

SWING: A Small World Iterative Navigation Greedy Routing Protocol in MANETs

Cong Liu and Jie Wu

Department of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431

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. A theoretical proof that it guarantees delivery is given.

Keywords: position-based (geometric) routing, mobile ad hoc networks (MANETs), simulation, small world model

I. INTRODUCTION

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 [7] and AODV [18] or proactively maintained at nodes as in DSDV [17].

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 [4] 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 [3] (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. Example of the existing greedy-face combinations are GFG [2], its variant GPSR [8] and GOAFR [11].

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 [11], 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 [19], which does not require geographic information for all of the nodes in the network, but assumes that peripheral nodes located at the boundary of the region of interest have location information. 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, while keeping the coordinates of the peripheral nodes on the boundary fixed.

It is inevitable that face routing could fail because of location errors in both virtual position and position from a GPS. Results in [21] 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

This work was supported in part by NSF grants, ANI 0073736, EIA 0130806, CCR 0329741, CNS 0422762, CNS 0434533, and CNS 0531410. Contact address: jie@cse.fau.edu

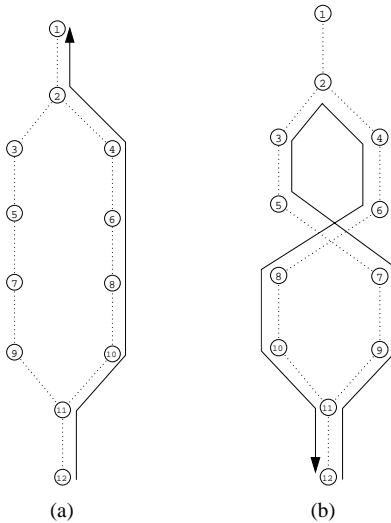


Fig. 1. Location error in face routing caused by inaccurate location information. (a) real position (b) derived virtual position.

in Figure 1(a) and Figure 1(b). The configuration of the derived virtual positions is possible because each of the virtual position is calculated by averaging the positions of the 1-hops neighbors. Many papers, such as [21], 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 [11], this paper tackles the above problem from two different methods. The first method is to construct a virtual small world network [15]. 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 [9]. 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 it. After this addition, the message is no longer in a local minimum, and the greedy method is recovered. The method is an iteration of greedy forwarding and local minimum removal. So, we call it iterative navigation greedy (ING) method. In ING, a list of the past local minima need to be stored in the message.

ING itself is not efficient. One reason for this is that the only path to the destination is usually close to a local minimum; and after the local minimum becomes the source of a positive

VF, the message might be cut 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 also 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 [21] as well as on GFG and GOAFR. 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 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 [5] and a sooner back algorithm [5]. 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. Section II briefly summarizes the related works. We present SWING in Section III in which the construction of the virtual small world network, the greedy routing method in small world network, and the iterative navigation greedy method are presented in different subsections. In Section IV, we perform simulation analysis and comparison between SWING and different greedy-face combinations. Overhead and scalability of SWING is analyzed in Section V. Finally, Section VI concludes the paper.

II. RELATED WORKS

A. Position-based Routing

Localized algorithms are desirable in MANETs, in which nodes make routing decisions based only on information about its neighboring nodes and the position of the destination. One such method, the greedy routing algorithm, in which each node forwards the message to its neighbor closest to the destination, is based on the location information supplied by GPS [4].

In *greedy face greedy* (GFG) [2] and its variant *greedy perimeter stateless routing* (GPSR) [8], when a packet reaches a region where greedy forwarding is impossible, the algorithm recovers by routing around the perimeter of the region. The right-hand rule is used to route around the face, which requires a planar graph. A graph in which no two edges cross is known as planar. The *relative neighborhood graph* (RNG) [23] and *Gabriel graph* (GG) [6] are two planar graphs.

In [5] Datta, Stojmenovic and Wu improved GFG based on the concept of dominating sets. They propose to run GFG routing on the internal nodes. The network of internal nodes defines a connected dominating set (CDS), and each node must be either internal or directly connected to an internal node.

An extension to GFG/GPSR, *greedy other adaptive face routing* (GOAFR) [11], avoids routing beyond some radius by branching the graph within an ellipse of exponentially growing size to achieve worst-case optimality and average-case efficiency in term of average route length.

GVG [16] has its face forwarding section run over arbitrary non-planar graphs by adding virtual nodes at the crosses of the links. Representative examples, however, show that GVG has large average route length.

Routing algorithms based on DFS are proposed in [22]. SWING defers from the DFS based routing algorithms in the follows: (1) Strictly speaking, SWING is not a DFS. (2) A message or a node only memorizes a small fraction of the past traffic. (3) In SWING, messages are driven by the CVF.

Virtual position-based routing was proposed [19]. A method to alleviate the effect of location errors on routing in MANETs is proposed in [21].

B. The Small World Model

The small world model [15] corresponds to a phenomenon in a social network where any two people have “six degrees of separation”. More recently, it has been shown in [24] that this phenomenon is pervasive in many natural and artificial complex networks, and is captured by two measurements: small average path length and high clustering coefficient (defined as the average fraction of pairs of neighbors of a node that are also neighbors of each other).

Kleinberg [9] defined an infinite family of random network models that seek a simple framework that encapsulates the paradigm of Watts and Strogatz – rich in local connections, with a few long range connections. Rather than using a ring as the basic structure, it uses a 2-dimensional $m \times m$ grid and allows each node to have a directional long link to a remote contact with the distance in the r -exponent power-law distribution. [24] also proved that there is a unique “navigable” model ($r = 2$) within the family for which decentralized algorithms are bound by $O(\log^2 m)$. The extension to the navigable hierarchical network is discussed in [10].

Terminode [1] is based on the small world model that does not always forward packets directly towards the destination. In order to optimize routing in case of voids in the network topology, a node finds a list of remote contacts distributed all over the network, to which it maintains a good path. To find a route to the destination, a node asks its remote contacts that in turn ask their remote contacts, and so on. The right remote contacts found are added as a loose source path to the header of the data packets. Though Terminode finds short paths, it uses some sort of broadcast to discover routes and it does not guarantee delivery.

In [13] [14], the authors study the throughput capacity of hybrid MANETs, in which they investigate the use of limited infrastructure, in the form of wires, for improving the energy efficiency of MANETs.

III. SMALL WORLD ITERATIVE NAVIGATION GREEDY ROUTING ALGORITHM

A. Assumptions

The basic idea of SWING has been presented in Section I. To simplify our discussion, we make the following common assumptions: (1) An ideal environment in which radio ranges of all nodes are exact and symmetric and that there is no packet loss. (2) All nodes know their own positions, either from a GPS device [4], if outdoors, or through other means. This assumption becomes more and more realistic with the advent of inexpensive and miniaturized positioning systems. (3) A location registration and lookup service that maps node addresses to locations [12]. Queries to this system use the same geographic routing system as data packets. (4) All nodes are stationary or node movement is ignorable as in other literature. Although node mobility is one of the most important considerations in MANETs, we assume that the routing process is very fast relative to node movement. (5) To highlight the advantage of SWING, we assume that each data transmission has two phases: route discovery and data delivery, which is reasonable when the data to be transmitted is great in volume.

B. Virtual Small World Network

In this section, we will present our method to construct a virtual small world network by adding a number of virtual long links (VLLs) to each node in the network such that the distance (in hops) to a remote contact is under the power-law distribution. The method is that each node periodically sends out VLL discovery messages which go away and then come back to report a VLL. The first problem here is how to decide the maximum hops and the direction of a message. The second problem is how to select a subset of most valuable virtual long links when the storage in each node is limited.

When a VLL message (message for short in this subsection) is sent by a node (initiator), the message should go in a different direction from that of the previous messages so that the messages can explore different parts of the network. Also, the maximum hops of a message should be appointed in such a way that the algorithm is scalable.

The maximum hops of a message is decided conforming to the power-law distribution as follows:

$$MaxHops = MinHops + \log_2\left(\frac{1}{p}\right) \quad (1)$$

Here p is a random value between 0 and 1, and $MinHops$ is a constant, which equals 2 in our experiment.

The reason for choosing the 2-exponent power-law distribution is two-fold. First, a analytical study in [9] shows that there is an analogous small world model (in a $m \times m$ grid) for which decentralized algorithms are bound by $O(\log^2 m)$. The second reason to use the 2-exponent power-law distribution is for scalability: only when $r \leq 2$ will the average VLL length converge.

It is desirable that each message chooses a random direction to go to explore different parts of the network. We use an imaginary point that is about 1-hop distance away from the

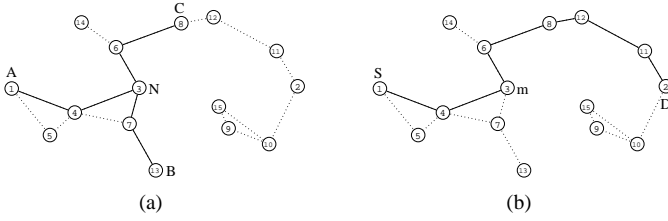


Fig. 2. Examples of (a) virtual long link in N and (b) virtual force based greedy protocol using virtual long links.

initiator of the message and the direction of the imaginary point to the initiator is chosen randomly. Then the message is let go and driven away by a virtual force (VF) from the imaginary point. This VF is inversely proportional to the distance between the imaginary point and the message's position. The message always tries to jump to the next node with the smallest virtual force (e.g., the node farthest from the imaginary point).

Not only should a message choose a random direction to go, it should preferably go to an area that has not been explored by earlier messages. Our method to accomplish this is to define a list of points that give VFs. This list includes the initiator, the imaginary point that gives a direction, and the endpoints of the VLLs established previously.

We define the VF between two points as:

$$force(a, b) = \frac{1}{1 + d(a, b)} + \lambda e^{-\lambda d(a, b)} \quad (2)$$

The terms on the right side of the equation are not chosen randomly. The left term makes sure that the value of VF is not negligible from any distance and decreases smoothly as the distance between the points increases. The right term (with a big enough λ) makes sure that the force is extraordinarily big (which is equal to λ) when the 2 points overlap. This is used to prevent the VLLs from overlapping in their endpoints.

We define the composition of VFs (CVF) in a point n from a list L of points as the sum of the forces between n and each point L_i in the list.

$$force(n) = \sum_{0 < i < |L|} force(n, L_i) \quad (3)$$

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 chooses its next hop on one of these omni-directional shortest paths which has the minimum force given a list L of point as the sources of the VFs. We define the force in a path P as the minimum force in the nodes on this path:

$$force(P) = \min_{0 < i < |P|} force(P_i, L) \quad (4)$$

A message will stop exploring the network and come back to the initiator when it reaches the maximum hop count or when it goes into a local minimum under the CVF. The past traffic of a message is then reported as a new VLL. Figure 2(a) is an example of the VLLs in node N. Where $MinHops = 2$ and the number of long links is 3. In this example, the VLLs of

node N in the random network is $NA (3, 4, 1)$, $NB (3, 7, 13)$ and $NC (3, 6, 8)$. We can see in the figure that the above algorithm can generate VLLs that lead to different areas of the network.

C. Evaluation of Virtual Long Links

We set an expiration time for each VLL considering that the continuously changing topology of the network may break some of the VLLs. Despite this, as a node periodically sends out VLL discovery messages, the required memory for the available VLLs can be larger than the limited storage in the node. Suppose the expiration time for a VLL is T_e and the time interval of sending consecutive VLL discovery messages is T_i , the maximum number of available VLLs is $k = T_e/T_i$. Roughly, if the storage limit is C_M VLLs and $C_M < k$, a node should discard $k - C_M$ less useful virtual long links.

Our replacement policy first lists all the possible combinations of C_M among k available VLLs and then calculates their usefulness. Only the VLLs in the set with the largest usefulness are retained. Our criterium of the usefulness of a set of VLLs is that the end points of the VLLs should be as far from each other and from the initiator as possible. The reason for this is that a node should have VLLs exploring different parts of the network in its vicinity (i.e., VLLs pointing to different directions) to make a forwarding decision for a message heading in any particular direction.

We found that entropy is suitable to measure the usefulness of a set of VLLs. Entropy is a measure of the internal microscopic disorder present in a system. Let say a set of points are in disorder if they are not close in position, we can use entropy to evaluate the level of position discrepancy of the points. Hence the larger the entropy, the larger the level of discrepancy in the points' positions and the more useful is the set of points in our criterium.

Suppose G is a *Gaussian window function*, d is the *Euclidean distance function*, the Renyi's entropy [20] of a set of points V is defined as follows:

$$G(a, b) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{d(a, b)}{2\sigma^2}} \quad (5)$$

$$Entropy(V) = -\ln \frac{\sum_{0 < i < |V|} \sum_{i < j < |V|} G(V_i, V_j)}{\frac{|V|(|V|-1)}{2}} \quad (6)$$

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

Extending the greedy protocol using VFs and VLLs is straightforward. The set of paths used in the new protocol contains the shortest path to all neighbor nodes and all VLLs. The destination is the source of a negative VF. The VF is defined as in Equation 2. The VF in a path from the destination is defined as the minimum VF of the nodes on the path similar to Equation 4. Thus the VF of a path equals the VF of the node in the path that is closest to the destination. The greedy protocol with virtual long links is shown below as Algorithm 1.

An example of this routing protocol is shown in Figure 2(b), where a message is sent from the source S to the destination

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.
-

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 .

The above protocol might cause a loop. Suppose there is a VLL (A, B, C) that is the best path for destination D when a message is in A , and there is also a VLL (B, A, E) that is the best path for destination D when a message is in B . When a message travels to A or B , a loop $(A, B, A \dots)$ will begin.

There is a simple solution to this problem, but with several bytes of additional transmission overhead: a message carries the path that it is currently being forwarded on and lets the next node on the path consider the path in its forwarding decision. We show how this method works still using the last example. Suppose C is closer than E to D . When a message is forwarded from A to B , it carries the path (B, C) . Thus when B makes a forwarding decision for the message, the path (B, C) will be chosen instead of (B, A, E) , and thus the loop is removed. In the following we will prove it formally.

Lemma 1: If a message m carries its current path, and m travels from node A to node B through a serial of path P_1, P_2, \dots, 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 .

Proof: Suppose $s_1, s_2, \dots, s_n, e_1, e_2, \dots, e_n$ are the starting points and the end points of the paths P_1, P_2, \dots, P_n respectively. We have $force(s_1, D) > force(e_1, D)$. And since in each s_i for all $2 \leq i \leq n$ the message chooses e_i instead of e_{i-1} , we have $force(e_i, D) < force(e_{i-1}, D)$. Therefore $force(A, D) = force(s_1, D) > force(e_1, D) > force(e_2, D) > \dots > force(e_n, D) = force(B, D)$. ■

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).

Proof: To prove *Theorem 1*, we need to prove that a message m will not travel to any node A infinite times. Suppose P_i is the path with minimum distance to destination D when m is in A for the i -th time, and e_i is the end point of P_i . According to Lemma 1, we have $force(e_1, D) > force(e_2, D) > \dots > force(e_n, D)$. Since the number of nodes in the network is finite, the number of e_i is finite. Therefore, m will not travel to A infinite times. ■

Since the VF based greedy protocol with VLLs is loop

free, the protocols produced by replacing the regular greedy algorithm in the greedy-face combinations guarantee delivery. We omit the proof here.

E. 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 routs 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 maintain 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 7. The other *forces* in this equation is defined in Equation 2 and Equation 3, where λ should be a large enough to recover the routing from local minima.

$$force(n) = \frac{\sum_{0 < i < |R|} force(n, R_i)}{|R|} - 2 \cdot force(n, D) \quad (7)$$

$$force(P) = \max_{0 < i < |P|} force(P_i) \quad (8)$$

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 failed in node M (which is a local minimum). The the message come 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

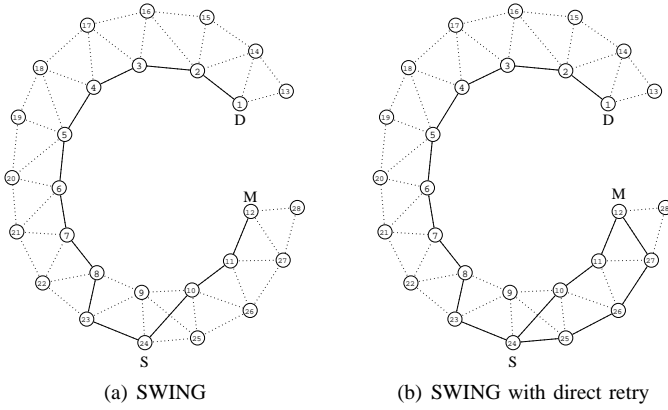


Fig. 3. Examples of routing with SWING and SWING with direct retry

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 $P = \{P_1 = source, P_2, \dots, P_n = destination\}$ 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 a 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 for a finite times T_k . If M does not go to P_{k+1} before it goes to P_k for finite times, there must be a path $L = \{P_k = L_1, L_2, \dots, L_n\}$ 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 ($K < T_1 \cdot T_2 \cdot \dots \cdot T_{k-1}$) is the size of repulsive list R in M , T_x ($T_x > 0$) is the times that M has went to L_n , we have:

$$\begin{aligned} force(P_{k+1}) &= force(P_{k+1}, destination) - \\ &\frac{1}{K} force(P_{k+1}, R) > -\frac{F_1}{K} \\ force(L_n) &= sm(L_n, destination) - \\ &\frac{1}{K} (force(L_n, (R \setminus L_n)) + T_x \cdot F_0) < F_1 - \frac{T_x \cdot F_0}{K} \\ force(P_{k+1}) - force(L_n) &> \frac{T_x \cdot F_0}{K} - \frac{(K+1)F_1}{K} \end{aligned}$$

Let $N = \frac{K+1}{T_x} < T_1 \cdot T_2 \cdot \dots \cdot T_{k-1}$. 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 . ■

F. 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 add to the local minimum at each iteration in SWING which make 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 continue 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 is added indeterministic. The route from S to D is this example is (24-10-11-12-27-26-25-24-23-8-7-6-5-4-3-2-1).

IV. SIMULATION

A. Evaluation assumptions and metrics

The objective our simulation is to measure and compare the performance of different geometric routing protocols. The 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 message sent. Route establishment delay is counted as the number of hops the message to discover a route traveled totally. The route length is length of 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 families, 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 [5], using virtual long link (VLL) in Greedy mode, using bound

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 ~ 450*
Network degree	4.71 ~ 14.13
Max routing hops count	(*)
Number of VLLs	0 ~ 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.

ellipse (BE) in face mode [11], the sooner back (SB) algorithm [5] (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 message is another node chosen randomly. We run each experiment 100 times to get the average value.

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

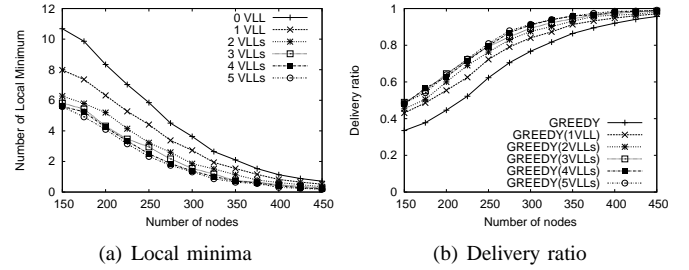


Fig. 4. Number of VLLs v.s. Average number of local minima and the delivery ratio of pure greedy routing.

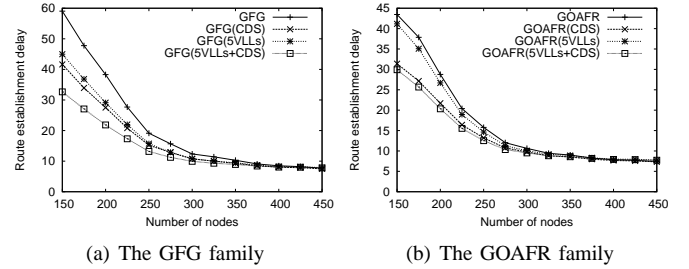


Fig. 5. Number of VLLs v.s. route establishment time of the GFG family and the GOAFR family.

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 increase. 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 term 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 $\sqrt{2}$ [11]. 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 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 need to generate 20,000 random graphs. Thus SWING with direct retry almost guarantees delivery in reality.

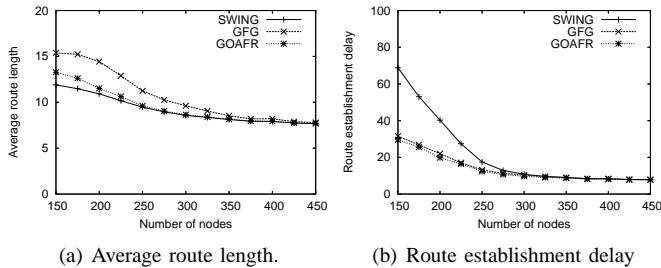


Fig. 6. Comparison of Average route length and Route establishment delay between SWING, GFG and GOAFR.

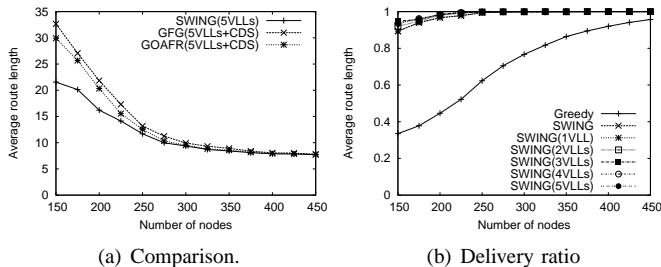


Fig. 7. Comparison of Route establishment time between SWING with direct retry, GFG and GOAFR. Delivery ratio of SWING with direct retry.

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 almost guarantees delivery except in extremely sparse density which seldom happens to real connected networks.

V. OVERHEAD & SCALABILITY ANALYSIS

In this section, we measure the scalability of SWING. For space limitation, we give the results directly. To gather omnidirectional link information, the communication overhead is $O(D)$ and the memory overhead is $(O(D^2))$. The amortized communication overhead for establishing VLLs per VLL message interval is $O(\text{MinHops} + 1)$. The amortized additional communication overhead of a routing message is the position information of the previous failures. To establish virtual long links, the computation overhead is $O(C_M^3)$ ($O(C_M)$ for the number of combinations and $O(C_M^2)$ for the entropy of each combination). Let C_M be the number of VLLs that can be stored in each node, the per-node memory overhead is $O(D^k) + O(C_M)$. The computation overhead for message forwarding is $O(D^k \cdot |R|)$.

VI. CONCLUSION

The paper has presented a research in position-based routing in MANETs. This paper solves the problem of the suboptimal 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 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.

REFERENCES

- [1] L. Blazevic, L. Buttyan, S. Capkun, S. Giordano, J.-P. Hubaux, and J.-Y. Le Boudec. Self-organization in mobile ad hoc networks: the approach of terminodes. *IEEE Communication Magazine*, page 166C175, June 2001.
- [2] 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.
- [3] 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.
- [4] 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.
- [5] 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.
- [6] K. Gabriel and M. Pearlman. A new statistical approach to geographic variation analysis. *Systematic Zoology*, 18:259–278, 1969.
- [7] D. Johnson and D. Maltz. Dynamic source routing in ad-hoc wireless networks. In *Proc. of ACM SIGCOMM*, 1996.
- [8] B. Karp and H.T. Kung. [gpsr]: greedy perimeter stateless routing for wireless networks. In *ACM MOBICOM*, 2000.
- [9] J. Kleinberg. The Small-world phenomenon: An algorithmic perspective. In *Proc. of the 32nd ACM Symposium on Theory of Computing*, 2000.
- [10] J. Kleinberg. Small-world phenomena and the dynamics of information. *Advances in Neural Information Processing Systems (NIPS) 14*, 2001.
- [11] F. Kuhn, R. Wattenhofer, and A. Zollinger. Worst-case optimal and average-case efficient geometric ad-hoc routing. In *ACM Mobihoc*, 2003.
- [12] J. Li, J. Jannotti, D. Decouto, D. Karger, and R. Morris. A scalable location service for geographic ad-hoc routing. In *ACM MobiCom*, 2002.
- [13] B. Liu, Z. Liu, and D. Towsley. On the capacity of hybrid wireless networks. In *IEEE INFOCOM*, 2003.
- [14] V. Mhatre and C. Rosenberg. Design guidelines for wireless sensor networks. *Communication, clustering, and aggregation. Ad Hoc Networks Journal, Elsevier Science*, 2:45–63, 2004.
- [15] S. Milgram. The small world problem. *Psychology Today 1*, 61 (1967).
- [16] M. Nesterenko and A. Vora. Void traversal for guaranteed delivery in geometric routing. In *Proc. of IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, 2005.
- [17] C. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. *Computer Communication Review*, 24:234–244, 1994.
- [18] 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.
- [19] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica. Geographic routing without location information. In *ACM MOBICOM*, 2003.
- [20] A. Renyi. On measures of entropy and information. In *Fourth Berkeley Symposium on Mathematical Statistics and Probability*, 1960.
- [21] 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.
- [22] I. Stojmenovic, M. Russell, and B. Vukojevic. Depth first search and location based localized routing and qos routing in wireless networks. In *IEEE International Conference on Parallel Processing*, August 21–24 2000.
- [23] G. Toussaint. The relative neighborhood graph of a finite planar set. *Pattern Recognition*, 12(4):261–268, 1980.
- [24] D. Watts and S. Strogatz. Collective dynamics of small-world networks. *Nature* 393, 440, 1998.