

Self-Adaptive On Demand Geographic Routing Protocols for Mobile Ad Hoc Networks

Xiaojing Xiang* Zehua Zhou* Xin Wang†

*State University of New York at Buffalo, Buffalo, NY, USA
{xxiang, zzhou5}@cse.buffalo.edu

†State University of New York at Stony Brook, Stony Brook, NY, USA
xwang@ece.sunysb.edu

Abstract—It has been a big challenge to develop routing protocol that can meet different application needs and optimize routing paths according to the topology change in mobile ad hoc networks. Basing their forwarding decisions only on the local topology, geographic routing protocols have drawn a lot of attentions in recent years. However, inaccurate local topology knowledge and the outdated destination position information can lead to inefficient geographic forwarding and even routing failure. Proactive local position distribution can hardly adapt to the traffic demand. It is also difficult to pre-set protocol parameters correctly to fit in different environments. We have developed two self-adaptive on-demand geographic routing schemes. The local topology is updated in a timely manner according to network dynamics and traffic demands. Our route optimization scheme adapts the routing path according to both topology changes and actual data traffic requirements. Each node can determine and adjust the protocol parameter values independently according to different network environments, data traffic conditions and node's own requirements. Our simulation studies have shown that the proposed routing protocols are more robust and outperform the existing geographic routing protocol. Specifically, the packet delivery latency is reduced almost four times as compared to GPSR at high mobility.

I. INTRODUCTION

It is challenging to develop robust routing protocol for dynamic Mobile Ad Hoc Networks (MANET). Geographic routing protocols [1] [2] are generally more scalable and reliable than conventional topology-based routing protocols [3] [4] with their forwarding decisions based on the local topology. Geographic routing assumes mobile nodes are aware of their own positions through certain positioning system (e.g., GPS), and a source can obtain the destination's position through some kind of location service [5]. An intermediate node makes packet forwarding decisions based on its knowledge of the neighbors' positions and the destination's position inserted in the packet header by the source. By default, the packets are greedily forwarded to the neighbor that allows for the greatest geographic progress to the destination. When no such neighbor exists, perimeter forwarding [1] is used to recover from the local void, in which the packets traverse the face of the planarized local topology subgraph by applying the right-hand rule until greedy forwarding can be resumed.

Although better than topology-based routing, the inaccurate knowledge of local geographic topology and destination position can greatly affect geographic routing performance. To obtain the local geographic topology, each mobile node in current geographic routing protocols [1] periodically broadcasts a

beacon containing its position. Such proactive mechanism not only creates a lot of control overhead when there is no traffic, but also results in "outdated" topology knowledge under highly dynamics (Section III). Additionally, relying on only one-hop topology information in current geographic routings may lead to non-optimal forwarding and blind forwarding. Furthermore, it is hard to preset the routing parameters to the correct values for any scenarios, which will impact routing performance.

We propose two self-adaptive on-demand geographic routing protocols. To summarize, our contributions in this work include:

- Analyzing the effect of outdated position information on the performance of geographic routing.
- Proposing two novel on-demand geographic routing protocols with different schemes to obtain and maintain topology information. One protocol purely relies on one-hop topology information as other geographic routing schemes; the other one assumes a hybrid scheme which combines geographic and topology-based mechanisms for more efficient routing, while avoiding the performance degradation of conventional geographic routing due to the constraints in *local* view of topology.
- Introducing route optimization schemes. To our best knowledge, this is the first geographic routing scheme that adapts the path to the change of network topology and traffic demand.
- Designing an efficient position distribution mechanism that can adapt its behavior under different dynamics and according to the routing requirements to reduce the control overhead and provide more accurate and updated position information for efficient routing.
- Adapting parameter settings in both protocols according to network environments, data traffics and mobile nodes' own requirements.

The rest of this paper is organized as follows. In Section II, we discuss some related work. Section III makes an analysis on the effect of outdated topology knowledge on geographic routing. A detailed description of the two protocols is given in Section IV. Section V shows the simulation results. Finally, Section VI concludes the paper.

II. RELATED WORK

The conventional on-demand routing protocols (e.g., [3] [4]) often involve flooding in route discovery, which limits the

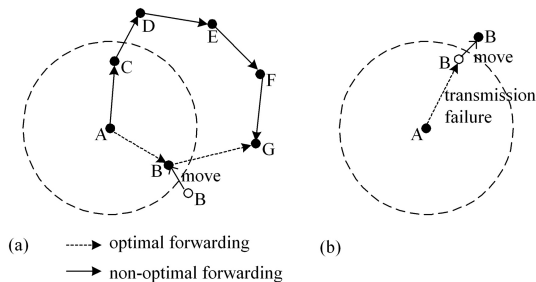


Fig. 1. Negative effects of outdated topology information on geographic routing: (a) non-optimal routing; (b) forwarding failure.

scalability. To reduce overhead, LAR [2] reduces the flooding range by making use of the nodes' position information.

Unlike topology-based routing protocols, geographic routing is based on mobile nodes' positions. Existing geographic routing protocols have many limitations as discussed in Section I. Authors in [6] attempted to remove the proactive beacons in geographic routing protocols. However, the simple contention-based scheme adopted may lead to redundant packet forwarding and higher collision probability, and hence it cannot work properly when the traffic load is high. Son et al. [7] conducts a simulation-based study on the negative effect of mobility-induced location error on routing performance. Instead, we propose two on-demand adaptive geographic routing protocols that can meet different application and traffic needs and adapt to different conditions.

III. ANALYSIS ON THE IMPACT OF POSITION INACCURACY ON GEOGRAPHIC ROUTING

Fixed-interval beaconing commonly adopted in current geographic routing protocols may result in outdated local topology knowledge at the forwarding node, which leads to *non-optimal routing* and *forwarding failure*. 1) *Non-optimal routing*. Fig. 1 (a) shows an example of non-optimal routing due to the outdated local topology knowledge. Node B just moved into A's transmission range, which is unknown to A. Without knowing any neighbor closer to the destination G, A will forward the packet to node C then D by using perimeter forwarding. The resulted path has five hops, while the optimal path between A and G should have only two hops after B bridges the void between A and G. 2) *Forwarding failure*. In literature work [1], a node will keep a neighbor's information until timeout even when the neighbor has moved out of its transmission range and the timeout interval is often set as multiple beaconing intervals. Forwarding failure will happen when the node forwards packets to such a "false" neighbor (e.g., Fig. 1 (b)) and result in packet dropping or rerouting [1]. More seriously, before detecting the unreachability, the continuous retransmissions at MAC layer will reduce the link throughput and fairness, and increase the collisions. This will further increase delay and energy consumption.

IV. SELF-ADAPTIVE ON DEMAND GEOGRAPHIC ROUTING PROTOCOLS

In this section, we first propose two Self-adaptive On-demand Geographic Routing (SOGR) protocols, and then

introduce our route optimization schemes. In both protocols, we assume every mobile node is aware of its own position, a source can obtain the destination's position through some kind of location service, and promiscuous mode is enabled on mobile nodes' network interfaces. In the following presentation, except when explicitly indicated, F represents the current forwarding node, D is the destination, N denotes one of F's neighbors, pos_A is the position coordinates of A and $dis_{(A,B)}$ is the distance between node A and B.

A. Scheme 1: SOGR with Hybrid Reactive Mechanism (SOGR-HR)

In SOGR-HR, we use a geographic and topology-based combined mechanism to reactively search for the next-hop. By incorporating topology-based path searching, information of a larger range topology can be obtained when necessary to build more efficient routing path.

1) *Geographic-based greedy forwarding*: Normally F will attempt to forward a packet greedily to a neighbor closest to D and closer to D than itself. With no next hop to D cached, F buffers the packet first and broadcasts a request message $REQ(D, pos_D, pos_F, hops)$ with $hops = 1$ to restrict the searching range to one-hop neighbors. A neighbor node N closer to D than F will send back a REPLY. F will record N as the next hop to D with transmission mode as *greedy* and unicast the data packet to N. To avoid collisions, N will wait for a backoff period before sending the REPLY and the pending REPLY will be cancelled if it overhears a REPLY from another neighbor closer to D than itself. To make sure the neighbor closer to D responds sooner and suppresses others' REPLYs, the backoff period should be proportional to $dis_{(N,D)}$ and bounded by the max value $hops \times Intval_{backoff}$, where $Intval_{backoff}$ is a protocol parameter, and $hops = 1$ in greedy forwarding. The backoff period is calculated as:

$$backoff = hops \times Intval_{backoff} \times \left(1 - \frac{dis_{(F,D)} - dis_{(N,D)}}{hops \times R}\right), \quad (1)$$

where R is the transmission range of mobile nodes.

2) *Topology-based recovery forwarding*: F may not have neighbors closer to D, resulting in a local "void". We use a recovery strategy with expanded ring search, which is normally used in path searching in topology-based routing [4] [3].

After sending a REQ with $hops = 1$, if there is no REPLY after $1.5 \times hops \times Intval_{backoff}$, F will increase its searching range to two hops and broadcast a REQ with $hops = 2$. When a second-hop neighbor of F gets this REQ and is closer to D, it sends a REPLY following the reverse path of the REQ message, with backoff period calculated from Eq. 1 with $hops = 2$. The intermediate nodes will record the last hop of the REPLY as the next hop towards D with transmission mode as *recovery*. To avoid overhead, an intermediate node will drop a REPLY if it already forwarded or overheard a REPLY from a node closer to D than the current replier. F will unicast the data packet to the detected next hop. If the route searching fails with $hops = 2$, F may expand the searching range again by increasing $hops$ until $hops$ reaches Max_{hops} .

3) *Adaptive parameter settings*: SOGR-HR has two parameters, $Intval_{backoff}$ and $Intval_{timeout}$. Both are set adaptively. A larger backoff interval will lead to longer next-hop searching delay, while too small an interval will give the first replier less time to suppress others' REPLYs, leading to larger control overhead. $Intval_{backoff}$ is determined by F and is inserted in its REQ message. $Intval_{backoff}$ for D is initialized as $Ref_{backoff}$. During each one-hop route searching process for D, if the number of REPLYs F received is larger than 1, $Intval_{backoff}$ is increased by $Inc_{backoff}$; otherwise, it is decreased by $Inc_{backoff}$. The updated $Intval_{backoff}$ will be used in the next route searching process for D. Both REQ and REPLY messages carry the senders' pos , and data packet contains forwarder's pos to refresh their pos maintained by the receivers. To control how long a node will cache the pos , each pos is sent with an $Intval_{timeout}$ value. A node estimates $Intval_{timeout}$ according to its velocity v as: $Intval_{timeout} = Dist_{timeout}/v$, where $Dist_{timeout}$ is a distance threshold. $Intval_{timeout}$ is bounded by the range $[Min_{timeout}, Max_{timeout}]$ to avoid too frequent position information invalidation and too long timeout interval for slowly moving or static nodes.

B. Scheme 2: SOGR with Geographic Reactive Mechanism (SOGR-GR)

SOGR-GR depends only on one-hop neighbors' positions to make greedy and perimeter forwarding like other geographic routing protocols [1]. However, it adopts a reactive beaconing mechanism. The periodic beaconing is triggered only when a node overhears data traffic from its neighbors the first time, and the beaconing will be stopped if no traffic is heard for a pre-defined period. A forwarding node may broadcast a request (REQ) message to trigger its neighbors' beaconing when necessary.

1) *Adaptive position distribution*: Every node keeps three time values t_{req} , $t_{reqHeard}$ and t_{beacon} , in which t_{req} records the time when the latest REQ or data packet was sent out, $t_{reqHeard}$ is the time when the latest REQ or data transmission was heard, and t_{beacon} saves the last beaconing time. REQ message and data packet also serve as beacons since they contain forwarder's position.

Whenever a node receives a REQ or overhears data transmission from its neighbor, it will broadcast a BEACON if $t_{cur} - t_{beacon} \geq Intval_{beacon}$, where t_{cur} is the current time. This is to ensure that periodic beaconing is only triggered by the first heard REQ or data packet after a silent period. The beaconing interval $Intval_{beacon}$ is bounded within $[Min_{beacon}, Max_{beacon}]$ as to be described in the following subsection. To avoid synchronous beaconing from multiple neighbors, the BEACON sending time is jittered by a random delay less than $Intval_{jitter}$. After beaconing at time t , at the next beaconing time $t + Intval_{beacon}$, the node sends a beacon only when $t_{cur} - t_{reqHeard} < Intval_{beacon}$; otherwise, it keeps silence, so that beaconing will be stopped when there is no traffic for a period.

Before forwarding a packet, if $t_{cur} - t_{req} \geq Min_{beacon}$, F sends out a REQ to trigger its neighbors' beaconing, and

delays its forwarding decision for a period $3 \times Intval_{jitter}$ to collect the neighbors' positions; otherwise, F will make the forwarding decision directly based on the existing local topology information.

2) *Adaptive parameter settings*: The main parameters in SOGR-GR are $Intval_{beacon}$ and $Intval_{timeout}$. A node can decide its $Intval_{beacon}$ according to different rules, for example, its remaining energy or moving speed. Like $Intval_{timeout}$ setting in SOGR-HR, in our simulation, the $Intval_{beacon}$ is determined according to the node's velocity and is limited within $[Min_{beacon}, Max_{beacon}]$ to avoid too frequent beaconing or too long beaconing interval from certain "lazy" nodes. $Intval_{timeout}$ is the caching time of position information and is set as $2 \times Intval_{beacon}$.

C. Route Adaptation and Optimization

With the movements of nodes, the cached topology information will get outdated and the routing path may become inefficient. Motivated by the analysis in Section III, in our optimization scheme, the validity of the cached topology information is evaluated before packet forwarding to avoid *forwarding failure* due to outdated neighbor information, and the routing path is optimized with the cooperation of the forwarding node and its neighbors to avoid *non-optimal routing* due to the inaccuracy in topology knowledge. The optimization mechanisms are applicable to both protocols. For the convenience of presentation, we will describe these mechanisms mainly based on SOGR-HR.

1) *Validity estimation of next hop*: In SOGR-HR, after a route searching phase (Section IV-A), the current best next hop, say C, is cached to reduce the delay and control overhead for route searching. Node C may move out of the transmission range of F or may be no longer the best next hop. Before forwarding a packet, F estimates the current position of C (x, y) as:

$$\begin{cases} x = x_{new} + (x_{new} - x_{old})(t_{cur} - t_{new})/(t_{new} - t_{old}), \\ y = y_{new} + (y_{new} - y_{old})(t_{cur} - t_{new})/(t_{new} - t_{old}). \end{cases} \quad (2)$$

where (x_{new}, y_{new}) and (x_{old}, y_{old}) are C's newest two positions recorded by F with t_{new} and t_{old} as their obtained time, and t_{cur} is the current time. If C's estimated position is out of F's transmission range, or is no longer closer to D when transmission mode is *greedy*, a route searching process will be triggered to find a valid next hop. Similarly, in SOGR-GR, before F forwards a data packet, the neighbors' positions are estimated. Invalid neighbors' information is removed.

2) *Optimization for the forwarding path*: In SOGR-HR, due to the local topology change, the cached next hop C may no longer be the best one towards D. To achieve more optimal routing, F's neighbors monitor whether F makes correct forwarding decisions.

After F forwards a packet to C which continues forwarding towards D, a neighbor N overhears both forwarding and gets pos_F , pos_C and pos_D . A packet forwarded using recovery mode will also carry the position of the node (say node S) where the recovery forwarding started, pos_S . If N determines that it is a more optimal next hop than C, it sends to F a

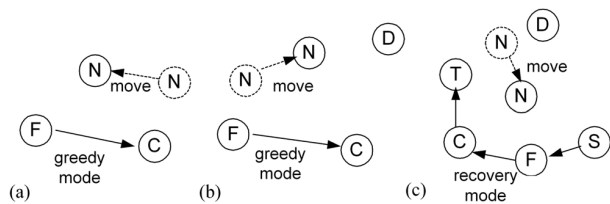


Fig. 2. Route optimization: (a) case 1; (b) case 2; (c) case 3.

message $CORRECT(pos_N, D)$ asking it to change its next hop to N. We consider three route optimization cases, using the examples in Fig. 2. The criterion for N to send a CORRECT in each case is as follows with $mode_{(A,B,D)}$ representing the forwarding mode from A to B towards destination D:

- 1) (Fig. 2 (a)) N is the destination of the packet, which means F should forward the packet directly to N.
- 2) (Fig. 2 (b)) $mode_{(F,C,D)} = greedy$, and $dis_{(N,D)} < dis_{(C,D)}$.
- 3) (Fig. 2 (c)) $mode_{(F,C,D)} = recovery$. There are two cases. Case 1: F is the last hop of the recovery mode, so $dis_{(C,D)} < dis_{(S,D)}$. If $dis_{(N,D)} < dis_{(C,D)}$, F should forward packet to N. Case 2: F is not the last hop of the recovery forwarding, so $dis_{(S,D)} \leq dis_{(C,D)}$. If $dis_{(N,D)} < dis_{(S,D)}$, F should forward the packet to N and N can resume the greedy forwarding. Overall, if $dis_{(N,D)} < dis_{(S,D)}$ and $dis_{(N,D)} < dis_{(C,D)}$, N needs to send a CORRECT to F.

Through this process, more optimal routing can be achieved. In case 2 and 3, to avoid that multiple neighbors detect non-optimal forwarding and send CORRECT simultaneously, the CORRECT message will also be pended and suppressed as that for REPLY message with $hops = 1$. There is also another possibility for the recovery forwarding. Suppose recovery forwarding starts at F, F sets its next hop to C in order to reach node T which is closer to D than F. Since F is not aware of the positions of non-neighboring nodes on the recovery path to T, a node on the recovery path should notify F with an ERROR message whenever it detects that its next hop is unreachable. T should also notify F if it is no longer closer to D than F. F will start a new route searching process.

SOG-GR assumes similar route optimization schemes and has three similar possible cases as described above. When N detects a non-optimal forwarding from F, it indicates that F may have an outdated pos_N , so N will broadcast a BEACON message. The BEACON will also be backoffed and suppressed as the CORRECT message described above.

3) *Handling inaccurate destination position*: Suppose the position of the destination node D obtained from location service is pos_{inac} , and D is currently located at pos_D . SOG-HR and SOG-GR are robust to the destination position inaccuracy by nature. Suppose F is the last hop towards pos_{inac} , in SOG-HR, as long as D is located in the route searching range of F, F will build the path to D. In SOG-GR, F will forward the packet directly to D if D is its neighbor. Furthermore, in the case 1 of the optimization process (Section IV-C.2), D will notify its neighbors whenever it detects that a neighbor didn't forward the packet directly to it. These can handle the

case that D is within the transmission range of any node on the forwarding path towards pos_{inac} . If the inaccuracy is too large to reach D by using the above methods, the last hop can start a limited-range searching for D, but this is generally unnecessary since the inaccuracy of location service is normally smaller than transmission range [5].

V. PERFORMANCE EVALUATION

A. Simulation Overview

We implemented SOG-HR and SOG-GR within the Global Mobile Simulation (GloMoSim) [8] library. For performance reference, we also run simulations on the classic topology-based on-demand routing protocol AODV [3], LAR [2], an on-demand routing protocol utilizing position information to restrict the flooding range of route searching, and geographic routing protocol GPSR [1]. In SOG-HR's adaptive parameter settings, $Ref_{backoff}$, $Inc_{backoff}$, $Dis_{timeout}$ and $[Min_{timeout}, Max_{timeout}]$ were set as 10ms, 2ms, 300m and [10s,30s] respectively with Max_{hops} as 2. In SOG-GR, we set Dis_{beacon} and $[Min_{beacon}, Max_{beacon}]$ as 150m and [5s,15s] with $Intval_{jitter}$ as 10ms.

The simulations were run with 300 nodes randomly distributed in the area of $3000m \times 1500m$. The movement of nodes follows the *random waypoint* mobility model [9]. The moving pause time was set as 0 second and minimum speed was 0 m/s. IEEE 802.11b was used as the MAC layer protocol and the nominal transmission range was 250m. Each simulation lasted 900 simulation seconds. A traffic flow was sent at 8 Kbps using CBR between randomly chosen source and destination pair with packet length 512 bytes. There are 30 CBR flows. A simulation result was gained by averaging over six runs with different seeds.

We study the following metrics:

- 1) *Packet delivery ratio*: The ratio of the packets delivered to those originated by CBR servers.
- 2) *Control overhead*: The total number of control message sending accumulated over each hop divided by the total number of data packets received.
- 3) *Average number of data packet forwarding per delivered packet*: The total number of data packet forwarding accumulated from each hop (including rerouting) over the total number of data packets received. Both the non-optimal routing and rerouting due to unreachable next hop will increase the forwarding overhead.
- 4) *Average end to end delay*: The average time interval for the data packets to traverse from the CBR sources to the destinations.

B. Simulation Results

We study the performance of various protocols by varying maximum moving speed from 0m/s to 50m/s. The scalability of LAR and AODV is limited by the involved network-range or restricted range flooding. The end-to-end paths obtained during route discovery phases are easily broken under network dynamics resulting in packet droppings. As a contrast, the geographic routing protocols determine the next hop based

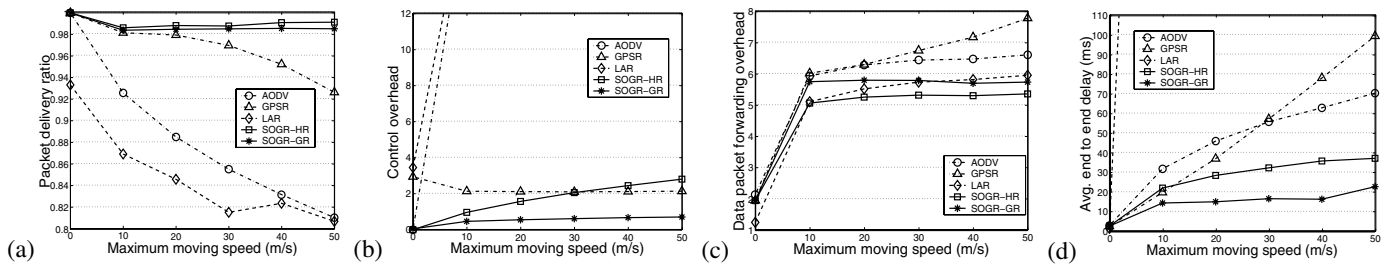


Fig. 3. Performance with different maximum moving speeds (300 nodes, 3000m × 1500m, 30CBR): (a) packet delivery ratio; (b) control overhead; (c) average number of data packet forwarding; (d) average end to end delay.

only on the knowledge of local topology, and are hence more scalable and robust. In Fig. 3 (a), all the three geographic routing protocols have much higher delivery ratio, while the delivery ratio of GPSR drops quickly when the moving speed is higher than 20m/s. The stable performance of SOGR-HR and SOGR-GR shows their adaptability and robustness to network dynamics. When mobile nodes move faster, the local topology information is easier to be invalid. The adaptive parameter settings and more flexible position distributions in SOGR-HR and SOGR-GR will intelligently generate necessary control messages to better track mobility.

This is verified by Fig. 3 (b), where both SOGR-HR and SOGR-GR generate more control messages as mobility increases; while GPSR maintains fixed beaconing interval, which results in unnecessary control overhead when the mobility is low, and cannot catch the topology change when the mobility is high. SOGR-HR is seen to generate slightly higher control overhead than SOGR-GR. In SOGR-HR, whenever the next hop is invalid, the forwarding node will start a new route searching phase; while in SOGR-GR, the forwarding node just needs to pick another valid next hop from its neighbor table without incurring extra control overhead.

As expected, GPSR needs more packet forwarding to deliver a packet as shown in Fig. 3 (c), due to the non-optimal routing and rerouting caused by the outdated local topology knowledge. As AODV and LAR usually search for the shortest path to the destination, they have fewer forwarding. Both SOGR-HR and SOGR-GR have much fewer forwarding under high dynamics as compared to GPSR. These are due to their more efficient position distribution mechanism and route optimization process to adapt the route more quickly to the topology change. SOGR-GR has a little more forwarding than SOGR-HR because its perimeter forwarding may introduce more packet forwarding, while SOGR-HR can build more efficient routing path without being constrained to one-hop information.

In Fig. 3 (d), LAR and AODV are seen to have a longer end-to-end delay due to the time required to build the whole path before packet forwarding in traditional on-demand routing protocols. GPSR's end-to-end delay increases faster with increased mobility due to its more rerouting and non-optimal routing paths. It has more than four times delay as compared to SOGR-GR in high mobility case. SOGR-HR has slightly longer delay than SOGR-GR as SOGR-HR will start a new next-hop search whenever the next hop is invalid.

VI. CONCLUSIONS

In this work, we propose two self-adaptive on-demand geographic routing protocols. The two protocols adopt different schemes to obtain and maintain local topology information on data traffic demand. One protocol purely relies on one-hop topology information for forwarding as other geographic routing schemes; the other one combines both geographic and topology-based mechanisms for more efficient path building. With parameter adaptation schemes, each node can determine and adjust the protocol parameter values independently according to different network environments, data traffic conditions and mobile nodes' own requirements. To alleviate the negative effects of outdated local topology information on geographic routing, we design more efficient position distribution mechanisms to update the local topology knowledge in time and adaptively based on demand. We also develop a set of route optimization schemes in which a forwarding node and its neighbors can collaborate to adapt the path to both topology change and traffic demand. The simulation results show that our protocols can efficiently adapt to different scenarios and perform better than the existing geographic routing protocols. Nearly four times delay reduction has been observed in high mobility case.

REFERENCES

- [1] B. Karp and H. T. Kung, "Greedy perimeter stateless routing for wireless networks," in *ACM/IEEE MOBICOM*, August 2000, pp. 243–254.
- [2] Y. Ko and N. Vaidya, "Location-aided routing in mobile ad hoc networks," in *ACM/IEEE MOBICOM*, August 1998.
- [3] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop on Mobile Comp. Sys. and Apps.*, February 1999, pp. 99–100.
- [4] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad-hoc wireless networks," *Mobile Computing*, T. Imielinski and H. Korth, Eds., Kluwer, pp. 153–81, 1996.
- [5] J. Li and et al, "A scalable location service for geographic ad hoc routing," in *ACM/IEEE MOBICOM*, 2000, pp. 120–130.
- [6] M. Heissenbittel and T. Braun, "BLR: beacon-less routing algorithm for mobile ad-hoc networks," *Elsevier's Comp. Commun. Journal*, vol. 27, no. 11, pp. 1076–1086, July 2004.
- [7] D. Son, A. Helmy, and B. Krishnamachari, "The effect of mobility-induced location errors on geographic routing in mobile ad hoc and sensor networks: analysis and improvement using mobility prediction," *IEEE Trans. Mobile Computing*, vol. 3, no. 3, Jul.-Sep 2004.
- [8] UCLA Parallel Computing Laboratory, "Glomosim," <http://pcl.cs.ucla.edu/projects/glomosim/>.
- [9] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. G. Jetcheva, "A performance comparison of multihop wireless ad hoc network routing protocols," in *ACM/IEEE MOBICOM*, 1998, pp. 85–97.