Spatial Aware Geographic Forwarding for Mobile Ad Hoc Networks

Abstract

Stateless greedy forwarding based on physical positions of nodes is considered to be more scalable than conventional topology-based routing. However, the stateless nature of geographic forwarding also prevents it from predicting holes in node distribution. Thus, frequent topology holes can significantly degrade the performance of geographic forwarding. So far the approaches mostly depend on excessive state maintenance at nodes to avoid forwarding failures at topology holes. In this paper, we propose and analyse *spatial aware geographic forwarding* (SAGF), a new approach that proactively avoids constant topology holes caused by spatial constraints while still preserving the advantage of stateless forwarding. *Geographic source routes* (GSR) based on intermediate *locations* are selected to bypass topology holes. Proactive route selection based on the spatial knowledge is a general approach, and thus can be used with any geographic forwarding algorithms. We evaluate our approach by extending greedy forwarding with spatial knowledge. Simulation results comparing with GPSR show that even simple spatial information can effectively improve the performance of geographic forwarding.

1. Introduction

A number of routing protocols have been developed for mobile ad hoc networks (MANETs). Generally, they can be classified into two categories: topology-based routing and position-based routing [4]. Topology-based routing uses the up-to-date topology information of the network to perform packet forwarding, while position-based approach uses the nodes' physical location information.

Topology-based routing can be further divided into three classes by considering how the topology information is exchanged. In the *proactive* mode, routing tables are exchanged periodically between all the nodes, e.g. in DSDV [9]. Routing protocols in the *reactive* mode discover and maintain the routing information only for active communications, e.g. DSR [10]. In the *hybrid* mode, both proactive and reactive routing information exchange modes are used depending on the regions of networks, e.g. ZRP [11]. Simulations show that in MANETs reactive protocols are generally more scalable than proactive protocols and produce less routing overhead [1,7,15]. Nevertheless, due to the dependency on up-to-date topology information, all topology-based protocols are sensitive to the mobility of networks. In networks consisting of highly mobile nodes, such as cars driving on the highway, the network topology changes so rapidly that the excessive topology updates can result in significant routing performance degradation.

In contrast, position-based routing performs packet forwarding based on the nodes' physical positions, thus the overhead of frequent topology updates can be eliminated. However, two additional requirements must be fulfilled here: *positioning* ability and the existence of a *location service*. Positioning ability means that all nodes know their up-to-date locations. Different outdoor and indoor positioning technologies are already proposed or are in daily use, such as GPS and Cricket [14]. Although mobile nodes can disseminate their positions using simple flooding algorithms, an efficient and scalable location service is important for scalability. A location service helps a source node to detect the location of the destination node. Some distributed location services have been proposed, e.g. Grid's Location Service (GLS) [5].

Two different approaches are commonly used in position-based routing: *restricted flooding* and *geographic forwarding*. The basic idea of restricted flooding is to set up a region using the position information of the source node and the destination node, and then flood the packets through this region to the destination. Examples for this approach are *Location Aided Routing* (LAR) [12] and *Distance Routing Effect Algorithm for Mobility* (DREAM) [6]. Although restricted flooding provides high reliability and is simple to implement, it consumes much bandwidth and can result in serious congestions in the network. Thus this approach is often used for route discovery, rather than for data transmissions. In contrast, geographic forwarding only forwards packets to one neighbor each time based on its local state, e.g. *Greedy Perimeter Stateless Routing* (GPSR) and *Intermediate Node Forwarding* (INF) [2,8]. Thus, it produces less routing traffic than the restricted flooding approach. Since geographic forwarding is based on purely local routing decisions, it does not rely on the knowledge of the global topology and thus is nearly stateless.

Simulations show that geographic forwarding generates routing traffic independent of the length of the routes through the network, and therefore keep the routing traffic at a constant low volume as mobility increases [2].

Unfortunately, the stateless nature of geographic forwarding is also its biggest constraint. While the stateless strategy helps geographic forwarding to reduce routing overhead caused by topology updates, its lack of global topology knowledge prevents a mobile node from predicting topology holes as well as forwarding failures. Although there are some methods proposed to route around the holes [2,8], they are used only *after* the geographic forwarding fails, incurring extra cost in detecting forwarding failure and searching for new routes. Moreover, geographic forwarding protocols often assume a uniform distribution of nodes, thus the topology holes only appear occasionally. Nevertheless, this assumption can often be violated in the real world. Spatial constraints and obstacles such as road infrastructures and buildings make the non-uniform distribution of nodes more likely to be the rule than exception.

If a node can predict such topology holes, it can optimize its forwarding decision accordingly to avoid routing to fail. We observe that the topology holes caused by natural or man-made spatial constraints, e.g. lakes or road intersections, are quite predictable with the external knowledge of spatial environments. Thus we investigate the utilization of spatial knowledge, such as digital maps used in navigation systems, to *proactively* avoid routing failures caused by such constant topology holes. For example, cars driving on the road can be simply modeled as nodes moving on the edges of a graph. Using digital maps, a graph can be constructed to model the major topology of the road. This graph model can help the source node to indicate topology holes caused by road structures and to select geographic routes to bypass such holes. Moreover, external spatial knowledge can also help to speed up the recovery process in case of a forwarding failure.

We studied the suggested method by extending normal greedy forwarding with spatial knowledge; the proposed mechanism is generic and can be used to enhance any geographic forwarding approach, like Greedy Perimeter Stateless Routing (GPSR), Intermediate Node Forwarding (INF) and Terminodes Routing [2,3,8].

The remainder of this paper is organized as follows. Section 2 gives a more detailed description of related work. Section 3 introduces the impacts of the spatial constraints on geographic forwarding protocols. Section 4 describes the assumptions of spatial aware geographic forwarding (ASGF). The algorithms are introduced in Section 5. Section 6 presents the simulation environment we used to evaluate the proposed protocol and discusses the simulation results. Finally, section 7 concludes our paper.

2. Related Work

Karp and Kung [2] propose the *Greedy Perimeter Stateless Routing* (GPSR) protocol. GPSR assumes that all nodes know their current locations and each source node can detect the approximate location of the destination using an appropriate location service.

GPSR consists of two methods of packet forwarding: *greedy forwarding* and *perimeter forwarding*. All packets are marked with locations of destinations by the source node. In greedy mode, each forwarding node checks its immediate neighbors' positions, and chooses the neighbor that is geographically closest to the packet's destination as the next hop. This greedy forwarding process will repeat until the destination is reached, or a greedy forwarding failure occurs because of encountering a "local maximum" state. In a local maximum state, the forwarding node can not find any neighbor that is geographically closer to the destination than itself, and the only route to the destination requires the packet to be forwarded to a neighbor node that is temporarily farther in geometric distance from the destination. To recover from this greedy failure, the forwarding node switches the packet into perimeter mode, and forwards the packet using a simple planar graph traversal. Using perimeter forwarding, packets can be routed around the topology holes. As soon as the packet reaches a node that is closer to the destination than the node on which greedy forwarding fails, the packet will be switched back to greedy mode and forwarded further greedily.

Benefiting from its small per-node routing state, GPSR achieves a higher scalability than topology-based routing protocols that rely on end-to-end state concerning the whole forwarding path. However, the advantage in scalability has its price: GPSR will greedily forward a packet for potentially many hops, before a greedy forwarding failure is recognized or the packet is considered to be undeliverable. Thus, the stateless strategy can only make locally optimal forwarding decisions rather than global optimal. Moreover, the perimeter forwarding requires strictly identical radio range of nodes to construct a connected planar graph. This requirement is not always fulfilled in the reality due to obstructions and interferences.

Couto and Morris [8] propose the *Intermediate Node Forwarding* (INF) protocol, a probabilistic solution for routing around geographic topology holes via intermediate geographic locations. In contrast to perimeter forwarding in GPSR, it does not assume identical radio ranges. The basic concept of INF is: When greedy forwarding fails, the forwarding node picks a *random* intermediate position through which to forward the packet. The intermediate position serves as a weak source route, and the packets are routed from source node to this intermediate position using geographic forwarding, and from intermediate position to the destination node using geographic forwarding again. The intermediate location for a destination is chosen from within a circle centered around the location halfway between the originator of the packet and the destination. The circle is initiated with the radius of one quarter of the distance. If routing still fails, the radius is doubled, and a new intermediate location will be chosen within the larger circle. Multiple intermediate locations can also be used if one intermediate location is not sufficient.

Since INF does not rely on the assumption about identical radio ranges, it can be used in more realistic situations where this assumption does not hold true and perimeter forwarding is likely to fail. However, this approach has other constraints: INF will not be started until the originator of the packet successfully receives the negative acknowledgement packet (NAK) for the packet. Thus INF can not avoid the geographic forwarding failures *proactively*. Since the start of INF depends on explicit NAK messages, it must deal with the routing failure of NAK messages, which increases the complexity of the protocol. Moreover, intermediate locations are chosen *randomly*. This probabilistic strategy is very simple, but also unreliable.

Blazevic *et al.* [3] describe routing in a wide area mobile ad hoc network called Terminode Network. *Terminode routing* combines a hierarchical approach with position-based routing and consists of Terminode Local Routing (TLR) and Terminode Remote Routing (TRR). TLR is a short-distance routing method and is based on a proactive distance vector scheme, whereas TRR is a greedy geographic forwarding approach for long-distance routing. TRR is again composed of two elements: Anchored Geodesic Packet Forwarding (AGPF) and Friend Assisted Path Discovery (FAPD). The principle of AGPF is similar to INF: data packets are sent along an anchored path, which defines a rough route from the source to the destination and is given with a list of *anchors* (geographic coordinates). Between anchors, greedy forwarding method is employed. However, while INF chooses the intermediate positions randomly, TRR uses FAPD, a discovery process to query a set of so-called *friends* nodes to find an anchored path to the destination.

While INF is only used as a recovery method after a greedy forwarding failure is detected, Terminode routing uses AGPF to *proactively* bypass topology holes. Nevertheless, this proactivity of Terminode routing requires each node to maintain the state information of a set of *friends* that are out of its radio range. Since path maintenance is required on each node, it is not a stateless protocol. Although hierarchy is introduced to improve the scalability, it also increases the complexity of protocol implementation. Moreover, if AGPF fails, Terminode routing uses perimeter forwarding for recovery, which still depends on the identical radio range.

3. Impact of Spatial Constraints

So far, geographic forwarding protocols only utilize position information on the geometric level, i.e. the position of a node is simply considered as a geographic coordinate. Based on these coordinates we can calculate the Euclidian distance between a node and the packet's destination. The following topological assumption is commonly used in geographic forwarding: the nodes that are physically close are likely to be close in the network topology [5]. Based on this assumption, geographic forwarding often takes physical distance as the basis for forwarding decisions: data packets are forwarded to the neighbor node with the shortest physical distance to the destination node as long as such a neighbor node can be found in the radio range of the current node.

However, in the real world, positions have more meaning than just coordinates taking their spatial environments into account. Therefore, the correctness of forwarding decisions based *only* on physical distances is questionable in situations with holes in node distribution, since the topological assumption described above is likely to be violated.

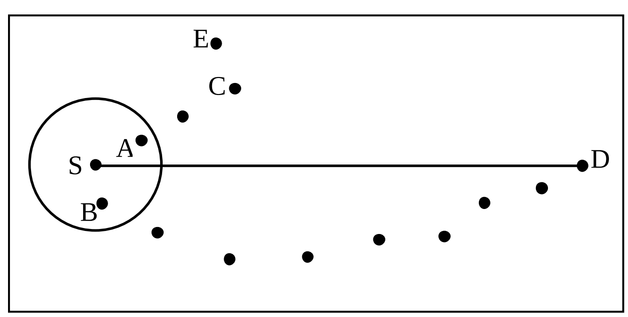

Figure 1. Geometric view of network

Figure 1 shows a snapshot of an ad hoc network consisting of cars driving on the roads. It is obvious that the cars are not uniformly distributed in the whole plane. The circle centered around the node S indicates its radio range. Node S wants to forward a packet to the destination node D, while two nodes A and B are currently located in its radio range. As a basic prerequisite of geographic forwarding, we assume each node knows its current position, the position of its immediate neighbors and the approximate location of the packet's destination. For simplicity we also assume all nodes have identical radio range and a connected path exists between source and destination.

According to the geographic forwarding strategy used in GPSR and INF, S will forward the data packet to A since it has the shortest Euclidian distance to the destination.

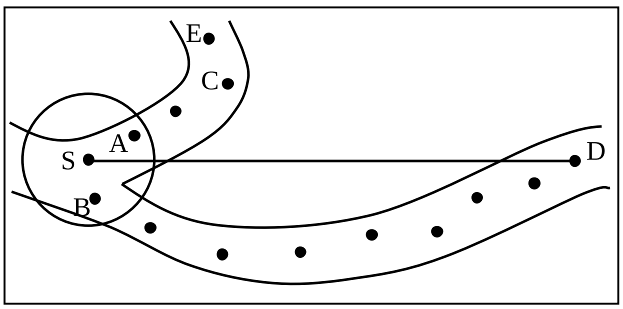

Figure 2. Topological view of network

However, as Figure 2 reveals, this decision is far from being optimal, in fact it is wrong. Considering the underlying road structure shown in Figure 2, we can understand the cause of the non-uniform nodes' distribution. As we see, all nodes only are located along roads; a big topology hole thus occurs at the fork of the road. Although node A is physically closer to the destination than node B, it is on the branch that goes further away from the destination instead of approaching it. So actually node B is the right choice for packet forwarding at node S. However, node S is not aware of this fact, since the underlying spatial environment is not taken into account. Thus positions are still considered at the geometric level as shown in Figure 1.

Using the greedy forwarding strategy, S forwards the packet to A, which will leads to a greedy failure at C, as the positions of nodes in Figure 1 indicate. Perimeter forwarding or INF will then be started for recovery:

- In perimeter forwarding, C will forward the packet to E, trying to route round the topology hole.
- In INF, S will receive a NAK message from C and select a location between S and D randomly as intermediate destination. It is possible that the selected location is located above the line S-D.

Nevertheless, both perimeter forwarding and INF can fail if there is no connected link existing above the line S-D.

The aforementioned example illustrates the impact of spatial environments on both geographic forwarding and recovery methods: while geographic forwarding fails at constant topology holes due to spatial constraints, the proposed recovery methods may also fail even if a connected path from the source to the destination exists in the network.

Most performance studies of routing protocols so far assume topology holes to occur rarely. However, this assumption is only valid if the network has a high density *and* nodes are uniformly distributed in the whole area. Our scenario shows spatial constraints can cause frequent topology holes even with high network density. Although we choose a road scenario here, the impact of spatial constraints on routing can be found in many other scenarios, such as pedestrians on the street, ships in the river or people in the building, etc.

Generally, routing protocols are making trade-off between high proactivity and low per-node state. While geographic forwarding is nearly a stateless approach, it can not predict topology holes due to its lack of global topology awareness, such as GPSR and INF. In contrast, proactively avoiding forwarding failures often leads to path maintenance on each node, which requires continual state exchange between remote nodes, such as Terminodes routing. Nevertheless, we try to achieve high proactivity while still keeping a small per-node state in our geographic forwarding approach. Our basic idea is to make use of the spatial knowledge to predict and avoid forwarding failures at constant topology holes caused by spatial constraints.

4. Assumptions

Basically, our protocol is based on two assumptions: location awareness and spatial-model awareness.

- *Location awareness* requires the positioning ability and a location service. As described in the introduction, this is also the general prerequisite of geographic forwarding protocols.
- *Spatial-model awareness* is a step forward from location awareness: It requires a node to possess the model information of the geographic space wherein it is located. Thus, a position is not simply a geographic coordinate, but is considered together with the context in its spatial model.

Spatial-model is a high-level abstraction of the spatial objects and their relationships. Generally, we can use *graphs* to model spatial environments with vertices reflecting significant locations and edges representing connectedness between locations. The weight of edges can be used to represent characters of connectedness, such as the physical length. Thus, nodes moving from one location to another location can be considered as moving from one vertex to another vertex along the edges in the graph model. For example, a graph can be used to represent the road's topology as shown in Figure 2. Similarly, we can use a graph to model the internal structure of a building.

The graph model is generic and compact, and can be extracted from external spatial knowledge, such as digital road maps or building charts. For resource-unconstrained nodes such as cars, spatial knowledge is commonly integrated inside the nodes. Resource-constrained devices, such as PDAs or sensors can query spatial knowledge from the outside: static infrastructures such as *Infostations* [13] or mobile *proxy nodes* with sufficient capacity.

Since our focus in this paper is geographic forwarding, for simplicity, we assume that the spatial-model awareness is already available at mobile nodes. The next section describes the basis algorithm of geographic forwarding based on spatial knowledge.

5. Algorithms

The spatial aware geographic forwarding (SAGF) algorithm consists of two methods for forwarding packets: *geographic source route forwarding* (GSRF) and *greedy forwarding*.

5.1 Geographic Source Route Forwarding

Using the graph spatial-model, a source node can calculates the shortest path from its current position to the destination with basis algorithms in graph theory, such as Dijkstra algorithm. For simplicity, we assume that the spatial graph is connected, i.e. that there is a path from any vertex to any other vertex in the graph.

All data packets are marked by source nodes with their destinations' locations. The source node sets the shortest path from its current location to the destination along the spatial graph edges as the *geographic source route* (GSR) in the packet header and switches the packet into GSRF mode. The GSR is based on a list of intermediate vertices along the geographic path in graph, i.e. that GSR is based on *locations* instead of hops.

In the GSRF mode, a data packet should be forwarded along the selected geographic path to the destination. Periodical beacons are used for each node to keep the state information of its immediate neighbors that are located in its radio range. We use the varied inter-beacon intervals to avoid synchronization of neighbors' beacons, similar as the beaconing mechanism used in GPSR.

The forwarding node maps positions of its immediate neighbors into the graph, and forwards the packet to the neighbor that is on GSR and has the shortest graph distance to the destination. Thus, the packet will be forwarded along GSR from a vertex to the next vertex. After a vertex in the GSR is reached, this vertex will be removed from the source route and GSR in the packet header will be updated. Figure 3 presents the SAGF algorithm at the source node in pseudo code.

Source node *S* has a packet for destination node *D*:
N is the list of the neighbors of *S*.

set GSR to shortest_path (*S*, *D*) with Dijkstra algorithm
set packet to GSRF mode

if ($\exists\, n \in N$: *n* is located on GSR and has the shortest
distance along GSR to *D*)
  forward packet to *N*
else
  switch packet to greedy mode

Figure 3. SAGF algorithm at the source node

The complexity of Dijkstra algorithm for shortest path between any two vertices of the graph is $O(n^2)$. However, here *n* does not mean the number of nodes in the network, but the vertices of the graph. Therefore, the complexity of shortest path computation is *constant* with the increasing number of nodes. The scalability of this approach, however, strongly depends on the number of vertices in the graph. Several methods can be used to keep the number of vertices in the graph minimal. For example, we can use *multi-resolution modeling* technique to present the graph on different level of detail. The number of vertices is proportional to the increasing level of detail. Depending on the current requirements, appropriate level is selected to reduce the vertices number in the graph.

DSR also uses source routes for packet forwarding. However, the source routes in DSR are based on the intermediate nodes. Therefore, such source routes are vulnerable to the high mobility of nodes and must be reconstructed whenever a link within the route is broken. In contrast, GSR based on locations is static and keeps constant with the increasing nodes' mobility and no route maintenance is needed for GSR.

However, since GSR is based on locations instead of existing links, it is possible that no neighbor could be found located on GSR. GSRF fails when a forwarding node can not find any neighbor that is located on GSR *and* has a shorter distance along GSR to the destination. The forwarding node then removes the GSR from the packet header and switches the packet to greedy mode.

5.2 Greedy Forwarding

In greedy mode, a packet will be forwarded progressively closer to the destination, the same as in other geographic forwarding protocols, such as GPSR or INF.

Greedy forwarding will repeat until it fails at a local maximum. Then the forwarding node will calculate the shortest path from its current position to the destination along the spatial graph and set this new path as GSR in the packet header. The packet will then be switched back to GSRF mode and forwarded along the new computed geographic path to the destination. If the packet can be forwarded in neither GSRF mode nor greedy mode, the forwarding node will drop the packet after the maximum timeout value of the packet is exceeded. Figure 4 represents the SAGF algorithm at intermediate nodes in pseudo code.

```
Intermediate node R receives a packet P for D:
N is the list of the neighbors of R.
if ( P has been received before ) or ( TTL is exceeded )
    drop packet
else if ( P is in GSRF mode )
    if ( ∃ n ∈ N : n is located on GSR and has the shortest
                   distance along GSR to D )
        forward packet to N
    else
        switch packet to greedy mode
else if ( P is in greedy mode )
    if ( ∃ n ∈ N : n is physically closest to D )
        forward packet to N
    else
        set GSR to shortest_path (R, D) with Dijkstra algorithm
        switch packet to GSRF mode
```

Figure 4. SAGF algorithm at an intermediate node

6. Simulations

To evaluate the proposed approach, we simulate our algorithm in a Java-based environment. Since our routing protocol is independent from the concrete communication technology, and only the high level functionality is concerned, we do not model the physical layer of wireless communications. A simple implementation of MAC layer is used to prevent more nodes from accessing the same wireless channel simultaneously. To compare the behavior and performance of our protocol with existing geographic forwarding protocols, we also implement the GPSR protocol. The next section gives a short description of our simulation environment.

6.1 Simulation Environment

We use two different mobility models in our simulations: the *random waypoint* model, which is commonly used in mobile ad hoc simulations, and a *graph walk* model, which is used to reflect the impact of spatial constraints on the nodes' movement in the real world.

In the random waypoint model, nodes move randomly in an area about 2500 m × 1800 m. Each node begins at a randomly chosen position, picks a new random position as destination, and moves straightly to it at a speed randomly chosen in a range from 30 km/h to 50 km/h. After reaching the destination point, the node makes a pause varied from 0 to 100 seconds, and then moves to another randomly selected point. Each node repeats this behavior for the duration of the simulation run.

In the graph walk model, nodes are not moving randomly, but along the edges of a graph. Figure 5 shows the graph used in our simulations, which models a city center. The graph contains 115 vertices describing significant locations within the city and 150 edges representing roads interconnecting them. The graph also covers an area of approximately 2500 m × 1800 m. It is connected, and the nodes are uniformly distributed on its edges. Each node moves from one randomly chosen vertex to another randomly selected vertex on a shortest path. After reaching the destination vertex, a pause time between 0 and 100 seconds is chosen before the node moves towards the next vertex chosen randomly. Such behavior repeats in the duration of the whole simulation. In our scenario, the nodes are modeling cars driving in the city with the speed varying between 30 km/h and 50 km/h.

For both mobility models, the following numbers of mobile nodes are used in our simulation: 30, 50, 75, 100, 150, and 200. All nodes have the identical radio range of 250 m. Both mobility models use the same constant bit rate (CBR) communication model with 30 nodes sending 64-byte packets at 2 Kbps. The time-to-live (TTL) value of all packets is set to 2 seconds. All simulations are set to run 900 seconds.

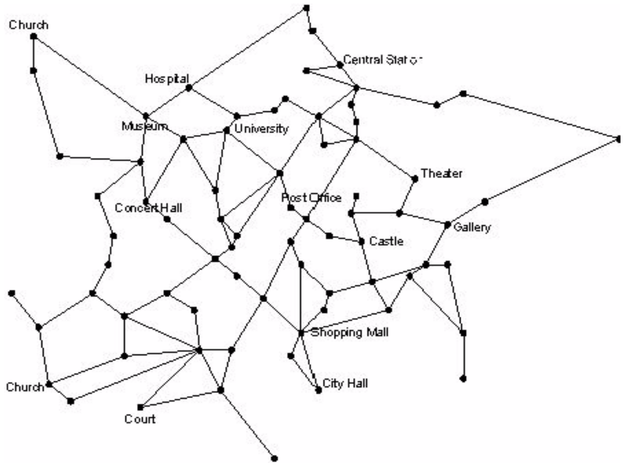

Figure 5. The graph model used in simulation

6.2 Simulation Results

The simulation results presented in this paper are based on 30 randomly generated scenarios based on each mobility model.

We evaluated the spatial aware geographic forwarding using the following metrics:

- *Packet delivery ratio*: The fraction of originated data packets that are successfully delivered to their destination nodes.
- *Packet delivery delay*: The average time between a data packet is originated at the source node and it is delivered to its destination. Packet delay is only measured for packets that are successfully delivered to their destinations.
- *Routing packets overhead*: The total number of packets that are originated by the routing protocol. Here we use the percentage of routing packets instead.

6.2.1 Packet Delivery Ratio

Figure 6 presents the packet delivery ratio of GPSR in both random waypoint and graph walk models. With the number of nodes less than 150, GPSR in graph walk model delivers evidently more packets. This is because of the higher density in graph walk model than in random waypoint model, since the nodes are concentrated on the graph edges instead of uniformly distributed in the whole area. However, with 200 nodes GPSR delivers more packets in random waypoint model. Thus, with the similar high density in both models, GPSR shows its weak point in dealing with topology holes caused by the uneven node distribution. Although the graph's structure in our simulations does not cause frequent topology holes considering the 250 m radio range used, the impact on the performance of GPSR is obvious.

Figure 7 shows the packet delivery ratio of GPSR and SAGF in our implementation. Generally, SAGF delivers more packets than GPSR. This advantage becomes more obvious with a small number of nodes, since topology holes occur more frequently with the decreasing number of nodes. The delivery ratio difference between the two protocols is represented more clearly in Figure 8. We note that SAGF can deliver relatively 18% more packets than GPSR with 30 and 50 nodes. In the range of small numbers of nodes, the advantage of SAGF is more evident, since the low density causes more frequent topology holes.

Although we reflect the spatial constraints using the graph structure, we did not model obstacles that are opaque to radio propagations in our graph model. Thus, in many situations a node can get over the spatial gap between edges and reach nodes in other edges with its 250m-radio range. However, in reality the spatial gaps are filled with buildings that prevent the radio propagation, so the topology holes will occur more frequently than in our simulations, and the advantage of SAGF compared to GPSR should be more evident.

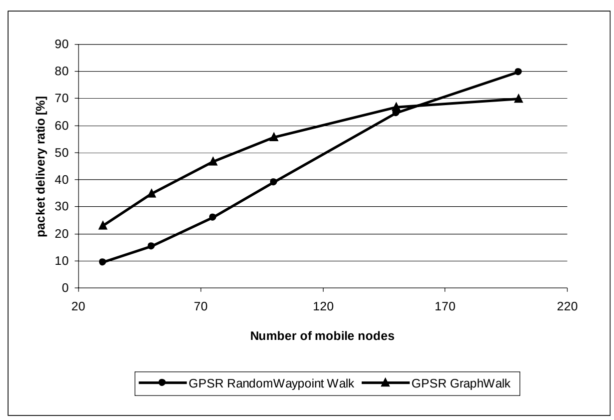

Figure 6. Packet Delivery Ratio of GPSR in random waypoint and graph walk models

6.2.2 Packet Delivery Delay

Figure 9 presents the average packet delivery delay with both GPSR and SAGF in graph walk model. As we see, generally SAGF achieves a lower delivery delay than GPSR. Similar to the delivery ratio, the difference between the two protocols is more evident with a small number of nodes. Thus, SAGF shows its advantage more clearly with frequent topology holes.

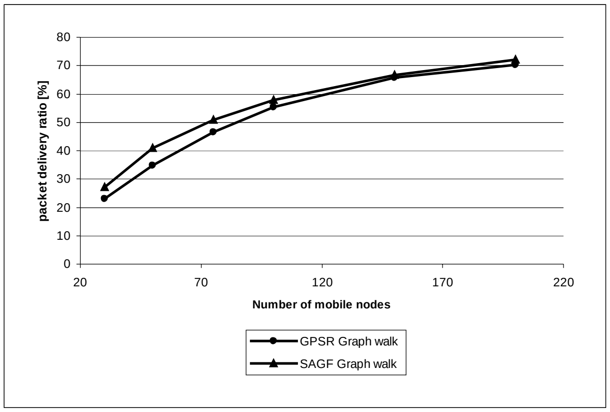

Figure 7. Packet Delivery Ratio with GPSR and SAGF in graph walk model

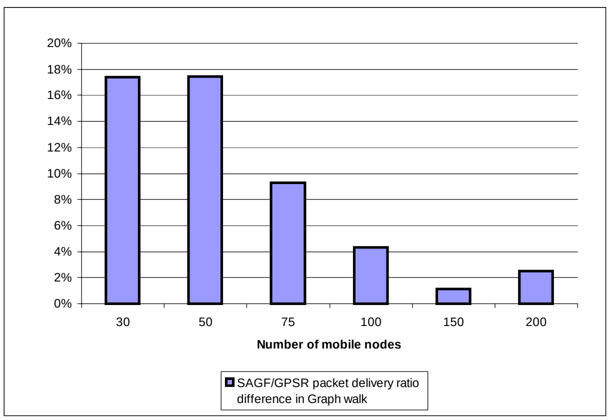

Figure 8. Difference in packet delivery ratio with GPSR and SAGF in graph walk model

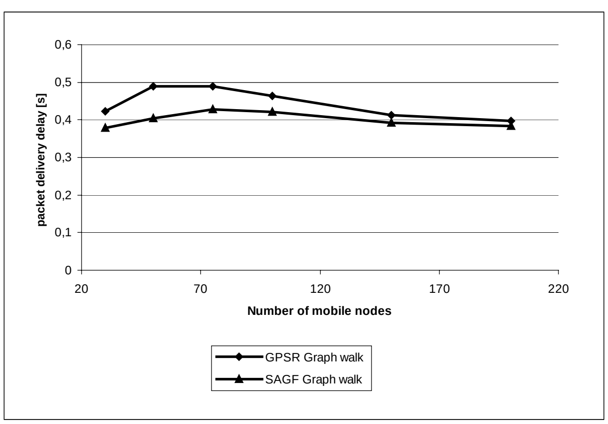

Figure 9. Packet Delivery Delay with GPSR and SAGF in graph walk model

6.2.3 Routing Packet Overhead

Figure 10 presents the routing packet overhead in GPSR and SAGF in graph walk model. The routing packet overhead increases with the number of nodes in both protocols. This is because more packets are delivered with the increasing density of nodes in the network.

However, with any number of nodes in network, SAGF generates significantly less routing control packets than GPSR with more data packets delivered at the same time.

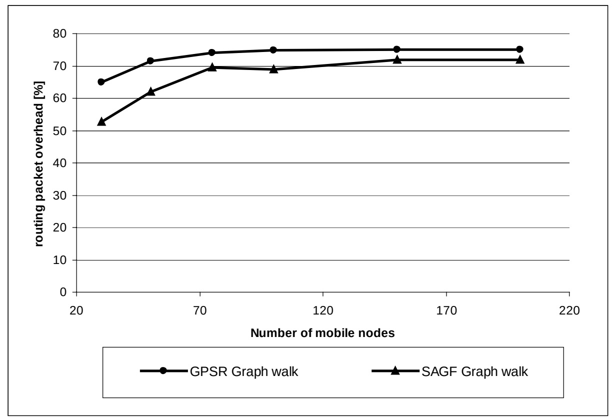

Figure 10. Routing Packet Overhead with GPSR and SAGF in graph walk model

7. Conclusion

In this paper, we presented *Spatial Aware Geographic Forwarding*, a new approach that makes use of external spatial knowledge to improve its proactivity while keeping small per-node routing state. Based on the spatial model of its environment, a source node can select a geographic source route that bypass constant topology holes caused by spatial constraints.

We used a novel graph walk model to simulate the movement of nodes in the real world. Simulation results based on GPSR show that constant holes in node distribution can significantly degrade the performance of geographic forwarding. We implement our approach by extending normal greedy forwarding with spatial knowledge.

Our simulations of both SAGF and GPSR in the graph walk model show that spatial knowledge can effectively improve the routing performance with constant topology holes existing in the network. Compared to GPSR, the implementation of our protocol is much simpler. While GPSR relies on strictly identical radio range for perimeter forwarding algorithm, our protocol does not depend on such an assumption. Thus, our protocol will also work in real-world situations where this assumption can not be guaranteed.

The idea of using spatial knowledge to improve geographic forwarding performance is generic, and can be used in any existing geographic forwarding approach, e.g. GPSR, INF, Terminodes Routing, etc.

The SAGF introduced in this paper is very simple, but it is not the only possible implementation. Further optimizations can be made, for example, considering alternative geographic paths. We will integrate such optimizations in our future implementations. Moreover, we want to integrate obstacles and individual movement profiles of nodes in our graph walk model, in order to provide a more realistic simulation environment for our further studies.

References

[1] J. Broch, D. Maltz, D. Johnson, Y. Hu, and J. Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In *Proc. ACM/IEEE MobiCom*, October 1998.

[2] B. Karp and H. T. Kung. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In *Proc. ACM/IEEE MobiCom*, August 2000.

[3] L. Blazevic, S. Giordano and J.L. Boudec. Self Organized Terminode Routing. *Technical Report, DSC/2001/024*, Swiss Federal Institute of Technology, Lausanne.

[4] M. Mauve, J. Widmer and H. Hartenstein. A Survey on Position-Based Routing in Mobile Ad-Hoc Networks. In *IEEE Network Magazine*, November 2001.

[5] J. Li, J. Jannoti, D. Decouto, D. Karger and R. Morris. A Scalable Location Service for Geographic Ad Hoc Routing. In *Proc. ACM/IEEE MobiCom*, August 2000.

[6] S. Basagni, I. Chlamtac, V. Syrotiuk, and B. Woodward. A Distance Routing Effect Algorithm for Mobility (DREAM). In *Proc. ACM/IEEE MobiCom*, October 1998.

[7] E.M. Royer and C.K. Toh. A review of current routing protocols for ad hoc mobile wireless networks. In *IEEE Personal Communications*, April 1999.

[8] D. Couto and R. Morris. Location Proxies and Intermediate Node Forwarding for Practical Geographic Forwarding. *Technical Report MIT-LCS-TR-824*, MIT, June 2001.

[9] C. Perkins and P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. *In Proc. ACM SIGCOM*, 1994.

[10] D. Johnson and D. Maltz. Dynamic source routing in ad hoc wireless networks. *In Mobile Computing*, edited by T. Imielinski and H. Korth, 1996.

[11] Z. Haas and M. Perlman. The zone routing protocol (ZRP) for ad hoc networks (*Internet-Draft*). 2000.

[12] Y.-B. Ko and N. H. Vaidya. Location-aided routing in mobile ad hoc networks. In *Proc. ACM/IEEE MobiCom*, October 1998.

[13] U. Kubach and K. Rothermel. Exploiting Location Information for Infostation-Based Hoarding. In *Proc. ACM SIGMOBILE MobiCom*, July 2001.

[14] N. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket location-support system. In *Proc. ACM/IEEE MobiCom*, August 2000.

[15] P. Johnansson, T. Larsson, N. Hedman, B. Mielczarek and M. Degermark. Scenario-based Performance Analysis of Routing Protocols for Mobile ad hoc Networks. In *Proc. ACM SIGMOBILE MobiCom*, August 1999.