1	5-6	1	Hello
5	4	5	2	4-5	1	Full dump 4
6	6	-	1	6-6	1	Initialization

Table (8): Routing table of the new node (6) after receiving the full dump

From the previous tables we can see that each node has one or more entries for each destination in the network, except node 1 and 3. They have no routes to node 6, because both are outside of node 6 routing zone. Therefore, both should initiate a route request *on demand* when need to send data to node 6.

V. MDVZRPA: ROUTE ON DEMAND

If a node needs to communicate with another node in the network and it has no route available in its routing table to that node because it is outside its routing zone, in this example we assume that node 3 as a *Source* node needs to communicate with node 6 as a *Destination*. The *S* node broadcasts a route request message *RREQ* with the *D* address to find a route to the required destination, as shown in figure (6). A route can be determined when the route request *RREQ* reaches a node that offers accessibility to the destination, (e.g., *one of the destination's 1st hop neighbours node 1, 4 or one of the peripherals nodes 2, 5*). As shown in figure (6) and from table (6), node 4 has two routes in its routing table to the Destination. The shortest route is in 1 hop distance from node 6, where *link-id* is 4-6, but it is asymmetric link (*Symmetric=0*), therefore it is useless. The second route is longer, in 2 hops distance from node 6, where *link-id* is 5-6. It is better because the link is symmetric (*Symmetric=1*). Also from table (4) node 2 has a route in 2 hops distance to the destination. The route is made available by unicasting a *RREP* back to the source node and is written in its routing table. The source node has got two routes to the required

destination the first rout in 3 hops distance through node 4 as a 1st hop, node 5 as a 2nd hop where the link-id is 5-6. The second route in 4 hops distance through node 1 as a 1st hop, node 2 as a 2nd hops and the link-id is 5-6. But because the two routes are node joint (node 5) and link joint (5-6), then node 3 chooses only one of them, the shortest one as shown in table (9).

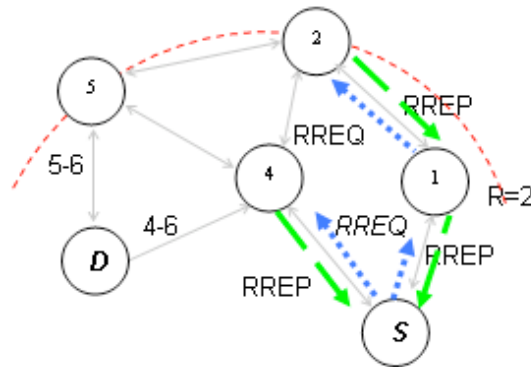


Fig (6): Route on Demand along an Asymmetrical Network

Destina tion	1 st hop	2 nd hop	Metric	Link-id	Symme tric	Notes
1	1	-	1	1-3	1	...
2	1	2	2	1-2	1	...
2	4	2	2	2-4	1	...
3	3	-	0	3-3	1	Initialization
4	4	-	1	3-4	1	...
5	1	2	3	2-5	1	...
5	4	5	2	4-5	1	...
6	4	5	3	5-6	1	RREP

Table (9): Routing table of node 3 after RREP message

VI. MDVZRPA: BROKEN LINKS DISCOVERY

A node discovers the broken link between itself and its neighbour using the *time-out* field. It is the time in which the node expects to receive a Hello message from that neighbor confirming that it is still exists (reachable). This field is set to the expected next hello message time once the node received the *Hello* message from that neighbor. If the node didn't receive a *Hello* message in the expected time then, the node considers that neighbor unreachable. In this case the node assigns any entry in its routing table where the *destination* or *1st hop* fields are equal to the address of that neighbor as a broken link, by setting the *Metric* field to infinity as shown in tables (10, 11), and then generates and broadcasts a route error message (*RERR*) carries the *link-id* of the broken link. Each node receives this message, checks if it has any entry with the same *link-id* to assign it as a

broken link as shown in tables (12, 13). The next section is an example for node movement and broken link in details.

VII. MDVZRPA: NODE MOVEMENT

Node movement is one of the biggest challenges in MANET, where mobile nodes cause broken links as they move from place to place. Any node that discovers a broken link should generate and broadcast a forwarded route error *RERR*, where the nodes update their routing tables regarding to that error message. In figure (7), we assume that node 6 has moved away, both node 5 and 6 discovered that the link-id between them is broken, each node (5, 6) searches for the direct route to the other in its routing table to get its *link-id* and assigns it as a broken link, assigns any route it has where the 1st hop field is the non reachable node as shown in tables (10, 11) and then, broadcasts *RERR* message carrying the non reachable node address with the link-id (5-6) to be assigned by any neighbor it has a route carrying the same link-id in its routing table.

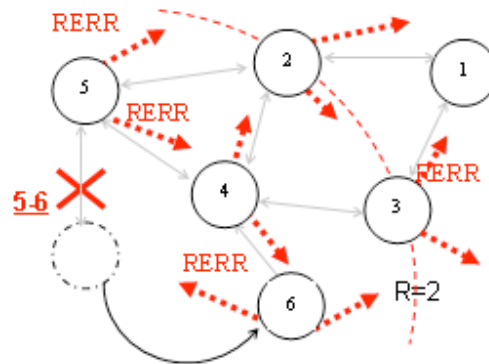


Fig (7) Route Error Message along an Asymmetrical Network

Each node receives *RERR* message, assigns any entry in its routing table with the same link-id as shown in tables(12, 13), and rebroadcasting the same error message *RERR* and so on, unless the node has no route carrying that link-id, in this case it discards the *RERR* message.

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric	Notes
1	2	1	2	1-2	1	...
2	2	-	1	2-5	1	...
3	4	3	2	3-4	1	...
4	4	-	1	4-5	1	...
4	6	4	∞	4-6	1	Deleted
5	5	-	0	5-5	1	Initialization
6	6	-	∞	5-6	1	Deleted

Table (10): Routing table of node 5 after discovering the broken link

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric	Notes
1	5	2	∞	1-2	1	Deleted
2	4	2	2	2-4	1	Full dump 4
2	5	2	∞	2-5	1	Deleted
3	4	3	2	3-4	1	Full dump 4
4	4	-	1	4-6	1	Full dump 4
4	5	4	∞	4-5	1	Deleted
5	5	-	∞	5-6	1	Deleted
5	4	5	2	4-5	1	Full dump 4
6	6	-	1	6-6	1	Initialization

Table (11): Routing table of node 6 after discovering the broken link

Destination	1 st hop	2 nd hop	Metric	link-id	Symmetric	Notes
1	1	-	1	1-2	1	...
2	2	-	0	2-2	1	Initialization
3	1	3	2	1-3	1	...
3	4	3	2	3-4	1	...
4	4	-	1	2-4	1	...
4	5	4	2	4-5	1	...
5	4	5	2	4-5		...
6	5	6	2	5-6		Deleted

Table (12): Routing table of node 2 after receiving the *Update* messages

Destination	1 st hop	2 nd hop	Metric	Link-id	Symmetric	Notes
1	2	1	2	1-2	1
1	3	1	2	1-3	1
2	2	-	1	2-4	1
3	3	-	1	3-4	1
4	4	-	0	4-4	1	Initialization
5	5	-	1	4-5	1
6	5	6	∞	5-6	1	Deleted
6	6	6	1	4-6	0	Hello

Table (13): Routing table of node 4 after receiving the *RERR* message

VIII. CONCLUSIONS AND FUTURE WORK:

In this paper, we proposed MDVZRPA a multipath routing protocol for asymmetric mobile ad-hoc networks. It is a development to our previous protocol MDVZRP [8] for symmetric mobile ad-hoc networks, which is extend to our MDSDV [3] protocol. MDVZRPA is a proactive for all destinations inside the routing zone and reactive for destinations outside the routing zone, supports both bidirectional and unidirectional links. It uses broadcast and unicast techniques to send the packets over both links. Nodes create their routing tables to save multi optimum paths using *Hello* and *Full dump* messages, where maximum number of optimum routes depends on number of neighbors of the

source node. Nodes update their routing tables using Update route, Route request (*RREQ*), Route replay (*RREP*) and Route Error (*RERR*).

If a node wants to initiate communication with a node to which it has no route, MDVZRPA will try to establish such a route using route request mechanism. The protocol allows sending packets by alternative paths in case of the primary path breaks; when a node discovers a broken link to one of its neighbors, broadcasts an error message using the *link-id* to identify the unreachable node. Any node receives the broken link error assigns the right route to that node using the *link-id* included in the error message as unusable route (∞). MDVZRPA gives the node ability to get information from any route pass through it.

In our future work, we are going to evaluate and compare MDVZRPA to DSDV, AODV and ZRP according to the following evaluation metrics (data throughput, packet delivery ratio, routing overhead and average packet delay).

REFERENCES:

- [1] Joseph E Macker, M. Scott.: Mobile Ad Hd Networking and the IETF,Information Technology Division, Naval Research Laboratory Washington, DC, USA Institute for Systems Research, University of Maryland, College Park, MD, USA
- [2] Perkins, C. E., Royer, E. M.: Ad-hoc On-Demand Distance Vector Routing, February 1999, Proc. 2nd IEEE Workshop on Mobile Computer Systems and Applications, pp. 90-100
- [3] Peter J.B King, A. Etorban and Idris Skloul Ibrahim: Multipath Destination Sequenced Distance Vector (MDSDV),8th PG Net, Liverpool John Moores University,Uk 28-29 June 2007, pp. 93-98.
- [4] Perkins, C. E., Bhagwat, P.: Highly Dynamic Destination-sequenced Distance-Vector Routing (DSDV) for Mobile Computers, October 1994, Computer Communications, pp. 234-244
- [5] Clausen, T., Jacquet, P., Laouti, A., Minet, P., Muhlethaler, P., Qayyum,A., Viennot, L.: "Optimized Link State Routing Protocol", September 2001, IETF Internet Draft, draft-ietfmanet-olsr-06.txt
- [6] Nicklas Bejar.: Zone Routing Protocol (ZRP),Networking laboratory, Helsinki University of Technology P.O. Box 3000, FIN-02015 HUT, Finland,Nicklas.Bejar@hut.fi
- [7] Haas, Zygmunt J., Pearlman, Marc R.: A New Routing Protocol for The Reconfigurable Wireless Networks, Proceedings of 6th IEEE International Conference on Universal Personal Communications,IEEEICUPC'97.San Diego, California, USA October 12-16, 1997
- [8] Idris Skloul Ibrahim, A. Etorban and Dr Peter J.B King: Multipath Distance Vector Zone Routing Protocol (MDVZRP),9th PG Net, Liverpool John Moores University,Uk 23-24 June 2008.
- [9] Jacquet and Paul Mühlethaler and Thomas Clausen and Anis Laouti and Amir Qayyum and Laurent Viennot: Optimized Link State Routing Protocol, IEEE INMIC 01, 28-30 December 2001, Lahore, Pakistan, pp. 62-68,
- [10] Murthy, S., Garcia-Luna-Aceves, J. J.: An Efficient Routing Protocol for Wireless Networks, October 1996, MONET, Vol 1, No 2, pp. 183-197
- [11] Haas, Zygmunt J., Pearlman, Marc R.: Providing Ad-hoc connectivity With Reconfigurable Wireless Networks, Ithaca, New York, May 1998.
- [12] Pearlman, Marc R., Haas, Zygmunt J.: Determining the Optimal Configuration for the Zone Routing Protocol, August 1999, IEEE Journal on Selected Areas in Communications, Vol. 17, No. 8,pp.1395-1414
- [13] Park, V. D., Corson, M. S.: A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks, April 1997, Proc. IEEE INFOCOM '97, Kobe, Japan
- [14] Johnson, D. B.,Maltz, D. A.: Dynamic Source Routing in Ad-Hoc wireless Networking, Mobile Computing, T. Imielinski and H. Korth, Eds. Norwell, MA: Kluwer, 1996, pp. 153-181

- [15] L. M. Feeney. A Taxonomy for Routing Protocols in Mobile Ad Hoc Networks, October 1999, SICS Technical Report
- [16] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications, London, UK, Aug. 1994, pp. 234–244.
- [17] J. Jubin and J. D. Tornow, "The DARPA packet radio network protocols," Proc. IEEE, vol. 75,no. 1, pp. 21–32, Jan. 1987.