A New Polymorphic Multicast Routing Protocol for MANET

Adel Ben Mnaouer, Lei Chen, Chuan Heng Foh and Juki Wirawan Tantra
School of Computer Engineering
Center of MultiMedia and and Network Technology
Nanyang technological University
Singapore 637798
Email: {adelm, chenlei}@pmail.ntu.edu.sg, aschfoh@ntu.edu.sg, jw.tantra@ieee.org

Abstract— **We propose in this paper, a polymorphic, hybrid multicast routing protocol for MANET. This new polymorphic protocol attempts to take benefit of the high efficiency of proactive behavior and the low cost of network traffic of the reactive behavior. The protocol is based on the principle of adaptability and multi-behavioral modes of operations. The proposed protocol is able to change behavior in different situations in order to improve certain metrics like the battery power. Extensive simulations demonstrated the effectiveness of the proposed protocol in improving performance and in extending battery power longevity.**

Keywords— **polymorphic, hybrid, multicast routing, MANET**

I. Introduction

Multicast routing protocols used in ad hoc networks can have either a proactive behavior (constantly maintaining and updating routing state information), a reactive behavior (routing paths are constructed only on demand) or a Hybrid behavior (combining both types of behaviors). The hybrid approach tries to benefit from the virtues of proactive and reactive protocol features to perform better.

Proactive protocols include Dynamic Sequenced Distance-Vector Routing (*DSDV*)[11], Source-Tree Adaptive Routing (*STAR*)[5], Multicast Optimized Link State Routing (*MOLSR*)[6]. Reactive protocols include Ad hoc On-Demand Distance Vector Routing (*AODV*)[10], Dynamic Source Routing (*DSR*)[2], etc. Some well-known hybrid multicast routing protocol include the Multicast Zone Routing Protocol (*MZR*)[3], the Mobility-based Hybrid Multicast Routing (*MHMR*)[1], and Fisheye State Routing (*FSR*)[9].

Most of the proposed multicast routing protocols are mainly concerned about achieving high delivery ratios and low latencies. Other protocols tried to incorporate the power issue in order to achieve better power management [12] [13] [7]. However, in the future we need a different approach that strives to find the best compromises that would allow to achieve the best possible trade offs between the above factors. Or even better, future routing protocols should acquire more intelligence and dynamism in order to adapt to different situations with the purpose of maximizing certain factors in specific situation.

Thus, the objective of this research is to propose the design of a new power-efficient, hybrid and polymorphic multicast routing protocol for Mobile Ad Hoc networks. Polymorphic protocols represents the next step beyond hybrid or adaptive protocols. They take advantage of the proven features of proactive and reactive protocol behavior to adapt to different environmental situation and power resources to ensure better performance and better power life longevity. Under the rules of these protocols nodes will change behavior between proactive and reactive operation modes to satisfy different needs in different situations. In their mature state, nodes may be able to implements various protocols to adapt to different situations and may be called to synchronize their behaviors with their peers or to perform simple protocol conversions again for the sake of synchronization purpose.

In our proposal each Mobile Node (MN) will examine its own status, in terms of mobility speed, power level, space density or traffic density, to determine its behavior and use corresponding information to perform the routing procedure. When a MN detects any change of one of the above factors, it will modify its behavior to adapt to that change.

The proposed concept is generic in nature and the choice of the right behavior to implement is a matter of protocol design and depends on the adequacy of the specific protocol behavior to the specific deployment situation.

Section 2 describes the proposed protocol behavior. Section 3 presents the simulation process and discusses the main findings. Concluding remarks are given in Section 4.

II. Protocol Descriptions

First, we present the polymorphic protocol concept (i.e., its main algorithm) that defines node's behavior under different situations.

A. The Polymorphic Algorithm

In this proposed concept of the protocol we build the polymorphic behavior of the protocol based on the battery power levels and mobility speed levels. We define behavioral modes in which a MN will operate depending on specific threshold levels for the battery or the mobility speed. These are as follows:

Definitions:

The behavioral modes of a MN include: a Proactive Mode 1 (*PM1*), a Proactive Mode 2 (*PM2*), a Reactive Mode (*RM*), and a Proactive Ready Mode (*PRM*).

IN IEEE ICC 2005

Proactive Mode 1 (PM1):

In this mode, a node periodically sends out update packets to maintain the topology information. A fixed Time-To-Live (TTL) value is set to the update packet. When the node received update information from other nodes, it will update its Neighborhood Routing Table (NRT) using the information received. The node in this mode will propagate the received update packets if the TTL is not zero. The update interval is set to j, where j is a fixed time window.

Proactive Mode 2 (PM2):

In this mode, a node performs similar to *PM1*, while the update interval is set to $n \times j$, where j is a fixed time interval and n is an integer that is greater than one.

Reactive Mode (RM):

In this mode, a node will not periodically send out update packets and when receiving update packets from other nodes, it will just discard them.

Proactive Ready Mode (PRM):

In this mode, a node will not periodically send out update packets but updates its NRT upon receiving the update packets from other nodes. The node propagates the update packets received if the TTL is not zero.

Next, we define two power level thresholds P_TH1 and P_TH2, where $P\_TH1 > P\_TH2$. We also define a Mobility Level Threshold M_TH and Vicinity Density Level Threshold V_TH.

The Algorithm:

The Polymorphic algorithm includes two parts, the main algorithm and the mobility speed routine. We assume here that each node is equipped with means of detecting its current mobility speed level.

Algorithm 1 Polymorphic Algorithm

```
if Power > P_TH1 then
  If the node is not in PM1, it switches to PM1.
  Then, notifies neighbors about mode switch.
else
  if Power < P_TH2 then
    If the node is not in RM, it switches to RM.
    Then, notifies neighbors about mode switch.
  else
    Perform the mobility speed routine.
  end if
end if
```

When node's power level is high ($> P\_TH1$), the node is set to PM1 mode, so that it can be able to maintain topology information and react faster to topology changes. Thus, the nodes are allowed to operate in PM1 mode only if the power level is high enough.

On the other hand, when the node's power level is quite low ($< P\_TH2$), the node is forced into the RM mode in order to extend its battery life.

When the node's power level is within P_TH1 and P_TH2, the mobility routine is performed to help determining the node's behavior. The mobility routine is described next.

Algorithm 2 Mobility Speed Routine

```
if Mobility > M_TH then
  if Power > P_TH2 then
    if Vicinity < V_TH then
      If the node is not in PM2, it switches to PM2.
      Then, notifies neighbors about mode switch.
    else
      If the node is not in PRM, it switches to PRM.
      Then, notifies neighbors about mode switch.
    end if
  end if
else
  Node which is not in RM switches to RM.
  Notify the neighbors about the mode switch.
end if
```

If the mobility speed level of the node is quite high, it means that the topology around the node is changing quickly. Thus, the node is required to behave proactively in order to maintain better connectivity and awareness of the topology changes. This is triggered when the node's mobility speed level gets higher than the M_TH threshold.

The next consideration here relates to the node's vicinity density level. When it is high, it means that there are many nodes within the power range of the node. Thus, if we let the node engage in a proactive mode, then update packets would consume the channel capacity and jam the network with higher probability. Thus, when the vicinity density level is high, the node is forced into a reactive behavior.

On Receiving Notification:

When a node switches its behavior, it will broadcast a notification to all its neighbors to inform them about the mode change. When other nodes receive such a notification, they will change the corresponding node entry's lifetime in their NRTs.

When a node receives a notification that the source node is switched from other modes to PM1, the lifetime of the corresponding entry is set to $2 \times j$.

When a node receives a notification that the source node is switched from other modes to PM2, the lifetime of the corresponding entry is set to $3 \times j$.

When a node receives a notification that indicates the source node has switched from other modes to PRM or RM, the lifetime of the corresponding entry is set to $3 \times j$.

We also consider the case when a node is in PM1 or PM2 and it did not hear any update packets within a fixed time interval (all neighbors are in RM), $3 \times j$, the node switches to RM.

B. Routing Issues

We have implemented a routing protocol using the polymorphic algorithm described above. We have selected the Zone Routing protocol from the MZR as the proactive behavior, and selected the On-demand Multicast Routing Protocol (ODMRP)[14] as the basis of reactive behavior. Since we are using ODMRP, we maintain the multicast group topology as a mesh. We call the resulting protocol *P_ZODMRP*.

Routing Tables:

Each node maintains two routing tables, one is the Neighbor Table (NTable), the other is Multicast Routing Table (MRTable).

The NTable acts as the NRT we described in the algorithm, and actually, only nodes in PM or PRM modes maintain it. The main structure is like the zone routing table in the *ZODMRP* [15] . Each entry is for a neighbor node in the zone. Each entry contains the routing information to the node, including hop count and next hop address. In addition, each entry contains the multicast routing information of the node, such as the multicast group that the node belongs to. Each entry is assigned a life time, and stale entries would be removed from the NTable.

Each node should also maintain a multicast routing table (MTable) to maintain its own multicast routing information and to maintain the multicast routing topology. The structure of the MTable is the same as the routing table in *ODMRP*.

Packet Structure:

The packet structure in *P_ZODMRP* is the same as the the packet structure in *ODMRP*, and we add two types of packet, the notification packet, and the update packets.

The structure of the notification packet is very simple.

The first 8 bits is for packet type, and is set to 5 for Notification Packet in *P_ZODMRP*.

The next 16 bits is for the switch type, and each bit if for one type (there are 12 types of switching so that the first 4 bit is always set to 0).

The structure of the update packets includes the packet type, the address of the source node, last hop of the packet, and the multicast information of the source node.

Path Finding Procedure:

When a node have packets to a multicast group or wants to join the multicast group, it begins the path finding procedure.

If the node is in *RM*, it send out a *Join_Request* as the way in *ODMRP* and waiting for the Reply.

If the node is in *PM* or *PRM*, the node first look in its *NTable* to see whether there are nodes that belong to the destination multicast group. If so, the node unicasts *Join_Requests* to all these nodes and waits for replies. Otherwise, the node will broadcast a *Join_Request*.

When a node receives a *Join_Request*, and it is a member of the multicast group, it generates the Reply and sends it back to the source of the *Join_Request*, updating the *MTable* to record the route. If the node could not send a Reply, it checks its own behavior. If it is in *RM*, it just propagate the *Join_Request* and record the *Join_Request* in the route cache. If the node is in the *PM* or *PRM*, the node look in its own *NTable* to find the destination multicast group member. If they are members in its zone, it unicasts the *Join_Request* to all the members. Otherwise, it just propagates the *Join_Request*.

When the source node receives the Reply, it updates its *MTable* to record the route and begins data transmission.

Polymorphic Algorithm Operation:

Each node within the network will periodically examine its power level and determine the corresponding behavior.

If the power level is between P_TH1 and P_TH2, the node examine its mobility speed and vicinity density level to determine its behavior.

If the node have to switch its behavior, the node generates a notification Packet and broadcasts the packet to all its neighbors. The TTL of the notification packet is set to one, and only the one hop neighbor receives the notification packet.

III. Simulation and Results

A. Simulation Scenarios

We have performed two sets of simulations. In the first set, we have varied and evaluated the value of the parameter j to see the effect of the update interval in different situations. In the second set, we have performed a simulation based comparison among the *P_ZODMRP*, *ZODMRP* (Zone based ODMRP), *ZMAODV* (Zone based Multicast AODV) [15] and the *MOLSR*. The value of j is set to 5 for the *P_ZODMRP*.

The simulation of these protocols was implemented using the GloMoSim library[8]. Nodes are placed randomly within a 1000m $\times$ 1000m area. The radio propagation range for each node was set at 225m, and the channel capacity was 2Mbps. The IEEE 802.11 was used as the MAC protocol. The traffic type is Constant bit rate (CBR) protocol. The mobility model used was the random waypoint model. The power model adopted is the one proposed by L.F. Feeney's work[4]. In the second simulation set, the simulation time is set to 1000s and the power level of nodes are set to vary from one node to another. This setting is done to validate the effect of the proposed protocol on the delivery ratio when in the course of the simulation some nodes will die off due to lack of battery power. Thus, we can check the effect of the polymorphic protocol on battery longevity. Thus we set 20% of the nodes to have 100% power, 20% of the nodes have 90% power, 20% of the nodes have 80% power, and 40% of the nodes have 75% power. The following metrics were used in the performance evaluation:

- Packet delivery ratio;
- Number of control packets transferred per data packet delivered;
- Average Percentage of Power Conservation

We have evaluated the above metrics against Mobility speed, Network Traffic Load and the Total Number of Nodes parameters.

B. Results and analysis

1) Mobility Speed

Experimental Scenario:

There are 50 nodes within the area, each node moves constantly with a predefined speed. The node movement speed was varied from 0m/s to 60m/s. In this scenario, we had 20 multicast members and 5 source nodes.

Fig.1 depicts the delivery ratio against mobility speed for different values of the parameter j.It can be seen that the curve

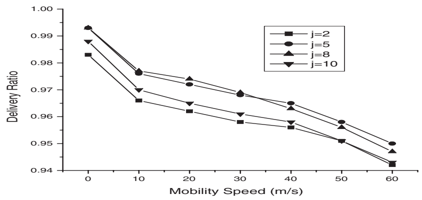

Fig. 1. Delivery ratio Vs. mobility speed

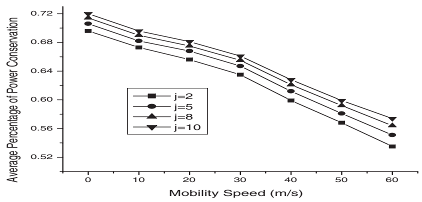

Fig. 2. Average power conservation Vs. mobility speed

for $j = 5$ and $j = 8$, have nearly the same performance. While $j = 2$ or $j = 10$, perform badly. When $j = 2$, the update interval is short and thus many update packets are generated occupying much of the channel capacity and reducing the delivery ratio.

Fig.2 shows the average power conservation as a fucntion of mobility speed. When j is 10, the update interval is longest and could generate least periodical update packets, and thus could save up the power usage. When j is 2, the worst performance is recorded.

Fig.3 plots the delivery ratio as a function of mobility speed. We could see that the *P_ZODMRP* has the best performance, and the performance of *ZODMRP* and the MOLSR are very near to the *P_ZODMRP*. The *ZMAODV* had the poorest performance. All nodes have equal power here.

Fig.4 shows the delivery ratio as a function of mobility speed, when variable power levels and longer simulation is considered. Since some nodes would have used up their battery power and gone off, the delivery ratio decreases greatly. We can see that in this scenario, the *P_ZODMRP* exhibits the best performance. The battery life of nodes using *P_ZODMRP* has been extended so that the delivery ratio could be increased greatly. In other experiments we have found that in terms of end-to-end delay the performance of the *P_ZODMRP* is almost

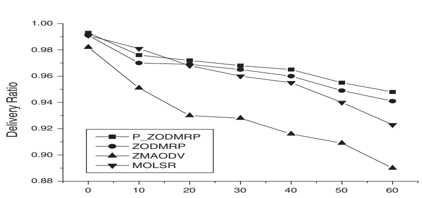

Fig. 3. Delivery ratio Vs. mobility speed

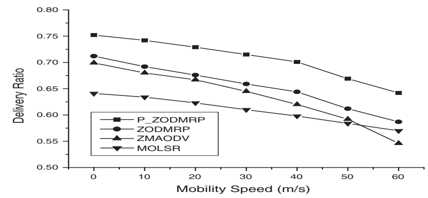

Fig. 4. Delivery ratio Vs. mobility speed

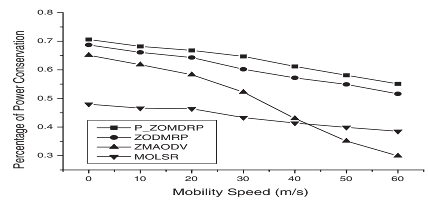

Fig. 5. Average power conservation Vs. mobility speed

comparable to that of the *ZODMRP*, but both achieved better than the ODMRP.

Fig.5 shows the average percentage of power conservation as a function of mobility speed. We can see that *P_ZODMRP* could extend node's battery life better than the other routing protocols. With the increase of the speed, the power usage of *ZMAODV* increases greatly because *ZMAODV* generates more control packets in high speed.

2) Node Vicinity Density

Experimental Scenario:

The total number of nodes within the area varies from 20 to 80, each node moves constantly with a predefined speed 5m/s. In this scenario, we had 15 multicast members and 5 source nodes.

Fig.6 shows the average end-to-end delay as a function of nodes vicinity density for several values of j. The setting os $j = 5$ and $j = 8$ were found to perform better. Moreover, with an increase in density, the setting of $j = 8$ exhibits better performance because it increases the update interval and reduces the control overhead.

Fig.7 shows the average power conservation as a function of nodes vicinity density. Again, since the $j = 10$ has the lowest update frequency, it generates least control overhead and brings the best performance.

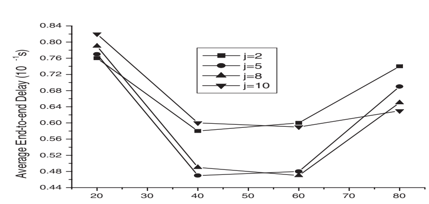

Fig. 6. Average end-to-end delay Vs. node vicinity density

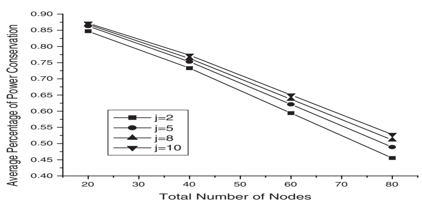

Fig. 7. Average power conservation Vs. node vicinity density

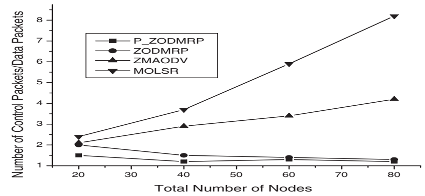

Fig. 10. Number of control packets per data packets Vs. node vicinity density

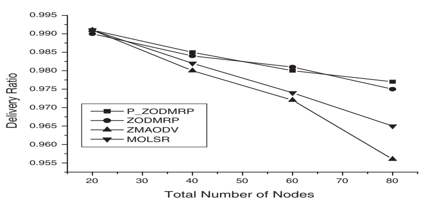

Fig. 8. Delivery ratio Vs. node vicinity density

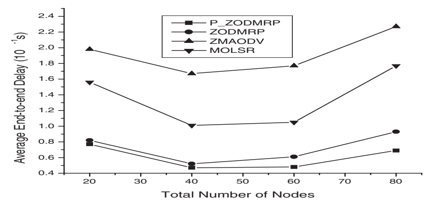

Fig. 11. Average end-to-end delay Vs. node vicinity density

Fig.8 shows the delivery ratio as a function of nodes vicinity density for a simulation time of 500s. We can see that the *P_ZODMRP* and *ZODMRP* achieved the best performance because they are using a mesh topology to maintain multicast group topology. This enabled them to guarantee data packet delivery to destination nodes while ignoring broken link repairs.

Fig.9 shows the delivery ratio as a function of nodes vicinity density for a simulation time of 1000s. It can be sees that the performance of all four protocols could slightly be increased with the increase of the vicinity density. That's because in this scenario, some nodes could use up the battery and turn off. When the number of nodes gets smaller, some nodes get isolated from the others and won't be able to communicate. This decreases the delivery ratio. The *P_ZODMRP* exhibits again the best performance because of its power saving features.

Fig.10 shows the number of control packets per data packets as a function of node vicinity density. Since the *MOLSR* is a pure proactive protocol and its update packets should reach all through the network, the number of control packets increases greatly. In addition, with the increase of the number of nodes, the *MAODV* was found to generate higher control packets.

Fig.11 shows the average end-to-end delay as a function of node vicinity density. When the density is low, some nodes may get isolated and cause an increase in delay. When the density is high, nodes could have more neighbors thus build better path finding information, but would generate more control overheads that will occupy the channel and delay data packets. That is why the average delay of all the four protocols is high when the density is either low or high. Again, the *P_ZODMRP* achieves the best performance by reducing the control packets than the *ZODMRP*.

Fig.12 shows the average power conservation as a function of nodes vicinity density. Since the power model used is mainly concerned with transferred packets, the *MOLSR* used the most power within these protocols because the large amount of periodical updates it generates. *P_ZODMRP* could save in terms of the number of control packets it transferred, and achieved the best performance.

3) Network Traffic Load

Experimental Scenario:

There are 50 nodes within the area and the number of packets the sources send varies from 1 to 50 packets per second, each node moves constantly with a predefined speed 5m/s. In this scenario, we had 20 multicast members and 5 source nodes.

Fig.13 shows the delivery ratio as a function of network traffic load when the simulation time is 1000s. With the in-

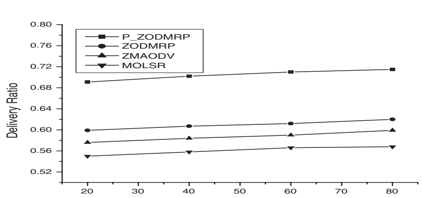

Fig. 9. Delivery ratio Vs. node vicinity density

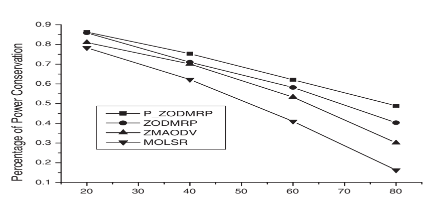

Fig. 12. Average power conservation Vs. node vicinity density

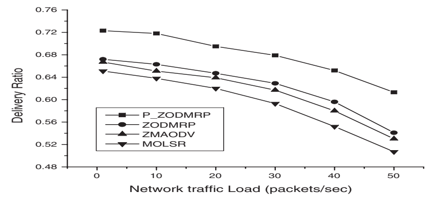

Fig. 13. Delivery ratio Vs. network traffic load

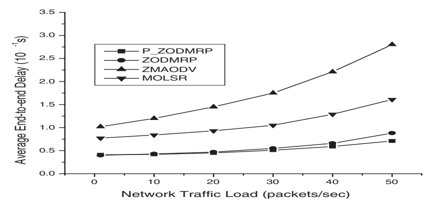

Fig. 14. Average end-to-end delay Vs. network traffic load

crease of the traffic, clear decline of performance is observed. However, the performance of the *P_ZODMRP* is by far the best among all four protocols. The gain is clearer than in the case of equal initial battery power setting (not shown).

Fig.14 shows the average end-to-end delay as a function of network traffic load. We can see that with load increase, the average delay increases in all four protocols. The *ZMAODV* has the poorest performance due to the large amount of control overhead for repairing broken links. When the load is high, the *P_ZODMRP* performs better than *ZODMRP* because its polymorphic features that helps save up control packet overhead on periodical updates.

Fig.15 shows the average power conservation as a function of network traffic load. As the transfer of data packets constitutes the main part of power consumption, in this scenario, the power of nodes in all of the four protocols decreases greatly. Again, with its power saving mechanisms, the *P_ZODMRP* did better than the others.

IV. Conclusion

We have presented in this paper a polymorphic, hybrid multicast routing protocol for MANET. This new polymorphic protocol attempts to exploit the high efficiency of proactive behavior and the low cost of network traffic of the reactive behavior. The concept is generic in nature and the choice of the right protocol (proactive or reactive) to use depends on its proven performance and on its applicability to the situation where the protocol is deployed. We have applied this concept to the *ZODMRP* protocol (proposed earlier) and found that the polymorphic *ZODMRP* (or *P ZODMRP*) is indeed superior to all the other protocols to which it was compared. The superiority lies in the fact that on the long run, the protocol was able to extend battery life and enhance survivability of the mobile Ad Hoc nodes. Hence, it has increased data deliverability ratio while keeping the control packet overhead at acceptable levels.

we think that this new concept of polymorphic protocols constitutes the next trend in the design of efficient multi-behavioral routing protocols for wireless, power-constrained networks such as MANETs. There is a dire need for efforts to fully characterize such protocols, in order to synthesize and efficiently combine the emerging ideas contributing to the betterment of routing protocols in MANETs. That will be our next objective in this research endeavor.

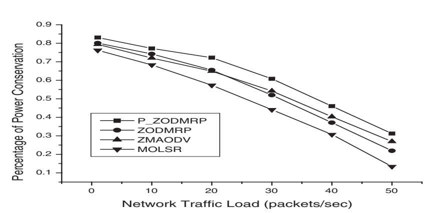

Fig. 15. Average power conservation Vs. network traffic load

References

[1] B. An and S. Papavassiliou. A mobility-based hybrid multicast routing in mobile ad-hoc wireless networks. In *Proc. of IEEE Military Communications Conf. (MILCOM2001)*, Oct. 2001.

[2] Josh Broch, D. B. Johnson, and D. A. Maltz. The dynamic source routing protocol for mobile ad hoc networks. *Internet Draft, draft-ietf-manet-dsr-03.txt*, Oct. 1999.

[3] Vijay Devarapalli and Deepinder Sidhu. Mzr: A multicast protocol for mobile ad hoc networks. In *IEEE International Conference on Communications (ICC)*, Helsinki, Finland, June 2001.

[4] Laura Marie Feeney. An energy consumption model for performance analysis of routing protocols for mobile ad hoc networks. *Mobile Networks and Applications*, 6(3):239-249, 2001.

[5] J. J. Garcia-Luna-Aceves, M. Spohn. Source tree adaptive routing (star) protocol. *Proc. of 7th Intl. Conf. on Network Protocols*, p 273, 1999.

[6] Philippe Jacquet, Pascale Minet, Anis Laouiti, Laurent Viennot, Thomas Clausen, and Cedric Adjih. Multicast optimized link state routing. *Internet Draft, draft-ietf-manet-olsr-molsr-01.txt*, April 2002.

[7] Dongkyun Kim, J.J Garcia-Luna-Aceves, Katia Obraczka, Juan-Carlos Cano, and Pietro Manzoni. Power-aware routing based on the energy drain rate for mobile ad hoc networks. In *Proc. of 11th Intl. Conf. on Computer Communications and Networks*, 2002.

[8] UCLA Computer Science Department Parallel Computing Laboratory Glomosim: a scalable simulation environment for wireless and wired network Systems. http://pcl.cs.ucla.edu/projects/glomosim/.

[9] Guangyu Pei, Mario Gerla, and Tsu-Wei Chen. Fisheye state routing: a routing scheme for ad hoc wireless networks. In *IEEE Intl. Conf. on Communications, 2000*, 2000.

[10] Charles E. Perkins, Elizabeth M. Belding-Royer, and Samir R. Das. Ad hoc on-demand distance vector (aodv) routing. *Internet Draft, draft-ietf-manet-aodv-13.txt*, Feb. 2003.

[11] Charles E. Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing for mobile computers. Computer Communications Review, pp. 224–244, October 1994.

[12] K. Scott and N. Bambos. Routing and channel assignment for low power transmission in pcs. In *IEEE Int'l Conf. Universal Personal Comm.*, 1996.

[13] S. Singh, M. Woo, and C.S. Raghavendra. Power-aware with routing in mobile ad hoc networks. In *Fourth Ann. ACM/IEEE Int'l Conf. Mobile Computing and Networking*, 1998.

[14] Yunjung Yi, Sung-Ju Lee, William Su, and Mario Gerla. On-demand multicast routing protocol (odmrp) for ad hoc networks. *Internet Draft, draft-ietf-manet-odmrp-00.txt*, Mar. 2003.

[15] L. Chen and A. B. Mnaouer. Performance Evaluation of New Hybrid Multicast Routing Protocols for Ad-Hoc Networks. *9th IEEE Intl. Conf. on Communications Systems (ICCS 2004)*, 6-8, Sept. 2004, Singapore.