This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2007 proceedings.

A Novel Scheme to Reduce Control Overhead and Increase Link Duration in Highly Mobile Ad Hoc Networks

Ehssan Sakhaee[†], Tarik Taleb[‡], Abbas Jamalipour[†], Nei Kato,[‡] and Yoshiaki Nemoto[‡]

[†]School of Electrical & Information Engineering, University of Sydney, Australia

[‡]Graduate School of Information Sciences, Tohoku University, Japan

***Abstract*—Flooding-based approaches are incorporated in reactive routing protocols as the fundamental strategy for route discovery. They overtly affect traffic as the frequency of route discovery increases along with the mobility of users in a Mobile Ad Hoc Network (MANET). This paper presents a scheme for reducing overall traffic and end-to-end delay in highly MANET networks. Firstly a new routing algorithm is proposed to reduce the frequency of flood requests by elongating the link duration of the selected paths. In order to increase the path duration, non-disjoint paths are also considered. This concept is a novel approach in route discovery as previous reactive routing protocols seek only disjoint paths. Secondly another novel approach is presented to estimate the Link Expiration Time without the need for Global Positioning System (GPS) devices. To prevent broadcast storms that may be intrigued during the path discovery operation, another scheme is also introduced. The basic concept behind the proposed scheme is to broadcast only specific and well-defined packets, referred to as "best packets" in the paper. The new protocol is simulated with regard to traffic overhead. Although our main aim in this paper is to reduce the net control traffic in a MANET network, there are other benefits arising from the proposed schemes, namely the increase in link duration, reduction in the end-to-end communication delay, less disruption in data flow, and fewer path setups.**

I. INTRODUCTION

Flooding is the most essential mechanism for the route discovery process in reactive (on-demand) Mobile Ad Hoc Networks (MANETs). The number of flood request packets and the frequency of the flooding occurrence greatly affect network performance due to the introduction of additional network traffic into the system and interruption in data transmission. The frequency of route discovery, and hence flood requests, can be linked to another fundamental issue in a MANET: path or link duration, also known as path stability. When a path breaks, not only are portions of data packets lost, but also in many cases there is a significant delay in establishing a new path. This delay depends on whether another valid path already exists (in the case of multi-path routing protocols) or whether a new route discovery process needs to take place. The latter scenario introduces yet another problem. In addition to the delay in discovering new paths, flooding required for path discovery would greatly degrade the throughput of the network as it introduces large amount of network traffic, especially if the flooding is not locally directed, as in the case of Location Aided Routing (LAR) protocols [1]. However if the locations of destination nodes are unknown, omni-directional flooding is inevitably the only option. In a highly mobile system, where link breakage is frequent, flooding requests would largely degrade system performance.

Two methods can be used to decrease overall control overhead in a MANET network. One is by reducing mass broadcasting, known as broadcast storms [2], and the other is by decreasing the frequency of broadcast flooding. The broadcast storms result in heavy traffic contention and consequently collisions of packets due to the mass flooding broadcasts between neighboring nodes.

Limiting flooding has been widely considered in recent literature [3]. Frequency of broadcasting is directly dependent on the route discovery process, and how often this needs to be performed. Multipath routing [4-6] attempts to minimize the need for frequent route discovery processes by selecting an alternate path if one path fails. However if such alternate paths have also expired, a route discovery is inevitability carried out. A notable algorithm that tends to reduce the frequency of route discovery is Associativity Based Routing (ABR) [7] which tries to choose paths that are more stable and have long link durations. However, ABR does not rebroadcast a request packet more than once, in order to retain the property of disjoint paths. Although the property of disjoint path discovery has been previously considered as a correct approach to routing, this paper will show how disjoint path discovery can be quite limiting in certain scenarios, particularly in a pseudo-linear mobile environment. Furthermore this paper will demonstrate how ABR and other path-disjoint routing protocols are disadvantageous in a highly mobile pseudo-linear mobile environment with no pause time, such as an aeronautical ad hoc network introduced in [8].

Other attempts at predicting and selecting stable links have been proposed in [9-11], however they all depend on statistical analysis and probabilistic models of link duration.

The schemes proposed in this paper simultaneously deal with the two concerns of MANETs: link duration and control/broadcast overhead. Additionally, unlike previous algorithms, the proposed routing scheme, dubbed as Receive

On Most Stable Path (ROMSP) updates cost parameters on the reverse (reply) path as the reply packet traverses back to the source node. This is to provide the source node with the most updated information for the selection of the most stable path.

There is also a setup time when a route is not readily available, and packets are placed in a queue until a new path is established for routing [4]. This obviously affects the end-to-end delay and wastes the memory of nodes. Therefore, maintaining paths with long durations would ultimately reduce the frequency of occurrence of the above scenario, as paths do not break as often.

Maintaining a low level of control overhead is a primary goal in reactive routing protocols. In this paper, the proposed schemes will effectively reduce the control overhead in addition to other advantages, such as increased path duration, decreasing end-to-end delays, and effective reduction in overall network traffic. In general, control message overhead increases when nodes are highly mobile, due to the higher rate of link breakage. These overhead messages consist of Route Request (RREQ) messages generated during the route discovery process and Route Error (RERR) packets caused by abrupt link failures. The total amount of control messages in a MANET network can be reduced by four fundamental strategies:

1. Increasing path duration
2. Multipath routing
3. Rebroadcast minimization
4. Route discovery prior to path expiration

The first three scenarios have been dealt with in recent literature. In this paper, we introduce more suitable schemes to deliver more efficient results in highly MANET scenarios. The paper considers also the fourth strategy to reduce route error packets.

The novelty of the proposed routing protocol is that it exploits non-disjoint path discovery in order to effectively find more stable paths for routing. It is worth noting that although the use of non-disjoint paths requires the use of additional control messages, however by using the Forward Best Request (FOBREQ) technique, the added overhead is minimal and the overall reduction in the entire network traffic is significant. Hence the achieved throughput of the network will be more evident than in the case of traditional algorithms that do not take into account mobility, as demonstrated in the simulations of section IV. In the proposed protocol, due to the selection of stable, more durable paths, there will be fewer path breaks and handoffs. This consequently not only reduces the delay between new route establishments, but also causes fewer route discoveries and hence effectively reduces traffic flooding.

In this paper, we use a concept similar in spirit to the multi-path approach presented in [4-6]. However, the main difference is that cached paths are not necessarily disjoint. Indeed, when a path breaks, all paths that contain the broken link are simultaneously purged from the Path Cache, which includes all suitable paths to a destination. This process is described in more details in Section III.

The remainder of the paper is organized in the following fashion. Section II surveys previous attempts of reducing control overhead in MANETs and discusses algorithms that increase link duration. Section III introduces the proposed schemes of this paper and the routing protocol. Section IV simulates the proposed scheme, followed by results and discussions. Finally the paper concludes with a summary recapping the main advantages of the proposed system and future research work in Section V.

II. RELATED WORK

Traditionally, the Dynamic Source Routing (DSR) algorithm [12] is a widely implemented and well known routing algorithm for MANETs. However DSR, does not take into account mobility parameters during route discovery, resulting in paths which break often in highly mobile scenarios, causing excessive broadcasting and flooding the entire network for new routes to be discovered.

Location Aided Routing (LAR) [1], like other broadcast/flood reducing mechanisms [13, 14], directs broadcasting towards the estimated destination node. However if there is no knowledge of the estimated destination, then this kind of mechanism cannot work. In [15] broadcast flood is limited by only forwarding consecutive RREQ packets which have a path hop accumulation smaller than the previous identical or duplicate RREQ packet. Otherwise the newly arrived RREQ packet is dropped and hence not forwarded.

In [2] several approaches are introduced to minimize flooding in MANETs. One is the probabilistic scheme, which rebroadcasts a packet that has been received for the first time with a given probability *P*. In the counter-based approach a counter is incremented each time an identical packet is received, and is used to prevent further re-broadcasting of the same packet coming from different sources once the counter threshold value is reached. The distance-based scheme considers relative distances between the nodes and decides whether it is worth re-broadcasting a message depending on the distance from the source. If the distance from the source is small, then there is no additional coverage benefit from re-broadcasting this message. However, if the message is from a far away source, then re-broadcasting would provide a larger coverage. Consequently, if a message comes from a source which is at a distance greater than d_{min}, then the message is re-broadcasted. The Location-Based Scheme requires a Global Positioning System (GPS) to calculate the area of coverage if a message is re-broadcasted and decides whether it is worth re-broadcasting such a message.

Although the above-mentioned methods are quite satisfactory in providing efficient re-broadcasting with regard to coverage, integrating this broadcast minimizing schemes in routing does not ensure that the need for consecutive broadcasting is decreased, and neither does it consider path stability during the re-broadcasting procedure. Another disadvantage is that in less populated and sparse networks the re-broadcasting of messages is not guaranteed as a result of the threshold values being static [16]. Hence we need a scheme that takes these issues into consideration, whilst reducing broadcast overhead.

A routing algorithm that considers stability in the routing criterion is the Associativity Based Routing (ABR). ABR uses associativity "ticks" messages (TICKs), which are periodically broadcasted in order to estimate lifetime of links. If a node has high associativity ticks with its neighbor node, then the degree of stability (and hence link duration) is high. The destination node chooses nodes on a path which have a high degree of associativity.

If we consider ABR in a pseudo-linear mobile ad hoc network with no pause time, all nodes within a time range would receive *equal* associativity ticks regardless of their speed and direction. In this case, *high associativity* means that the neighbor node has been within range for a considerable period of time. It does not ensure that the mobile node will continue to remain within range, as the mobile node may already be close to the edge of the communication boundary. A better node which provides a more stable link may have just come into the range of the target node, and would consequently have a lower associativity value. Thus ABR would not be suitable for the considered mobility model. Fig. 1 illustrates this idea. Let nodes A and B have higher associativities with S than does C. Applying ABR to such a scenario will lead to the selection of either node A or B for communication. This obviously yields a poor performance of the entire network as nodes A and B will soon disappear from the range of node S. For this reason we introduce a scheme which takes into account the relative velocity and relative distances of nodes during route discovery in order to find most stable paths. Additionally the rebroadcast reduction scheme is based on re-broadcasting best request packets which would ultimately produce more stable links as the eventual path.

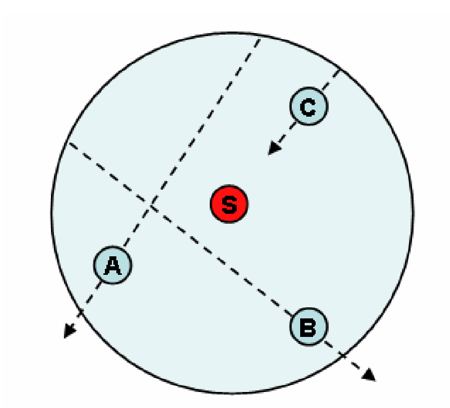

Fig. 1. An example scenario where ABR does not work.

III. PROPOSED ROUTING PROTOCOL FOR DECREASING FLOOD REQUESTS IN MANETS

A. The use of non-disjoint path discovery

In order to describe the significance of non-disjoint path discovery, a scenario shown in Fig. 2 is considered. In this scenario, if a traditional DSR algorithm is used during route discovery, only ABD will proceed to be forwarded in the RREQ packet at D. The duplicate RREQ that was received from B will arrive at D from C, but will be discarded. A DSR mechanism (and all traditional reactive routing algorithms such as AODV) will drop this duplicate packet. However, if we consider the Link Expiration Time (LET) of these two non-disjoint paths, we see the advantage of also rebroadcasting the identical RREQ coming from C. For simplicity, let us first assume all mobile nodes have the same magnitude of velocity with only their directions being different. If we consider the LET on each link on the path, it becomes clear that the Path Expiration Time (PET) would correspond to the minimum LET on a path. i.e. $PET(ABD) = min\{LET_{AB}, LET_{BD}\}$ and $PET(ABCD) = min\{LET_{AB}, LET_{BC}, LET_{CD}\}$. Now looking at each individual link on the two paths and considering the relative velocity of the pairs of nodes for each link, the relative velocity v_R of *node i* with respect to *node i-1* is shown in (1).

$$v_R = v\sqrt{2(1-\cos\alpha)} = 2v\sin\left(\frac{\alpha}{2}\right) \tag{1}$$

where v is the speed of each node, α is the angle between the velocity vectors of the two nodes.

The simplified link expiration time t_c for each link is given in (2), where $d_{initial}$ is the initial distance between the two nodes. Also let's consider all nodes are at the same distance from each other. Hence the factor affecting t_c is the angle α between the nodes.

$$t_c = \frac{d_{LOS} - d_{initial}}{2v\sin\left(\frac{\alpha}{2}\right)} \tag{2}$$

where d_{LOS} is the maximum range, or line-of-sight distance to any node.

We can clearly see that the smallest LET belongs to BD as it has the greatest angular difference α. Hence although path ABD is the shortest path (least hop), its stability is limited by link BD. It can be seen that $min\{LET_{BC}, LET_{CD}\}$ is greater than LET_{BD}. Hence $PET(ABCD) > PET(ABD)$. Thus if we also rebroadcast the duplicate broadcast message coming from C, we can forward the knowledge of a more stable path. This illustrates the advantage of using non-disjoint path discovery. However this will be at the price of additional broadcast flooding, which is minimized with another proposed scheme called the Forward Best Request (FOBREQ), which will be explained in part C.

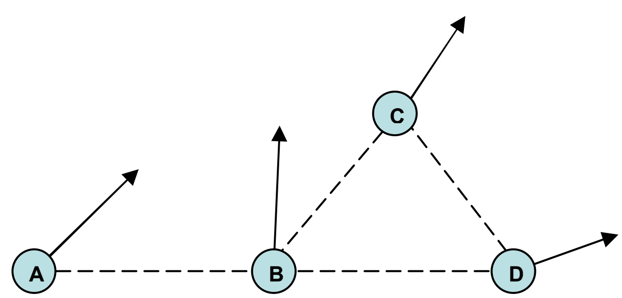

Fig. 2. An example of the use of non-disjoint path discovery.

B. Definition of Best Packets

Before we define the concept of best packets, it is important to describe the nature of Doppler shift. Doppler shift is the apparent change in the frequency of electromagnetic signals (such as radio waves) due to the relative movement of the two communicating entities. When there is a small amount of Doppler shift subjected to radio packets, there must be a relatively small motion or velocity between the two bodies. A new metric termed the Doppler Value based on the Doppler shift alone was introduced in [8]. The best packets (and the best paths) are defined with regard to stability in our model. We develop four schemes to formulate best packets:

1) *Packet subjected to the least Doppler shift (based on relative velocity alone)*
2) *Packets subjected to the smallest Doppler Value*
3) *Packet with the largest Link Expiration Time as estimated using the Doppler shift of packet and the power of signal*
4) *Packet with longest LET as calculated using GPS.*

The above schemes are simulated in section IV.

C. Forward Best Request (FOBREQ)

The FOBREQ scheme forwards the best packet, by only forwarding *better* packets as defined by *values* calculated using the schemes in part B. When a packet arrives at a node, the *value* of this packet is stored at this node, and compared with consecutive duplicate/identical packets that arrive at the node. If a consecutive packet has a better value than the stored value, then this packet is forwarded, else it is dropped. If each consecutive packet has a better value than the previous packet, then it is also forwarded. However as will be shown in the simulations of section IV, FOBREQ also effectively manages the *broadcast storm problem* [2] mentioned earlier by reducing rebroadcasts. FOBREQ also prevents the discovery of multiple paths that have the same bottleneck (link) values. Hence here the bottleneck value will be the least stable link (e.g., one with the smallest link duration). As the packet contains the bottleneck value, redundant paths are eliminated. These are paths that have common bottleneck links, in addition to paths that have bottleneck values below the best path.

D. Receive On Most Stable Path

The Receive On Most Stable Path (ROMSP) works on the principle of retrieving data from the most stable path based on any of the metrics described in part B. It is important to emphasize here that there may also be more than one node from which data can be retrieved from in ROMSP.

The algorithm is as follows. The requesting node broadcasts a *route request* (RREQ) to all nodes within range. The receiving node first checks whether the current request packet is *better* (according to Section C) than the previous identical request. If it is, it will then check whether it can provide the requested data, or whether it has knowledge of a path that can provide this requested data. If it does, it will produce a *route reply* (RREP), else it will add its own address to the request packet, add its value as the “best” value so far, and rebroadcast the packet. A new route discovery is always initiated prior to the link being expired. This happens at a time *t* before the estimated link expiration time. In addition to FOBREQ, setting a maximum lifetime for packets minimizes broadcasting. The lifetime of packet ensures that rebroadcasting of packets ceases after either certain number of rebroadcasts by different nodes (hopcount), or when the lifetime of a packet is reached (packet expiration).

We also like to point out that when data transfer is complete, the resources are released; however the path remains in the Path Cache of the node until it expires, and is then purged from the Path Cache. The reason behind this is that if there are further attempts of the same data retrieval or if another request is received wishing to obtain the same data, this path can be reused. Intuitively, nodes which forward the RREQ, will also learn about the new path and store this information in their Path Cache.

E. Packet Format

The route request (RREQ) packet format is as follows:

<CNA><#scheme><CSF><lifetime>

The *Cached Node Addresses* (CNA) field is where the addresses of the forwarding nodes are stored. Before a node forwards the packet, it adds its own address to the CNA. The *scheme* field identifies the cost scheme which is to be used as outlined in part B. The *Cost So Far* (CSF) field is used by FOBREQ in order to determine best packets to forward and ignore the rest. This field is also updated on the reverse (reply) path and is hence used to determine cost of paths in order to determine the best path for routing at the requesting node. The *lifetime* field will determine the expiration parameters for the RREQ packet so that the packet is not indefinitely rebroadcasted over the entire network. When the lifetime of a packet is up, it is dropped.

F. Calculation of Link Expiration Time

Although the use of GPS should become commonplace in mobile nodes, we introduce a scheme to estimate the LET without the need of GPS (in case the GPS is not able to effectively estimate the velocity of nodes or is simply not available). We use the Doppler shift subjected to packets to calculate the relative velocity of nodes. The distance between nodes is calculated using the scheme used in [3], which uses the power of signals to calculate the distance between the nodes by using the simplified free space propagation model given in [17]. For the mobility model it is assumed that mobile nodes are pseudo-linear, and highly mobile in nature. A good example of this kind of system is an aeronautical ad hoc network [8].

The *estimated* initial LET using the Doppler shift of packets and power of signal (of packets) is given by

$$LET \approx \frac{1}{2v}\left(\sqrt{2d^2 - 4\left(d^2 - R^2\right)} + d\sqrt{2}\right)$$

$$(if\ \frac{f}{f_o} < 1) \ \ldots \text{ for approaching nodes}$$

$$LET \approx \frac{1}{2v}\left(\sqrt{2d^2 - 4\left(d^2 - R^2\right)} - d\sqrt{2}\right)$$

$$(if\ \frac{f}{f_o} > 1) \ldots \text{ for receding nodes}$$

where f is the actual frequency of the signal, f_o is the observed frequency, R is the maximum communication range between two mobile nodes and d is the initial distance between two nodes given by

$$d = \frac{\lambda}{4\pi}\sqrt{\frac{p_t}{p_r}}$$

where p_r is the initial received signal power, p_t is the known transmission signal power and λ is the carrier's wavelength. LET_i is used for determining the best path for routing (during the routing decision).

Each node possesses a Path Cache which automatically expires and purges paths, once they expire or become invalid. This occurs when a node receives a RERR packet corresponding to a link in a path or when the *estimated* Link expiry is reached.

Control packets must be propagated using a much lower frequency than the actual data transmission, in order to minimize the effect of atmospheric attenuation and hence be able to effectively estimate the LET. Note that if f/f_0 is one, the LET is infinite and hence nodes will indefinitely remain within each other's range.

Additionally, GPS can be used to determine the distance between nodes. From [18], if we consider two mobile nodes i and j with a transmission or line-of-sight (LOS) range of r, speeds v_i and v_j, coordinates (x_i, y_i) and (x_j, y_j), and velocity angles θ_i and θ_j respectively, the LET is predicted by

$$LET = \frac{-(ab+cd)+\sqrt{(a^2+c^2)r^2-(ad-bc)^2}}{a^2+c^2} \qquad (3)$$

$$a = v_i\cos\theta_i - v_j\cos\theta_j$$
$$b = x_i - x_j$$
$$c = v_i\sin\theta_i - v_j\sin\theta_j$$
$$d = y_i - y_j$$

G. *Link Breakage*

When a primary path breaks, the node that notices this change sends a RERR packet back to the source node. For an example, let us assume link AB breaks in Fig. 2. In this scenario, node B notices that there are no acknowledgement of data packets being received from node A, and assumes the link is broken. Hence it will send a RERR packet specifying the suspected link that is broken. Here it is RERR(B, A). The RERR packet is forwarded to the source node S. First the source node will select the next best path that does not contain the link BA. The Path Cache is then updated by removing (purging) all paths that contain the link BA. If yet another unexpected link breakage occurs, a new path is selected from the (updated) Path Cache. When a link breaks, a local repair procedure takes place, similar to ABR. However as soon as the link is repaired, the node which is responsible for the repair will send a RERR. If there is a sudden broken link one of two things will happen:

1. If there is an alternative path at the node which realizes the link break, the alternative path is chosen, and a RERR packet containing the broken link information is sent back to the source node. The data packets already on their way (having the node caches containing the broken link) are sent via the new link. i.e. packets are *salvaged,* adapted from DSR packet salvaging [12], where the original route cache in packet is replaced by the new alternative route cache, and then forwarded.

2. If there is no alternative path, a local recovery similar to ABR is performed. If the broken link is less than h hops (where h is usually 1 to 3 hops) from the source, an RERR message with the detail of broken link is sent to the source node and the source node initiates a route discovery. Otherwise a local route recovery procedure takes place where the node detecting the broken link will broadcast a 2-hop recovery request similar to that of [19]. Once the node in charge discovers a new route to the destination, it will send a *Route Recovery* RREC(A-B, A-C) showing the broken link and the new link back to the source. The source will then update its Path Cache, purging and updating the paths in the Cache. However unlike [19], since the process resembles source routing, the source needs to know the local repair, so that if the node responsible for the local repair fails, the source node or the nodes on the upstream of the failed node can handle the broken link.

The proposed scheme also reduces RERR packets by selecting/choosing new paths before the path (link) expires. Thus it prevents the path to be broken and RREPs being sent. RREP packets are hence only produced due to unexpected link failures. This effectively reduces the total number of control messages.

At a time t before the primary link's estimated expiry, a new route discovery takes place, and the Path Cache at source is updated. At the time of link's estimate expiry, the new found route is selected. This is done so that the delay between actual link breakage, notification and path re-establishment is avoided. The alternate paths are only there to supplement unexpected link breakage. We note that in most cases the primary path usually has the longest link duration. Hence close to the expiry of this primary path the alternate paths have already been exhausted and most likely purged from the Path Cache. Effectively they are not suitable, and hence a new route discovery must take place.

IV. Simulation Of The Proposed Scheme

In the following simulation we can see the performance of ROMSP with regard to using different stability metrics. In these simulations 5000 nodes are simulated and their speeds varied to investigate the effect of the schemes with regards to control overhead. Nodes move in a linear fashion, in a predefined direction with no pause times. Fig. 3 shows the results of the effect of increasing speed on number of control messages propagated (overhead). From the figure, it can be seen that LET outperforms the other schemes, followed by Power-Doppler (PD) combination estimation of LET, and the Doppler Value (DV), and lastly the relative velocity (VREL) scheme which only looks at the Doppler shift subjected to a received packet. Fig. 4. shows the performance compared to that of DSR using shortest distance as the metric for path selection. In this case mobility parameters are not taken into account for path selection. Other traditional algorithms that do not consider mobility would yield similar results to DSR. In Fig. 4, the large overhead seen in DSR is a result of the excessive link-breakages due to the selection of random unstable paths, and hence the re-broadcasting of control (RREQ) messages for discovery of new paths. As the speed of nodes increase, the number of links and hence path breaks increase, consequently resulting in higher control overhead. The other schemes use mobility information to choose more stable paths, causing much fewer link breakages during the simulation, and hence dramatically decreasing the overhead caused by route discovery.

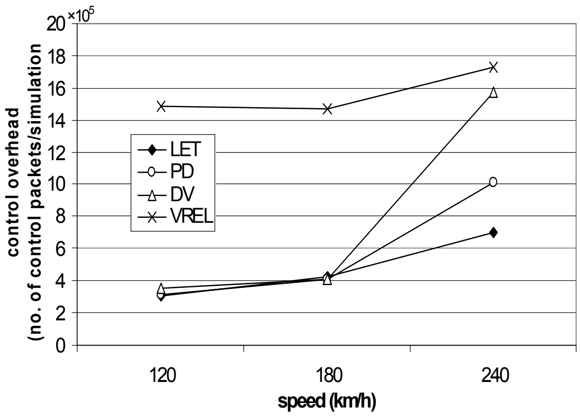

Fig. 3. Control overhead vs. speed.

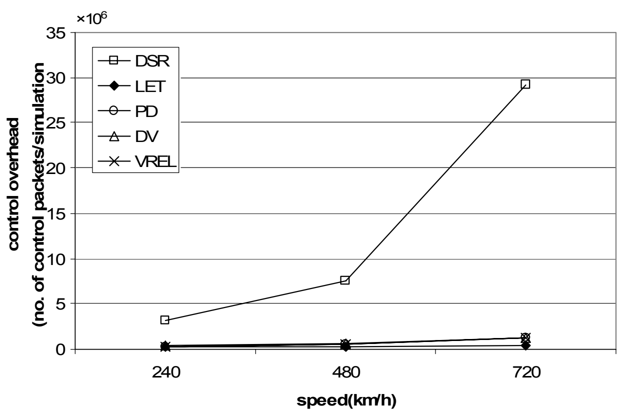

Fig. 4. Control overhead with respect to DSR.

V. CONCLUSIONS AND FUTURE WORKS

In this paper we introduced new schemes to reduce control overhead and increase link duration and stability in highly mobile ad hoc networks. Simulation results show the effectiveness of these schemes in highly mobile scenarios with respect to reducing control overhead. Future work should focus on further optimization of the proposed metrics and comparison with regard to other existing routing protocols used for MANETs. It is also believed that a better estimation of Doppler-Power equation can be derived to further enhance the performance of the routing protocol.

REFERENCES

[1] Y. Ko and N. H. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks," in *Proc. IEEE/ACM MobiCom*, pp. 66-75,. Dallas, Texas, USA, Oct. 1998.

[2] S. Y. Ni, Y. C. Tseng, Y-S. Chen, and J. P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proc. Int. Conf. on Mobile Computering and Networking (MobiCom '99)* pp. 151-162, 1999.

[3] D. Kim, C. K. Toh, J. C. Cano, and P. Manzoni, "A bounding algorithm for the broadcast strom problem in mobile ad hoc networks," in *Proc. IEEE Wireless Communication and Networking Conference, WCNC'03*, vol. 2, pp. 1131-1136, New Orleans, LA, USA, Mar. 2003.

[4] P. McCarthy and D. Grigoras, "Multipath associativity based routing," in *Proc. 2nd IEEE Annual Conference on Wireless On-demand Network Systems and Services (WONS'05)*, 2005.

[5] S. -J. Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in ad hoc networks," in *Proc. IEEE ICC'01,* vol. 10, pp. 3201-3205, Jun. 2001.

[6] A. Nasipuri and S. R. Das, "On-demand multipath routing for mobile ad hoc networks," in *Proc. IEEE ICCCN'99,* pp. 64-70, Oct. 1999.

[7] C.-K. Toh. "Associativity based routing for ad hoc mobile networks," *Wireless Personal Communications Journal, Special Issue on Mobile Networking and Computing Systems*, pp. 103-139, Mar. 1997.

[8] E. Sakhaee, A. Jamalipour, and N. Kato, "Aeronautical ad hoc networks," in *Proc. IEEE WCNC'06*, vol. 1, pp. 246-251, Apr. 2006.

[9] S. Jiang, "An enhanced prediction-based link availability estimation for MANETs," *IEEE Transactions on Communications*, vol. 52, Issue 2, pp. 183-186, Feb. 2004.

[10] M. Gerharz, C. de Waal, M. Frank, and P. Martini, "Link stability in mobile wireless ad hoc networks," in *Proc. 27th Annual IEEE Conference on Local Computer Networks, (LCN'02)* pp. 30-39, Nov. 2002.

[11] A.B. McDonald, and T. Znati, "A path availability model for wireless ad-hoc networks," in *Proc. IEEE WCNC'99*, vol. 1, pp 35-40, Sep. 1999.

[12] D. B. Johnson, D. A. Maltz, and Y. C. Hu, "The dynamic source routing protocol for mobile ad hoc networks (DSR)," http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt.

[13] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, "A distance routing effect algorithm for mobility (DREAM)," in *Proc. IEEE/ACM MobiCom,* pp. 76-84,. Dallas, Texas, USA, Oct. 1998.

[14] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. IEEE/ACM MobiCom*, pp. 243-254, Boston, Massachusetts, USA, Aug. 2000.

[15] X. Li and L. Cuthbert, "On-demand node disjoint multipath routing in wireless ad hoc networks" in *Proc. 29th Annual IEEE International Conference on Local Computer Networks*, pp 419-420, 16-18 Nov. 2004.

[16] G. Aggelou, "Mobile ad hoc networks : from wireless LANs to 4G networks," p 91. McGraw-Hill Professional Engineering, 2005.

[17] T. S. Rappaport, *Wireless Communications: Principles and Practice, Prentice Hall*, 1996.

[18] W. Su, S. J. Lee, and M. Gerla, "Mobility prediction and routing in ad hoc wireless networks, *Int. J. of Network Mgmt*, 2001, vol. 11, pp. 3- 30.

[19] G. Liu et al., "PATCH: a novel local recovery mechanism for mobile ad-hoc networks," in *Proc. IEEE Vehicular Technology Conference (VTC 2003)*, vol. 5, pp. 2995 – 2999, Oct. 2003, Florida, USA.