SWING: Small World Iterative Navigation Greedy Routing Protocol in MANETs

Cong Liu
Department of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431
Email: cliu8@fau.edu

Jie Wu
Department of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431
Email: jie@cse.fau.edu

***Abstract*— Routing is the foremost issue in mobile ad hoc networks (MANETs). In a wireless environment characterized by small bandwidth and limited computational resources, position-based routing is attractive because it requires little communication and storage overhead. To guarantee delivery and improve performance, most position-based routing protocols, e.g. GFG, forward a message in greedy mode until the message is forwarded to a node that has no neighbor closer to the destination, which is called a local minimum. They then switch to a less efficient mode. Face routing, where the message is forwarded along the perimeter of the void, is one example. This paper tackles the void problem with two new methods. First, we construct a virtual small world network by adding virtual long links to the network to reduce the chance of a protocol encountering local minima in greedy mode, and thus decrease the chance to invoke inefficient methods. Second, we use the virtual force method to recover from local minima without relying on face routing. We combine these two methods to be our new purely greedy routing protocol SWING. Simulation shows that SWING finds shorter routes than the state of art geometric routing protocol GOAFR, though with a longer route establishment time. More importantly, SWING is purely greedy which works even if position information is inaccurate, also it can be directly applied to the 3D MANET models. A theoretical proof that it guarantees delivery is given.**

I. INTRODUCTION & PRELIMINARIES

A mobile ad hoc network (MANET) is comprised solely of wireless stations. The communication between source and destination nodes may require traversal of multiple hops because of limited radio range. Existing routing algorithms can be broadly classified into topology-based and position-based routing protocols. Topology-based routing determines a route based on network topology as state information, which needs to be collected globally on demand as in routing protocols DSR [5] and AODV [12] or proactively maintained at nodes as in DSDV [11].

The scope of this paper is focused on position-based routing, also called geometric or geographic routing. Position-based routing protocols are based on knowing the location of the destination plus the location of neighbors in each node. They are attractive for MANETs for the following reasons: (1) they incur low route discovery overhead compared to flooding-based approaches in on-demand topology-based routing protocols, and hence save energy and bandwidth, and (2) they are stateless in the sense that nodes need not maintain per-destination information, and only neighbor location information is needed, either from a GPS [3] or through other means, to route packets.

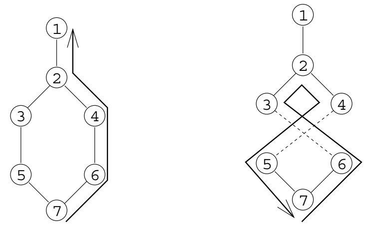
Most position-based routing protocols use greedy forwarding as their basic operation. In greedy forwarding, a forwarding node makes a locally optimal greedy choice in choosing the next hop for a message. Specifically, if a node knows its neighbors' positions, the locally optimal choice of next hop is the neighbor geographically closest to the destination of the message. Greedy forwarding, however, fails in the presence of a void (also called a local minimum or a dead end) where the only route to the destination requires a packet move temporarily farther in geometric distance from the destination.

In order to recover from a local minimum, most existing protocols switch to a less efficient mode, such as the face routing mode. Face routing [2] (also called perimeter routing or planar graph traversal) on a connected network theoretically guarantees the delivery of packets. Face routing runs on a planar graph, in which the message is routed around the perimeter of the void (face) surrounded by the edges using the right-hand rule. Examples of the existing greedy-face combinations are GFG [1], its variant GPSR [6] and GOAFR [8].

By observing simulations, we notice the following problem with the greedy-face combination. While a message always travels toward the destination in the greedy mode, it loses its direction in face mode. And in certain topologies, voids can lead to excessive retracing. This problem is mitigated by GOAFR [8], which restricts the traversal of the messages in face mode using a series of ellipses increasing in size and effectively decreases the average route length.

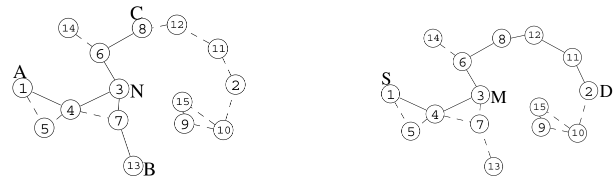
Recently, a new routing algorithm was proposed [13], which does not require geographic information for all of the nodes in the network. The algorithm is based on the use of a set of virtual coordinates which are calculated by averaging the x-y coordinates of each node in the network with its nearest neighbors.

It is inevitable that face routing could fail because of location errors in both virtual position and position from a GPS. Results in [15] show that even small location errors (of 10% of the radio range or less) can in fact lead to incorrect (non-recoverable) geographic routing with noticeable performance degradation. An example of a failure in face routing is shown



(a) (b)

Fig. 1. Location error in face routing caused by inaccurate location information. (a) real position (b) derived virtual position.



(a) (b)

Fig. 2. Examples of (a) virtual long link in N and (b) virtual force based greedy protocol using virtual long links.

in Figure 1(a) and Figure 1(b). The configuration of the derived virtual positions is possible because each of the virtual positions is calculated by averaging the positions of the 1-hop neighbors. Many papers, such as [15], have proposed new geographic routing algorithms to alleviate the effect of location errors on routing in wireless ad hoc networks. The results show that without global knowledge about the network, it is not possible to solve all the problems in face routing caused by location errors completely.

Unlike GOAFR [8], this paper tackles the above problem from two different methods. The first method is to construct a virtual small world network [10]. Specifically, each node in the network has some remote contacts connected by *virtual long links* (VLLs). Each VLL consists of multiple consecutive physical links. To be scalable, the length (in hops) of the VLLs conform to a 2-exponent power-law distribution, which is analogous to [7]. The purpose of introducing VLLs is mainly to reduce local minima for a greedy routing and hence the chance of turning to face mode.

The second method is a virtual force (VF) based greedy method. Its purpose is to reduce routing in face mode when the greedy mode needs to recover from a local minimum. In this method, a message is forwarded along the decreasing gradient of the composition of the VFs (CVF). Each VF has a source and the VF decreases as the distance from the source increases. The destination is the only source of a negative VF. Whenever the greedy method fails in a local minimum of the CVF, a new source of positive VF is added to the local minimum to remove the local minimum and recover the greedy routing. We call it iterative navigation greedy (ING) method. In ING, a list of the past local minima needs to be stored in the message.

ING itself is not efficient. One reason for this is when the only path to the destination is close to a local minimum which has become the source of a positive VF, the message might be deviated from the destination. However, when running in a virtual small world network ING has an interesting improvement, because the VLLs can help the message to "jump across" the source of the positive VF. Thus we have our new purely greedy protocol – *Small World Iterative Navigation Greedy protocol* (SWING). One important result of this paper is that it is theoretically proved that SWING guarantees delivery.

The advantage of SWING over the greedy-face combinations is that a message is always forwarded in awareness of the destination. Also, the pure greedy method has automatically solved the problem of localization errors on face routing [15] and it is applicable to the 3D networks. Simulation results shows that SWING guarantees delivery and the performance of SWING in terms of route length is better than that of the state of art position-based routing protocol GOAFR. However SWING has a longer route establishment delay than GOAFR. So we also present a trade-off variation of SWING, called direct retrial, which has both shorter route establishment delay and route length than GOAFR but fails to guarantee delivery in rare situations. We believe SWING will shed light on a new methodology for position-based routing in MANETs.

Extensive simulation is conducted to analyze SWING and to compare it with the greedy-face combinations, including GOAFR. In the simulation, we improve the performance of GOAFR with VLLs, CDS [4] and a sooner back algorithm [4]. Simulation results show that SWING is slightly better than GOAFR in terms of average route length. SWING with direct retry has better route establishment delay and route length than GOAFR, but fails to guarantee delivery in rare situations.

The rest of the paper is organized as follows. We present SWING in Section II, which includes the construction of the virtual small world network, the greedy routing method in small world network, and the iterative navigation greedy method. In Section III, we perform simulation analysis and comparison between SWING and different greedy-face combinations. Finally, Section IV concludes the paper.

II. Small World Iterative Navigation Greedy Routing Algorithm

A. Virtual Small World Network

To construct a virtual small world network, a number of virtual long links (VLLs) is added to each node in the network. The method is that each node periodically sends out VLL discovery messages which go away and then come back to report a VLL. For space limitation, we don't elaborate it here. Reader please refer to our previous work in [9].

Figure 2(a) is an example of the VLLs of node N. In this example, the three VLLs of node N in the random network are NA $(3,4,1)$, NB $(3,7,13)$ and NC $(3,6,8)$.

B. Virtual Force Based Greedy Routing in the Virtual Small World Network

The routing approach presented in this section defines a virtual force and route message using this virtual force (VF) instead of the distance to the destination. The introduction of VF is useful to the protocol to be presented in the next subsection. We define the VF between two points as:

$$force(a, b) = \frac{1}{1 + d(a, b)} + \lambda e^{-\lambda d(a,b)} \quad (1)$$

The term $\frac{1}{1+d(a,b)}$ makes sure that the value of VF is not negligible from any distance and decreases smoothly as the distance between the points increases. The term $\lambda e^{-\lambda d(a,b)}$ (with a large λ) makes sure that the force is extraordinarily big (which is equal to λ) when the 2 points overlap.

The composition of VFs (CVF) in a point n from a collection L of points is the sum of the forces between n and each point L_i in the collection.

$$force(n) = \sum_{0<i<|L|} force(n, L_i) \quad (2)$$

Assume that each node collects k-hops omni-directional link information, i.e. it maintains the omni-directional shortest paths to k-hops neighbor nodes. A message is forwarded along to the next hop on the next available best path, which is either a k-hops omni-directional link or a VLL. The best path is a path with the minimum force, and the force of a path is equal to the minimum force of the nodes on the path (here L contains the single source of force – the destination):

$$force(P) = \min_{0<i<|P|} force(P_i, L) \quad (3)$$

The virtual force based greedy routing in the virtual small world network is given in Algorithm 1.

Algorithm 1 Virtual force based greedy protocol using virtual long links

1: List the paths which contain the shortest path to all neighbor nodes and all virtual long links.
2: Calculate the virtual force in these paths from the destination.
3: Send the message to the next node on the path with the smallest virtual force.
4: Repeat the above steps until the message gets to the destination, a local minimum, or reaches the maximum hop count.

An example of this routing protocol is shown in Figure 2(b), where a message is sent from the source S to the destination D successfully. While a traditional greedy algorithm will fail on the local minimum m, our algorithm succeeds, since there is a VLL NC $(3, 6, 8)$ through which a message in m knows that node number 8 is closer than m (3) to D, and thus NC has a smaller VF than Nm. That is, the local minimum m is circumvented by the VLL NC.

In order to prevent loops in the above protocol, we piggyback in each message the best path it chose before it was forwarded, and this best path is include in the forwarding decision in the next host of the message. Below, we give some theoretical results from our previous work [9] directly without proof.

Lemma 1: If a message m piggybacks its current path, and m travels from node A to node B through a series of paths $P_1, P_2, \ldots, P_n$, and B is the end of P_n, then the distance $force(A, D) > force(B, D)$, where D is the destination of M.

Theorem 1: If a message m carries its current path, the VF based greedy protocol with VLLs is loop free (temporary loop is not counted).

Since the VF based greedy protocol with VLLs is loop free, it is arguable that the protocols produced by replacing the regular greedy algorithm in the greedy-face combinations guarantee delivery.

C. Routing with Small World Iterative Navigation Greedy (SWING)

It is inherited from the family of greedy algorithms that protocol 1 can go to a local minimum and fail. The best part of SWING is the iterative method that allows the message to continue to travel to the other parts of the network after failures in local minimums. In order to prevent the message from going along routes that have been explored in the previous failure trials, our method is to use a repulsive list. Whenever a message fails, the position of the local minimum node is added as a failure point to the repulsive list. We also use an attractive list, which usually contains the single attractive point – the destination, but can contain multiple destinations in geocasting.

In SWING each message maintains a list R of positions of local minima besides the position of the destination D. Given R and D, the CVF in a point n is defined in Equation 4. The other *force*s in this equation are defined in Equation 1 and Equation 2, where $lambda$ should be large enough to recover routing from a local minima.

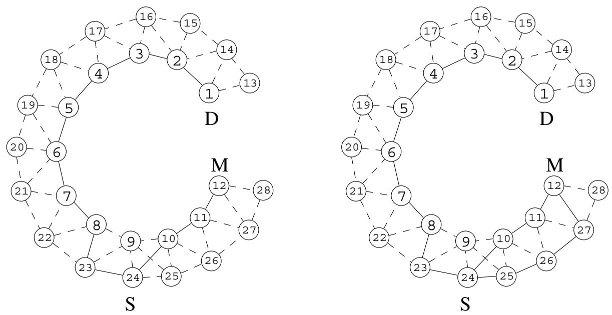
$$force(n) = \frac{\sum_{0<i<|R|} force(n, R_i)}{|R|} - 2 \cdot force(n, D) \quad (4)$$

$$force(P) = \max_{0<i<|P|} force(P_i) \quad (5)$$

Algorithm 2 Small World Iterative Navigation Greedy

1: List the paths which contain the shortest path to all neighbor nodes and all virtual long links.
2: Calculate the virtual force in these paths from the destination.
3: Send the message to the next node on the path with the smallest virtual force.
4: If the current node M is a local minimum under the CVF, add M to list R. Come back to source and repeat the above steps until the message gets to the destination or reaches the maximum hop count.

Figure 3(a) is an example of a running of the SWING protocol, where a message sent by source S to destination D succeeds in the second try. In this example, it first fails



(a) SWING (b) SWING with direct retry

Fig. 3. Examples of routing with SWING and SWING with direct retry

in node M (which is a local minimum). Then the message comes back to S and starts again with M added to its list of local minima R. The message successfully gets to D in the second iteration. The route from S to D in this example is (24-10-11-12-11-10-24-23-8-7-6-5-4-3-2-1).

Let $F_0 = force(n, n)$ (the force between a point and itself) and $F_1 = \max_{i \neq j} force(i, j)$ (the maximum force between two different points), we have:

Theorem 2: There exists an $N < \infty$, such that when $\frac{F_0}{F_1} > N$, SWING guarantees delivery.

Proof: In theory 1, we have proved that SWING is loop free within each iteration, so each iteration in SWING will finish in finite hops if the network has a finite number of nodes. Without loss of generality, we can assume that there is a path $P = \{P_1 = source, P_2, \ldots, P_n = destination\}$ such that the routing message will not go to the destination before it has traveled through this path. In the following, we will use mathematical induction to prove our theorem. Assume that a message M goes to P_i before it goes to P_{i-1} for finite times T_{i-1} for all $i \leq k$. We need to prove that M goes to P_{k+1} before it goes to P_k finite times T_k. If M does not go to P_{k+1} before it goes to P_k finite times, there must be a path $L = \{P_k = L_1, L_2, \ldots, L_n\}$ such that M goes along L from P_k to L_n for infinite times before it goes to P_{k+1}, since the network is finite and the possible path starting from P_k is finite. We will prove that there is an N that makes such an L_n impossible. Suppose in time t_0, M is in P_k, L is the link with the largest $force$ for M, K ($K < T_1 \cdot T_2 \cdot \ldots \cdot T_{k-1}$) is the size of the repulsive list R in M, T_x ($T_x > 0$) is the times that M has went to L_n, we have:

$$force(P_{k+1}) = force(P_{k+1}, destination) -$$

$$\frac{1}{K} force(P_{k+1}, R) > -\frac{F_1}{K}$$

$$force(L_n) = sm(L_n, destination) -$$

$$\frac{1}{K}(force(L_n, (R \backslash L_n)) + T_x \cdot F_0) < F_1 - \frac{T_x \cdot F_0}{K}$$

$$force(P_{k+1}) - force(L_n) > \frac{T_x \cdot F_0}{K} - \frac{(K+1)F_1}{K}$$

Let $N = \frac{K+1}{T_x} < T_1 \cdot T_2 \cdot \ldots \cdot T_{k-1}$. When $\frac{F_0}{F_1} > N$, $force(P_{k+1}) > force(L_n)$, and M will go to P_{k+1} instead of L_n. ∎

D. SWING with direct retry

The basic idea of SWING with direct retry is that the message doesn't go back to the source after failure in each iteration, but starts from the local minimum. That is possible since a new repulsive force is added to the local minimum at each iteration in SWING which makes the local minimum no longer a local minimum.

In SWING with direct retry, a message is routed greedily to the next node that has the smallest CVF. Whenever a message is blocked in a local minimum, the position of the current local minimum M is added to the list of local minima R such that with the new additional VF, M is no longer a local minimum, and the message can route greedily in the CVF again. The SWING protocol is shown below as Algorithm 3. Algorithm 3 differs from Algorithm 1 and 2 only in the last item.

Algorithm 3 Small World Iterative Navigation Greedy

1: List the paths which contain the shortest path to all neighbor nodes and all virtual long links.
2: Calculate the virtual force in these paths from the destination.
3: Send the message to the next node on the path with the smallest virtual force.
4: If the current node M is a local minimum under the CVF, add M to list R. Repeat the above steps until the message gets to the destination or reaches the maximum hop count.

An example of SWING is shown in Figure 3(b). In this example, the message starting in $S(24)$ fails on the first try in node $M(12)$. After adding m to list M, it continues routing greedily under the new CVF and finally succeeds to go to $D(1)$. Note that a path result from SWING is indeterministic since it relies on the VLLs which are added indeterministically. The route from S to D in this example is (24-10-11-12-27-26-25-24-23-8-7-6-5-4-3-2-1).

E. Overhead & Scalability Analysis

For space limitation, we give the results directly. The amortized communication overhead for establishing VLLs per VLL message interval is $O(MinHops + 1)$. The amortized additional communication overhead of a routing message is the position information of the previous failures. Let C_M be the number of VLLs that can be stored in each node, D be the average node degree and k be the hop count of neighbor information exchanged, the per-node memory overhead is $O(D^k) + O(C_M)$. The computation overhead for message forwarding is $O(D^k \cdot |R|)$.

III. Simulation

A. Assumptions and Evaluation Metrics

The objective of our simulation is to measure and compare the performance of different geometric routing protocols. The

Algorithm Name	G	F	CDS	VLL	BE	SB	ING
GREEDY	√						
GREEDY(VLL)	√			√			
SWING	√						√
SWING(VLL)	√			√			√
GFG	√	√				√	
GFG(VLL)	√	√		√		√	
GFG(CDS)	√	√	√			√	
GFG(VLL+CDS)	√	√	√	√		√	
GOAFR	√	√			√	√	
GOAFR(VLL)	√	√		√	√	√	
GOAFR(CDS)	√	√	√		√	√	
GOAFR(VLL+CDS)	√	√	√	√	√	√	

TABLE I
CLASSIFICATION OF THE SIMULATED ROUTING ALGORITHMS.

Parameter	Value
Field size	1000×1000
Transmission range	100
Transmission delay	10(ms)
Number of nodes	$150 \sim 450^*$
Network degree	$4.71 \sim 14.13$
Max routing hops count	$(*)$
Number of VLLs	$0 \sim 5$
Minimum length of a VLL	2
Max iterations in SWING	20
Time run for VLLs	10000(ms)
Time for running routing	10000(ms)

TABLE II
EXPERIMENT SETTINGS.

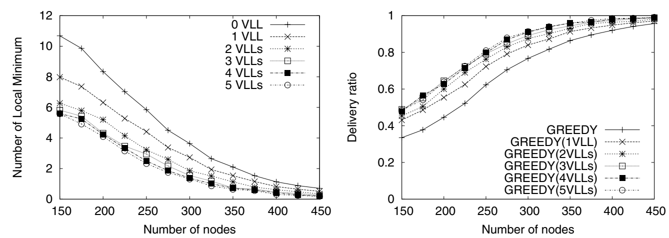
assumptions are: (1) an ideal MAC layer: collision free with constant transmission delay, (2) all needed position information is available without additional communication overhead, and (3) mobility is not considered.

The metrics we use to evaluate a protocol is delivery ratio, route establishment delay and route length. Delivery ratio is the ratio of the message delivered to the destination over the total amount of messages sent. Route establishment delay is counted as the number of hops the message makes to discover a route traveled totally. The route length is the length of the route discovered in hops. It is always shorter than the route establishment delay. For example, in Figure 3(a), the route establishment delay equals the hops of the path that the route discovery message traveled, i.e., (24-10-11-12-11-10-24-23-8-7-6-5-4-3-2-1). And the route length is the length of the shortest path derived from the above path, i.e., (24-23-8-7-6-5-4-3-2-1).

B. Simulation Environment and Settings

Simulations were conducted on three protocol families: the Greedy family, the GFG family and the GOAFR family. Table I shows all of these protocols (in rows) and the algorithms used in each of them (in columns). The algorithms used include the Greedy algorithm (G), the Face algorithm (F), using connected dominate set (CDS) for face mode [4], using virtual long link (VLL) in Greedy mode, using bound ellipse (BE) in face mode [8], the sooner back (SB) algorithm [4] (which makes the message routing in the face mode return to greedy mode faster when the current node has a neighbor whose distance to the destination is shorter than that of the last local minimum) and the VF based iterative navigation greedy method (ING).

We do the simulation on our custom simulator. In each experiment a connected graph with N (ranging from 150 to 450 in different experiments) nodes is randomly generated in a 1000×1000 square. After that, we let the simulator run for a period of time which is sufficient for the nodes to grow the virtual long links. Then, for each node, messages are added to be sent in the routing protocols listed in Table I. The destination of these messages is another node chosen randomly. We run each experiment 100 times to get the average value.



(a) Local minima (b) Delivery ratio

Fig. 4. Number of VLLs v.s. Average number of local minima and the delivery ratio of pure greedy routing.

The network density in our experiment ranges between two extremes. The sparse extreme is the only region where the shortest path is usually much longer than the direct connection between the source and the destination. This region is critical for routing algorithms, where finding a good path at low cost becomes a nontrivial task and a real challenge for position-based routing. In the dense region, all algorithms have similar performance since they all degrade to pure greedy. All the important parameters in our simulation are shown in Table II.

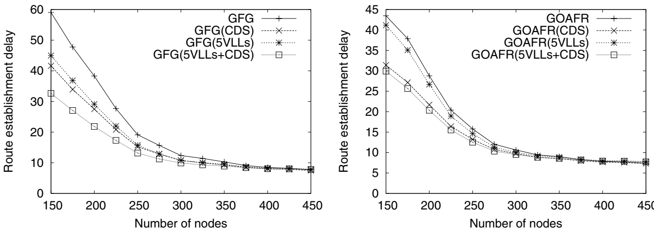
C. Simulation Results

Figure 4(a) shows the number of local minima decreases as the number of VLLs per node in the network increases. In the figure the number of local minima decreases rapidly before the the number of VLLs reaches 3. In the following experiments, we use 5 VLLs per node for all algorithms.

Figure 4(b) shows that the delivery ratio of the pure greedy protocol increases as the number of VLLs increases. Like the previous figure, the first 3 VLLs are more effective than the following ones.

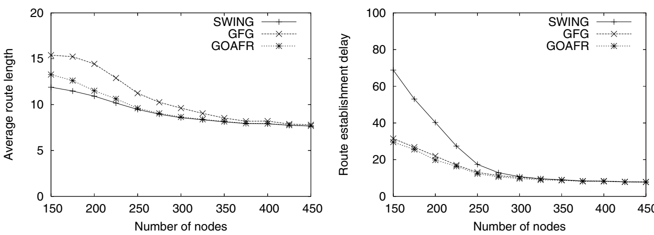
Figure 5(a) and Figure 5(b) are simulation results (in terms of route establishment delay) for comparison between the protocols in the GFG family and the GOAFR family. We use the best parameter setting for GOAFR, i.e., the major axis of the ellipse is $1.2|\overline{st}|$ and the multiple factor is $\sqrt{2}$ [8]. The comparison shows that GFG(5VLLs+CDS) and GOAFR(5VLLs+CDS) are the best in their families. We will use them to compare with SWING later.

Figure 6(a) compares the average route length of 3 routing



(a) The GFG family (b) The GOAFR family

Fig. 5. Number of VLLs v.s. route establishment time of the GFG family and the GOAFR family.

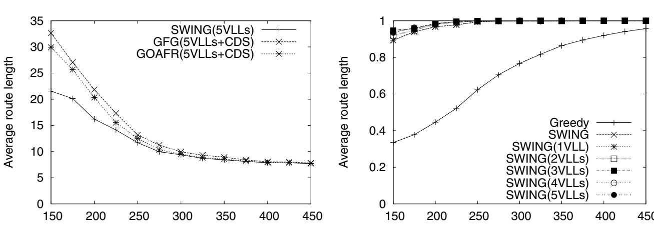


(a) Average route length. (b) Route establishment delay

Fig. 6. Comparison of Average route length and Route establishment delay between SWING, GFG and GOAFR.

protocols: SWING, GOAFR and GFG. We found that SWING is the best in terms of the average route length. Unfortunately, SWING has a longer route establishment delay than both GOAFR and GFG, as shown in Figure 6(b). However, in the application where the data transmission is often in 2 stages: route discovery and the transmission of large volume of data, SWING is a better choice.

As an alternative, SWING with direct retry is good for short route establishment time. The comparison of the 3 protocols: SWING with direct retry, GOAFR and GFG is shown in Figure 7(b). The drawback of SWING with direct retry compared to SWING is that it doesn't guarantee delivery, as shown in Figure 7(a). However, non-delivery will only occur when the network is extremely sparse. For example, to get a graph with 150 nodes in our setting, a computer usually needs to generate 20,000 random graphs. Thus SWING with direct retry almost



(a) Comparison. (b) Delivery ratio

Fig. 7. Comparison of Route establishment time between SWING with direct retry, GFG and GOAFR. Delivery ratio of SWING with direct retry.

guarantees delivery in reality.

To summarize the simulation, our new purely greedy position-based routing protocol SWING has an interesting improvement in terms of average route length over the greedy-face combinations. As a good choice for short route establishment delay, SWING with direct retry almost guarantees delivery except in extremely sparse density which seldom happens in real connected networks.

IV. Conclusion

The paper has presented a research in position-based routing in MANETs. This paper solves the problem of suboptimality that arises from void-recovery protocols. Rather than attempting a more optimal face-routing protocol, we improve routing from two different angles. First, we constructed a virtual small world network to reduce the chance of a protocol encountering local minima in greedy mode. Second, we used the virtual force method to recover from local minima without relying on face routing. In simulation, our algorithm SWING and SWING with direct retry were shown to be competitive with the state of art geometric routing protocol GOAFR with improvements in the metrics of route length and route establishment delay respectively.

References

[1] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. In *Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, 1999.

[2] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *ACM Wireless Networks*, 7(6):609–616, 2001.

[3] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. *IEEE personal communications, Special Issue on Smart Spaces and Environment*, 7(5):28–34, 2000.

[4] S. Datta, I. Stojmenovic, and J. Wu. Internal node and shortcut based routing with guaranteed delivery in wireless networks. *Cluster Computing, Special issue on, Mobile Ad Hoc Networks*, 5(2):169–178, Apr. 2002.

[5] D. Johnson and D. Maltz. Dynamic source routing in ad-hoc wireless networks. In *Proc. of ACM SIGCOMM*, 1996.

[6] B. Karp and H.T. Kung. [gpsr]: greedy perimeter stateless routing for wireless networks. In *ACM MOBICOM*, 2000.

[7] J. Kleinberg. The Small-world phenomenon: An algorithmic perspective. In *Proc. of the 32nd ACM Symposium on Theory of Computing*, 2000.

[8] F. Kuhn, R. Wattenhofer, and A. Zollinger. Worst-case optimal and average-case efficient geometric ad-hoc routing. In *ACM Mobihoc*, 2003.

[9] C. Liu and J. Wu. Position-based routing using virtual small world in manets. In *Proc. GLOBECOM*, 2006.

[10] S. Milgram. The small world problem. *Psychology Today 1, 61 (1967)*.

[11] C. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. *Computer Communication Review*, 24:234–244, 1994.

[12] C. Perkins and E. Royer. Ad hoc on-demand distance vector routing. In *Proc. of 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999.

[13] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica. Geographic routing without location information. In *ACM MOBICOM*, 2003.

[14] A. Renyi. On measures of entropy and information. In *Fourth Berkeley Symposium on Mathematical Statistics and Probability*, 1960.

[15] K. Seada, A. Helmy, and R. Govindan. On the effect of localization errors on geographic face routing in sensor networks. In *ACM Proceedings of the third international symposium on Information processing in sensor networks*, 2004.