# Path Vector Face Routing: Geographic Routing with Local Face Information

Ben Leong, Sayan Mitra, and Barbara Liskov
MIT Computer Science and Artificial Intelligence Laboratory
32 Vassar Street
Cambridge, MA 02139, USA
{benleong, mitras, liskov}@csail.mit.edu

## Abstract

*Existing geographic routing algorithms depend on the planarization of the network connectivity graph for correctness, and the planarization process gives rise to a well-defined notion of "faces". In this paper, we demonstrate that we can improve routing performance by storing a small amount of local face information at each node. We present a protocol,* Path Vector Exchange (PVEX)*, that maintains local face information at each node efficiently, and a new geographic routing algorithm,* Greedy Path Vector Face Routing (GPVFR)*, that achieves better routing performance in terms of both path stretch and hop stretch than existing geographic routing algorithms by exploiting available local face information. Our simulations demonstrate that GPVFR/PVEX achieves significantly reduced path and hop stretch than Greedy Perimeter Stateless Routing (GPSR) and somewhat better performance than Greedy Other Adaptive Face Routing (GOAFR+) over a wide range of network topologies. The cost of this improved performance is a small amount of additional storage, and the bandwidth required for our algorithm is comparable to GPSR and GOAFR+ in quasi-static networks.*

## 1. Introduction

With the development of wireless ad hoc networks like roofnets [1, 30] and sensornets [16, 29], there has been a proliferation of geographic routing algorithms in recent years [3, 5, 7, 10, 19, 22, 24]. These algorithms are more scalable compared to traditional routing algorithms [15, 27, 28] because they require $O(1)$ storage per node, as opposed to $O(N)$ storage, where $N$ is the number of reachable destinations.

Most existing geographic routing algorithms first attempt to forward packets greedily; each node forwards packets to a neighboring node that is closest to the destination. When greedy forwarding causes a packet to be trapped in local minima, the packet is forwarded by traversing a face of a planarized graph (also known as *perimeter forwarding*). Existing geographic routing algorithms can guarantee packet delivery only if the network connectivity graph can be planarized without disconnecting the network. The distributed planarization of a real radio network was once a major challenge because traditional planarization algorithms relied on idealized radio models [18], but recent work showed that planarization can be achieved in practical radio networks [20].

As a node knows only about its immediate neighbors, there is often insufficient information for it to make a good decision on the forwarding direction when a packet gets trapped in a local minimum and has to switch to perimeter forwarding. To deal with this problem, a node usually has to resort to an arbitrary choice, e.g., use the *right-hand rule* as in Greedy Perimeter Stateless Routing (GPSR) [19]. But this choice may be the wrong one, and the penalty for making a wrong decision may be very high. Most schemes just choose deterministically [5, 19]; Greedy Other Adaptive Face Routing (GOAFR+) [22] deals with the problem by bounding the search in each direction within an expanding ellipse, thus avoiding the full consequences of a wrong choice.

This paper proposes a different way of tackling the problem, by using more information about the planar graph. Our expectation was that having more information would lower the routing cost and lead to an algorithm with better routing performance in terms of both path and hop stretch and we show experimentally via simulation that this is in fact the case. Our algorithm is efficient: it requires little bandwidth to propagate the extra information and only a constant amount of extra storage at each node. Yet it outperforms GPSR substantially, and even does better than GOAFR+, which to our knowledge is the most efficient geographic routing algorithm previously available. Like existing algorithms, we assume in our work that even though network connectivity may change, the planarized graph is quasi-static.

We have made both theoretical and practical contributions to the understanding of geographic routing. The theoretical contributions of our work are as follows:

- We show that there exists an oblivious (memoryless) algorithm, *Oblivious Path Vector Face Routing* (OPVFR), that can guarantee packet delivery for any planar graph if nodes have complete face information;
- Bose et al. had shown earlier that deterministic oblivious routing cannot guarantee packet delivery for arbitrary planar graphs where nodes are only aware of the positions of their one-hop neighbors [4]. We extend this result by showing the impossibility of oblivious routing in planar graphs where nodes are limited to knowing about nodes up to $k$-hops away, for any finite $k$.

Since some planar faces can be extremely large in practice, the assumption that nodes can maintain complete face information is sometimes impractical and we augment these theoretical findings with the following practical contributions:

- We present a practical asynchronous distributed algorithm, *Path Vector Exchange (PVEX)*, that propagates and maintains local face information efficiently as well as reacts to network membership changes;
- We propose a non-oblivious algorithm, *Greedy Path Vector Face Routing* (GPVFR), that guarantees packet delivery even when nodes do not have complete face information;
- Through extensive simulations we evaluate the performance of GPVFR and show that it achieves significantly better routing performance in terms of both path stretch and hop stretch than GPSR and somewhat better performance than GOAFR+ for random networks with only a small amount of additional routing state at each node.

We are especially interested in the performance of geographic routing algorithms for random networks with node densities in the critical range between 4 to 8 nodes per unit disk. Kuhn et al. had shown earlier that this region is critical for all geometric routing algorithms since existing geographic routing algorithms are uniformly good for both very sparse and very dense random networks [22]. The performance benefits for GPVFR over GOAFR+ are not huge, but we have found in our simulations that GPVFR does sometimes choose the wrong direction in spite of the extra information at the nodes, leading to very bad paths; this suggests that we could combine our approach with the search technique of GOAFR+ to obtain a technique that does substantially better.

The remainder of this paper is organized as follows: in Section 2, we provide a review of existing and related work. In Section 3, we describe our algorithms and present the theoretical foundations of our work. We present the results of our simulations in Section 4 and discuss the implications of our work in Section 5. Finally, we conclude in Section 6.

## 2. Related Work

The early proposals for geographic routing, suggested over a decade ago, were simple greedy forwarding schemes that did not have any guarantees of packet delivery in a connected network [7, 13, 32]. The first geographic (or geometric) routing algorithm to provide *guaranteed delivery* was Face Routing [21] (originally called Compass Routing II) .

Greedy Perimeter Stateless Routing (GPSR) [19] and several other algorithms [2, 5] were proposed subsequently. These also provided delivery guarantees and were somewhat more efficient in the average-case than Face Routing, though none of them outperformed Face Routing in the worst case.

Kuhn et al. first proposed an algorithm called *Adaptive Face Routing* (AFR) that bounds the cost of execution for geographic routing [23]. The key idea is to use an ellipse to restrict the searchable area during routing so that in the worst case, the total cost is no worse than a constant factor of the cost for the optimal route. Their algorithm also achieves the optimal worst-case result. In [24], Kuhn et al. studied the performance of a family of geographic routing algorithms that combined Greedy forwarding and AFR in different ways. Among them was Greedy Other Adaptive Face Routing (GOAFR), which was proved to be both worst-case and asymptotically optimal. A clustering technique was subsequently applied to GOAFR to produce GOAFR+ [22].

There have been previous geographic routing algorithms that advocate the maintenance of information about nodes in some locality that is beyond immediate neighbors: Stojmenovic and Lin had previously proposed extending existing geographic routing schemes to two-hop neighborhoods [31]. Geographic Location Service (GLS) is a distributed location service for the storage and lookup of geographic coordinates [26]. GLS also uses greedy forwarding and two-hop neighborhood information to reduce the chances of routing queries to dead-ends. The use of two-hop neighborhood information also turns out to have a side effect of improving average-case Euclidean path length.

We are aware of two related algorithms that maintain information on planar faces at each node: Mobicast [14] (which is a geocasting protocol, and not a point-to-point geographic routing protocol) and Greedy Perimeter Probing (GPP) [17]. These algorithms use probe packets to explicitly map the planar faces, while we piggyback information on the existing *keepalive* beacons and propagate the information through mutual exchanges between neighboring nodes. Also, these algorithms attempt to map faces completely, while we only maintain information about faces up to a fixed number of hops.

As for non-geographic ad hoc routing protocols, the Zone Routing Protocol (ZRP) [11] expands the amount of state stored at a node to a local neighborhood up to a fixed number of hops away. ZRP requires both a route discovery mechanism and query control protocol to work efficiently [12]. Other similar protocols include the limited-radius variant of DSDV [28] and a modified $k$-hop DSDV variant proposed by De Couto and Morris [6].

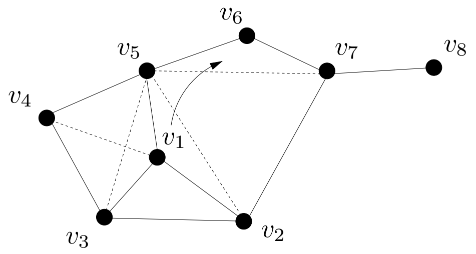

**Figure 1. An example planarized network to illustrate the path vector exchange protocol. Lines between two nodes indicate connectivity between them. The solid lines are the edges of the planar subgraph.**

## 3. Path Vector Face Routing

### 3.1. Preliminaries

In this work, we focus on the routing algorithm and assume that the underlying network graph can be and is planarized correctly [20]. Although our results apply to any planar graph, in our evaluation, we consider two well-known distributed planarization algorithms — *Gabriel Graph* (GG) [9], and *Relative Neighborhood Graph* (RNG) [34].

We consider a set of $N$ nodes in a bounded plane with identical broadcast range (normalized to one) and unique identifiers. These nodes induce a Unit Disk Graph (UDG) $G_u$, with the nodes as vertices and an edge between nodes $v_1$ and $v_2$ iff $\overline{v_1 v_2} \leq 1$. By applying a planarization algorithm (either GG or RNG) to $G_u$, we can obtain a connected planar subgraph $G$ that has no intersecting edges.

As mentioned in Section 1, the correctness of our algorithms depend on the usual assumption that the planarized graph $G$ is quasi-static, i.e., it does not change for the duration of a particular routing activity.

### 3.2. Path Vector Exchange Protocol (PVEX)

In this section, we describe the *Path Vector Exchange Protocol (PVEX)*, which is used by nodes to maintain local face information. Like other algorithms, nodes periodically broadcast beacons to inform neighboring nodes of their position and face information. On receiving beacons from its neighbors in the planarized graph, a node extracts the relevant information required to deduce the path vectors for its faces.

For example in Figure 1, the node $v_1$ is contained in and stores information about three faces that are *adjacent* to it in the planar subgraph — $\{v_1, v_5, v_6, v_7, v_2\}$, $\{v_1, v_3, v_4, v_5\}$ and $\{v_1, v_2, v_3\}$. We assign an ordering to the nodes on a face in the clockwise direction; for example, for the face $\{v_1, v_5, v_6, v_7, v_2\}$, we define *successor*($v_1$) = $v_5$ and *predecessor*($v_1$) = $v_2$. Every node will have a number of faces exactly equal to its degree (in the planar subgraph), though in some cases, some of the faces will be repeated. For example, node $v_7$ in Figure 1 has 3 faces, one is a pentagon ($\{v_1, v_5, v_6, v_7, v_2\}$) while the remaining two are the same face (the hull of the entire network).

More specifically, the state associated with PVEX at node $v$ has the following components:

- $\mathcal{F}_v$: An indexed set of the faces adjacent to $v$,
- $S_f$: list of successor nodes for face $f \in \mathcal{F}_v$,
- $P_f$: list of predecessor nodes for face $f \in \mathcal{F}_v$, and
- $seq_f$: sequence number for face $f \in \mathcal{F}_v$ (an integer).

In addition to the MAC address, node identifier, and its position information, the broadcast beacons contain this state, with $S_f$ and $P_f$ truncated to a maximum length $h$, which is a system parameter.

Node $v$ processes only messages that it receives from nodes with which it shares a face $f$ and where the received sequence number for $f$ is greater or equal to the local sequence number for $f$. When node $v$ receives $(u.S_f, u.seq_f)$ from successor node $u$, node $v$ updates its state as follows:

- If $v.seq_f \leq u.seq_f$ then $v.S_f := \{u, u.S_f\}$; and
- Additionally, if $v.seq_f < u.seq_f$ then $v.P_f := \{\}$ and $v.seq_f := u.seq_f$.

Likewise, when node $v$ receives $(u.P_f, u.seq_f)$ from predecessor node $u$, node $v$ updates its own state as follows:

- If $v.seq_f \leq u.seq_f$ then $v.P_f := \{u, u.P_f\}$; and
- Additionally, if $v.seq_f < u.seq_f$ then $v.S_f := \{\}$ and $v.seq_f := u.seq_f$.

The recorded sequence $seq_f$ for a face $f$ is incremented when (i) a node detects that its successor on $f$ has left the network[1]; or (ii) when a new node joins the network. In the latter case, the new node knows that it was not originally in the network, so it takes the existing sequence number for the face $f$ and increments it. Whenever a sequence number is incremented, it is incremented by a small but random amount so as to ensure that path vectors are refreshed correctly even when there are concurrent joins and departures. This protocol stabilizes locally for a node when the successor list intersects with the predecessor list and it has obtained the full information for a face, or when it has learnt about $h + 1$ nodes along both edges of the face. If $h = \infty$ or $2h > k - 1$, our protocol stabilizes for a face after $\lceil \frac{k}{2} \rceil$ rounds of message exchanges, where $k$ is the size of the face; otherwise, if $2h \leq k - 1$, it stabilizes after $h < \lceil \frac{k}{2} \rceil$ rounds.

A node will decide that the information for a face has stabilized as soon as the sequence number no longer increases for several inter-beacon intervals. If so, a node will only broadcast the sequence number for the face in subsequent rounds and not the associated path vectors. Thus, in the steady state, PVEX beacons are expected to contain only the node identifier, position and a small set of sequence numbers, and hence are only slightly larger than GPSR beacons. Nodes will resume broadcasting the path vectors for a face when there is an increase in the sequence number.

We jitter the inter-beacon transmission interval by up to 50% to avoid synchronization between the transmissions of neighboring nodes [8], so given a mean inter-beacon interval $T$, the actual inter-beacon transmission intervals are uniformly distributed in $[0.5T, 1.5T]$. One approach to achieve the stabilization of face information more quickly is to have shorter inter-beacon periods before stabilization and to have the mean inter-beacon period revert to $T$ only after a node is locally stable and no longer sends path vectors on its beacons.

### 3.3. Oblivious Routing with Complete Face Information

In this section we present *Oblivious PVFR*, a new geographic routing algorithm for planar graphs, where nodes have full face information. This information can be maintained by running PVEX with $h = \infty$. The general idea for this algorithm is as follows: when a node $v$ receives a packet $p$ for destination $t$, it finds the edge $e_v(t)$ among the set of all edges on its adjacent faces that has the minimum Euclidean distance to $t$. The minimum Euclidean distance of an edge $(x, y)$ to $t$ is the minimum distance over the set of all points in the line segment $\overline{xy}$ to $t$. If the segment $\overline{xy}$ contains the projection of $t$ onto it, the minimum distance will be the length of the perpendicular; otherwise it is $min(|xt|, |yt|)$. Let the node on $e_v(t)$ that is closest to $t$ be $n_v^*(t)$. Suppose $n_v^*(t)$ lies on face $f$. Node $v$ then forwards $p$ to the adjacent node on $f$ that is closest to $n_v^*(t)$ along face $f$. More formally,

**Algorithm 1 (OPVFR)** *Forwarding rule for node $v$ receiving a packet destined for node $t$:*

1. **Direct:** *If $t$ on face $f$ then forward the packet to* any *neighbor $v'$ in $f$ containing $v$ and $t$, such that $dist_f(v', t) < dist_f(v, t)$. Otherwise follow step 2.*
2. **Target node:** *Forward the packet to* any *neighbor $v'$ in the face $f$ containing $n_v^*(t)$, such that $dist_f(v', n_v^*(t)) < dist_f(v, n_v^*(t))$. If $v'$ does not exist, conclude that $p$ is undeliverable.*

where $dist_f(x, y)$ is the minimum distance between nodes $x$ and $y$ along a path on face $f$. The reason why we choose the target node in two steps instead of simply choosing the node in the set of all known nodes that is closest to the destination is because the latter may cause a packet to be trapped in a local minimum for some topologies.

An example of OPVFR routing from node $s$ to node $t$ is shown in Figure 2. The following theorem states the correctness of OPVFR.

**Theorem 1** *Given a connected pair of nodes $v$ and $t$ in a planar graph $G$, assuming that every node in $G$ completely knows all its faces, Oblivious PVFR guarantees packet delivery from $v$ to $t$.*

*Proof:* Let $<_v$ and $<_e$ be globally known operators that totally order the set of vertices and the set of edges, respectively[2]. Then, the *target edge* $e_v(t)$ of node $v$ with respect to the destination node $t$ is the $<_e$-smallest edge $\overline{xy}$, such that $x, y \in F(v)$ and $\overline{xy}$ has minimum Euclidean distance to $t$, where $F(v)$ is the set of vertices that share a face with $v$. Also, the *target node* $n_v^*(t)$ of node $v$ with respect to the destination $t$ is the $<_v$-smallest node $u$, such that $u$ has minimal distance to $t$ among the two vertices in $e_v(t)$.

Let $s$ and $t$ be two connected vertices in $G$, and consider the path $s, v_1, v_2, \ldots, v_n$ of a packet sent by $s$ to $t$. If there exists $v_i$ in the path such that $t \in F(v_i)$, then $v_i$ has a neighbor $v_{i+1}$ which shares a face with $v_i$ and $t$, and $dist_f(v_{i+1}, t) < dist_f(v_i, t)$. In all successive hops, the

[1] Since all nodes are expected to send periodic beacons, a node $v$ will assume that a neighboring node $u$ has failed, if $v$ does not hear from $u$ after a fixed interval that is $r$ times the inter-beacon interval, where $r$ is a system parameter.

[2] It is easy to construct such ordering operators. For vertices, for example, one could use the lexicographic ordering of the coordinates.

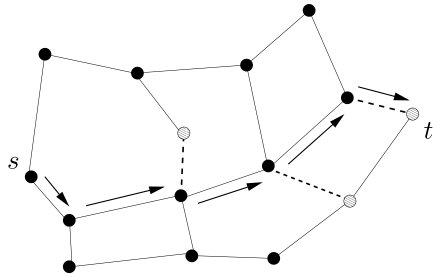

**Figure 2. An example of OPVFR routing from node $s$ to node $t$. The target edges are highlighted as dark dashed lines, while the target nodes are shaded in grey.**

packet is forwarded in the direct forwarding mode, and it gets closer to $t$. So, if there is any vertex in the path, that is in the same face as $t$, then the packet reaches $t$. The only way the packet from $s$ could not reach $t$ is if the path contained a cycle of nodes, such that none of the nodes in the cycle has a face adjacent to $t$.

First, we show that such a cycle cannot exist. For target $t$ we define the following function for every node $v$ in $G$, for which $t \notin F(v)$.

$$E_t(v) = \langle |n_v^*(t)t|, e_v(t), dist_f(v, n_v^*(t))\rangle$$

It is easy to see that for a given pair of nodes $s$ and $t$, this function has a finite minimum value. If we show that for all $i$, $E_{v_{i+1}}(t) < E_{v_i}(t)$, then we are done. Suppose this is not the case, then there are several possibilities:

**Case I:** $|n_{v_{i+1}}^*(t)t| > |n_{v_i}^*(t)t|$. As $v_i$ forwards the packet along an edge in the face containing the target edge, $v_{i+1}$ is in the same face as $v_i$. Therefore, $n_{v_{i+1}}^*(t)$ is at least as close to $t$ as $n_{v_i}^*(t)$, which contradicts the assumption.

**Case II:** $|n_{v_{i+1}}^*(t)t| = |n_{v_i}^*(t)t|$ and $e_{v_i}(t) <_e e_{v_{i+1}}(t)$. The distance from $t$ to the target points of $v_i$ and $v_{i+1}$ are the same. Since the target edges are not the same, $e_{v_{i+1}}(t) <_e e_{v_i}(t)$ because $v_i$ forwards the packet in the face containing $e_{v_i}(t)$, which contradicts the assumption.

**Case III:** $|n_{v_{i+1}}^*(t)t| = |n_{v_i}^*(t)t|$ and $e_{v_i}(t) = e_{v_{i+1}}(t)$ and $dist_f(v_i, n_{v_i}^*(t)) \leq dist_f(v_{i+1}, n_{v_{i+1}}^*(t))$. This is impossible since if $dist_f(v_i, n_{v_i}^*(t)) \leq dist_f(v_{i+1}, n_{v_{i+1}}^*(t))$, $v_i$ would not have forwarded the packet to $v_{i+1}$.

The only thing that remains to be shown is that there does not exist a node $v_n$ such that $n_{v_n}^*(t) = v_n \neq t$, $v_n$ does not share any face with $t$, and $v_n$ is connected to $t$. Suppose there exists such a node $v_n$. Consider the face $f$ adjacent to $v_n$ containing the two edges that the line segment $\overline{v_n t}$ subtends. $f$ cannot obscure $t$ from $v_n$ since the obscuring edge $e$ of this face would be the minimum edge for $v_n$. We need to consider two cases: (1) $t$ is located inside the face $f$, and (2) $t$ is located outside the face $f$. In both these cases, since $t$ and $v_n$ are connected, the planarity of the graph is contradicted.

■

### 3.4. Impossibility of Oblivious Routing with Incomplete Information

OPVFR relies on the availability of complete face information at each node. In a dynamic network, it is often impossible to maintain complete face information for every node accurately. Even in static networks, some faces can be very large ($O(N)$) and it might be impractical to propagate such large path vectors. Therefore it is desirable to limit the length of the path vectors propagated in PVEX to some constant $h$. This results in each node $v$ in $G$ having knowledge of all its faces, but up to depth $h+1$. The next theorem states that it is impossible to come up with a oblivious routing algorithm that guarantees packet delivery in all planar graphs, with such limited face information. This theorem generalizes Theorem 2 of [4].

**Theorem 2** *For any given non-negative integer h, there does not exist a deterministic oblivious routing algorithm that guarantees packet delivery for all planar graphs if nodes are limited to knowing only those nodes that are up to* $h+1$ *hops away.*

*Proof:* We construct a set of graphs, such that no oblivious algorithm can route correctly in all the graphs in the set, assuming that the nodes have complete face information only up to $h+1$ hops. In the graphs in Figure 3 every □ represents an identical chain of $h+1$ nodes, the other 16 nodes are located at the vertices of a regular 16-gon, and the destination node $t$ is located at its center.

Suppose, for the sake of contradiction, that there exists an oblivious algorithm $A$ that routes correctly in all graphs, where the nodes have correct face information up to $h+1$ hops. We claim that, according to algorithm $A$ all the gray nodes in all the 3 graphs in Figure 3 behave identically.

If not, then in graph 3(a) $v_i$ forwards CW and $v_{i+2}$ forwards CCW and the packet gets trapped in $\{v_i, v_{i+1}, v_{i+2}\}$, since there are no routes from the nodes in the □ nodes to $t$. Note that all the gray nodes have identical face information up to $h+1$ hops and are symmetric with respect to the location of $t$.

Let us assume that all the gray nodes forward packets CCW. Now, node $v_5$ cannot forward a packet towards node $x$ in graph 3(b), because then, the packet gets caught in the $\{x, v_1, v_{16}, v_1, v_2, \ldots, v_5\}$ circuit. From the point of view of node $v_5$ the graphs 3(b) and 3(c) are identical because it has the same face information up to $h+1$ hops. In graph 3(c), a packet from node $v_5$ would never enter the inner octagon, and therefore would never reach $t$. Similar

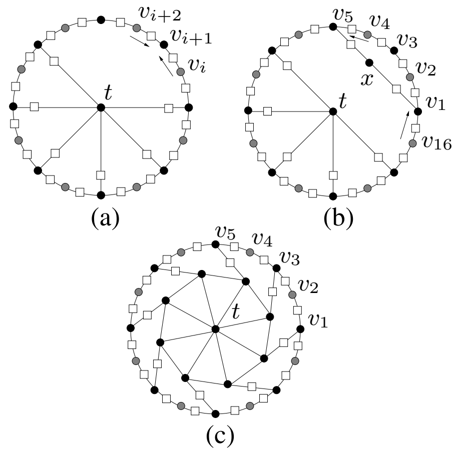

**Figure 3. Counter-examples showing non-existence of oblivious algorithm with limited face information.**

arguments can be made for the other black nodes by rotating the construction in 3(b) and 3(c). ■

Theorem 2 implies that the only way to achieve deterministic routing with incomplete face information is to resort to non-oblivious algorithms.

### 3.5. Non-Oblivious Routing

In this section we describe *Greedy PVFR*, a non-oblivious routing algorithm that does not require the participating nodes to have complete face information. GPVFR is designed as a tri-modal algorithm with the following modes:

- Greedy: greedy forwarding using neighbor information,
- OPVFR: greedy forwarding using face information, and
- Perimeter: perimeter traversal (as in GPSR).

Under GPVFR, packets are first routed in Greedy mode. When greedy forwarding to an immediate neighbor fails, a node may find that it knows of another node along its planar faces that is nearer to the destination than itself. Then, the node will apply the OPVFR algorithm to choose a target node, creating "virtual edges" for faces with incomplete information if necessary. Once a target node is chosen, it is recorded in the packet and the packet is switched to OPVFR mode and forwarded toward the target node. It is possible that this target node may be replaced by another if one that is nearer to the destination than the recorded target node is found while the packet is forwarded in OPVFR mode. OPVFR forwarding is like greedy forwarding except that nodes have a longer horizon, and packets are restricted to forwarding on the planar faces (edges). If there are multiple paths to the target node $n^*$, the choice can be made based on any performance metric that is monotonic along a path. In the example shown in Figure 4(a), node $s$ forwards a packet for $t$ to $k_1$ if we decide to minimize hop count and to $k_2$ if we choose to minimize the total Euclidean path distance.

Under both Greedy and OPVFR modes, a packet may end up at a node that does not know of any other nodes that is closer to the destination than itself, even including those along the planar faces. If so, we resort to Face Routing [21]. The node forwards the packet along the edges of the planar face that contains the imaginary line from the node to the destination. The choice of the direction to traverse the face is made based on the currently known set of path vectors instead of using an arbitrary *right-hand rule* like GPSR. In the example shown in Figure 4(b), a packet that is destined for $t$ is received by node $s$ that decides it has to forward the packet in Perimeter mode. Assuming limited-length path vector exchange and $s$ knows only nodes up to 3 hops away on any given face, $s$ forwards the packet for $t$ to $k_2$ since $|v_2t| < |v_1t|$.

We need to record the forwarding direction and mode in the packet, so that a node that receives the packet can determine the next edge of the face being traversed. In addition, face traversal can terminate under one of three possible circumstances: (i) a packet finds a face that is closer to the destination than the current face, (ii) a packet is undeliverable and ends up in a loop, or (iii) a packet finds a node that is closer to the destination than the node at which a packet first switched to Perimeter mode (which we call anchor node $s$). To detect these termination conditions, we record $s$ and the first edge traversed in the packet. We determine that a packet for destination $t$ has found a new face when it reaches a node that has an outgoing edge that intersects the imaginary line segment $\overline{st}$; if so, we set this node as the new anchor node and repeat the above process. It is necessary to record the first traversed edge because there are topologies for which a packet can end up back at the anchor node during perimeter traversal and yet not be in a loop.

When greedy forwarding works, it is usually the most efficient forwarding strategy, so we want to revert to greedy mode from OPVFR and Perimeter modes as soon as possible. In fact, we do so as soon as we find a neighboring node that is closer to the destination than the node recorded in the packet.

The following is a more detailed description of the GPVFR forwarding algorithm. A GPVFR packet $p$ is tagged with the following state components:

- $mode$: current forwarding mode, one of { Greedy, OPVFR, Perimeter},
- $s$: marker node (This is the target node in OPVFR mode and the anchor node in Perimeter mode),

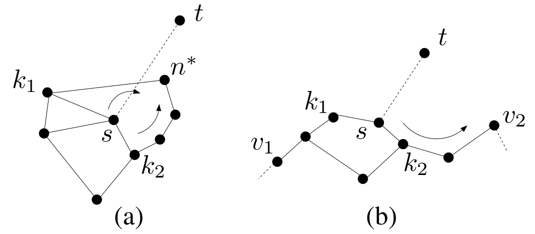

**Figure 4. Examples of path selection for GPVFR.**

- $e_f$: first edge traversed on the current face,
- $c$: forwarding direction (CW or CCW).

A node receiving a GPVFR packet will also record $n_p$, the node from which it received the packet.

**Algorithm 2 (Greedy PVFR)** *When a node $v$ receives packet $p$ for destination node $t$, do:*

1. **Check for switch to Greedy mode:** *If $p.mode$ = OPVFR or Perimeter and there is an immediate neighbor $u$ such that $|ut| < |(p.s)t|$, then set $p.mode$ := Greedy. Execute step 2, 3, or 4(b) respectively, according to $p.mode$.*
2. **Greedy:** *Find the node $u$ in the set of immediate neighbors that is closest to $t$. If $|ut| < |vt|$, forward the packet to $u$. Otherwise follow step 3.*
3. **OPVFR:** *Find the target node $n_v^*(t)$ (according to OPVFR). If $|n_v^*(t)t| < |vt|$ and $|n_v^*(t)t| < |(p.s)t|$, set $p.mode$ := OPVFR, and $p.s := n_v^*(t)$, and forward the packet along the shortest path towards $n_v^*(t)$. Otherwise follow step 4(a).*
4. **Perimeter:**
   - *(a)* **(Start)** *Find the incomplete face $f$ containing the line segment $vt$. Let $v_1$ and $v_2$ be the clockwise and anti-clockwise end nodes in the two path vectors for $f$. If $|v_1t| \leq |v_2t|$, forward the packet clockwise along $f$; if $|v_1t| > |v_2t|$, forward the packet anti-clockwise. Set $p.c$, $p.e_f$, $p.mode$ := Perimeter, $p.s := v$, and forward the packet accordingly.*
   - *(b)* **(Continue)** *If any adjacent edge intersects the line segment $\overline{(p.s)t}$ then set $p.s := v$ and go to step 4(a), else go to step 4(c).*
   - *(c)* **(End)** *Check if packet is undeliverable: If the next edge to be traversed along the current face $f$ is already recorded as $p.e_f$, conclude that the packet is undeliverable, else forward the packet to the next node along the current face $f$, as determined by $c$ and $n_p$, the node from which the packet was received.*

The correctness of this algorithm follows from the correctness of Face Routing [21]. The information required for OPVFR mode is propagated with PVEX as described in Section 3.2. The path vectors propagated are limited to a maximum length $h$, which is a parameter of the algorithm.

## 4. Evaluation

In this section we evaluate the performance of PVFR through simulation-based experiments. We compare the performance of both OPVFR and GPVFR to GPSR [19] and GOAFR+ [22] with respect to two commonly-used metrics: (i) *path stretch*, and (ii) *hop stretch*. Suppose $p_A$ is the path found by algorithm $A$ between a pair of nodes $s$, $t$. The path stretch of $p_A$ is the ratio of its Euclidean length to the shortest path (in Euclidean distance) between $s$ and $t$, and hop stretch is the ratio of the number of hops in $p_A$ to the number of hops in the shortest path (in terms of hops) between $s$ and $t$. Path stretch is a natural measure of goodness of any geographic routing algorithm, while hop stretch is of more practical interest because many radios transmit at a fixed power and smaller hop counts translate to shorter latencies and likely power savings. Apart from the above efficiency metrics, we also collect data on the amount of state propagated by the underlying path vector exchange protocol, while executing each of the algorithms.

To evaluate the various algorithms, we considered randomly generated unit disk graph networks for a range of network densities with nodes uniformly distributed on a $10 \times 10$ unit square using our own event-driven simulator [25]. For each density, we generated 100 networks and routed 16,000 packets on each network for each algorithm, and for both GG and RNG planarization. The source and destination nodes for each routed packet are chosen uniformly at random from the set of all nodes.

Our implementations of GPSR and GOAFR+ are based on the algorithms as described in [19] and [22] respectively. The configuration parameters for GOAFR+ are $\rho_0 = 1.4$, $\rho = \sqrt{2}$ and $\sigma = \frac{1}{100}$ as suggested in [22]; for GPVFR, unless stated otherwise, we limit the length of path vectors transmitted to 3 hops (i.e., $h = 3$). This means that each node will know about nodes up to 4 hops away on each adjacent face.

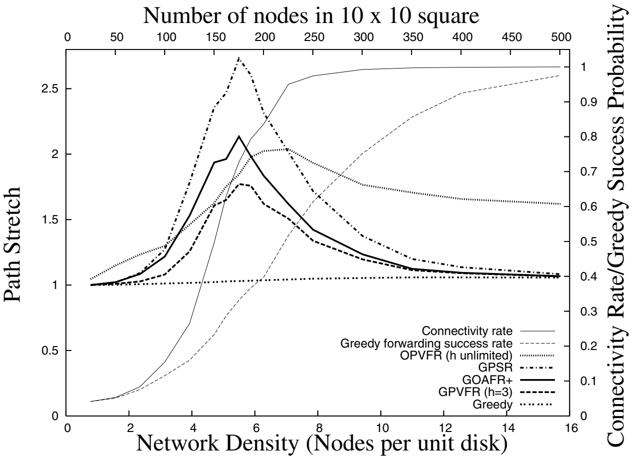

**Figure 5. Path stretch for OPVFR (unlimited $h$), GPSR, GOAFR+, GPVFR, and Greedy for UDGs of varying densities (under GG planarization).**

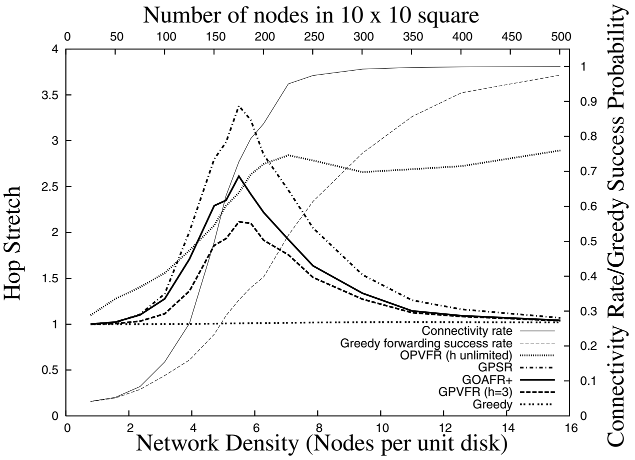

**Figure 6. Hop stretch for OPVFR (unlimited $h$), GPSR, GOAFR+, GPVFR, and Greedy for UDGs of varying densities (under GG planarization).**

## 4.1. Routing Performance

### 4.1.1 Comparison with Existing Protocols

Figures 5 and 6 show the average values of path stretch and hop stretch achieved by the different routing algorithms in randomly generated networks with increasing density under GG planarization. The performance of the simple greedy forwarding strategy and connectivity rate of the network are also plotted on the same graph for reference. The connectivity rate is the probability that two randomly-chosen nodes are connected, while the greedy forwarding success rate is the probability that greedy forwarding successfully delivers a packet between two randomly-chosen nodes.

We note that the simple greedy forwarding strategy outperforms all the others, and achieves a stretch of nearly 1 in almost all cases. Of course, one has to bear in mind that the greedy strategy does not always work. That is, a packet sent from $s$ to $t$ might get stuck at some intermediate node $v$ even when there exists a path from $s$ to $t$. As shown in Figures 5 and 6, this actually happens quite often. For example, at a density of 8 nodes per unit disk, Greedy fails approximately 40% of the time, even though any two randomly chosen nodes are 99% likely to be connected.

Our results show that GPVFR outperforms existing geographic routing algorithms, even GOAFR+ (which is known to be asymptotically optimal) in our experiments. It outperforms GPSR by a significant margin (i.e., reduces both the maximum path stretch and maximum hop stretch by about 37%) for networks in the critical region where node density is between 4 and 8 nodes per unit disk, and is better than GOAFR+ by a smaller margin (i.e., reduces both the maximum path stretch and maximum hop stretch by about 20%).

### 4.1.2 Effect of Planarization Algorithm

We also investigated the performance of the various routing algorithms for the same random networks under RNG planarization instead of GG planarization to understand how planarization affects routing performance. The corresponding results for RNG planarization are shown in Figures 7 and 8. As shown in the figures, all the routing algorithms seemed to perform slightly worse under RNG planarization, but the relative performance between the algorithms remained mostly unchanged.

### 4.1.3 Effect of Path Vector Horizon ($h$)

Since increasing the size of the exchanged path vectors (i.e., increasing $h$) increases the amount of knowledge that each node maintains about the topology of the network, it is natural to expect that increasing $h$ will improve routing performance. Figures 9 and 10 show that this is intuition is mostly correct, except that for denser networks, where GPVFR with full face information performs worse than GPVFR with $h$ limited to 5 or 6. The likely explanation for this is that OPVFR mode restricts packet forwarding to the edges on the planarized graph which often prevents more efficient direct forwarding that would be possible on the non-planarized edges. If the source and destination nodes both lie on the hull of the network and the hull is completely known (since $h = \infty$), packets will routed correctly, but albeit inefficiently along the hull. Also, these figures show even if $h = 1$, GPVFR performs better than GPSR, and that if we increase $h$ from 3 to 6, we increase the performance margin for GPVFR over GPSR and GOAFR+ by another 15%.

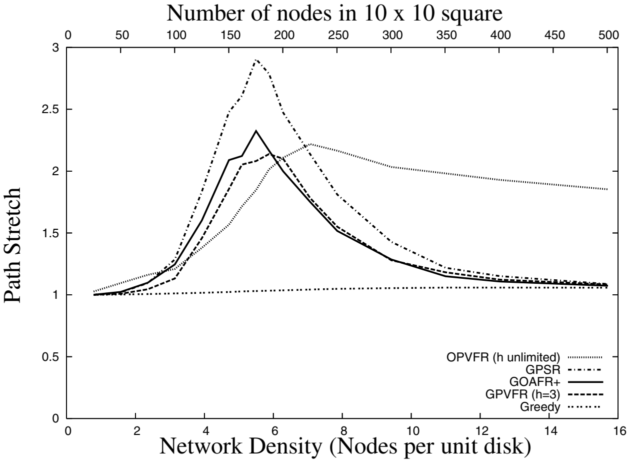

**Figure 7. Path stretch for OPVFR (unlimited $h$), GPSR, GOAFR+, GPVFR, and Greedy for UDGs of varying densities (under RNG planarization).**

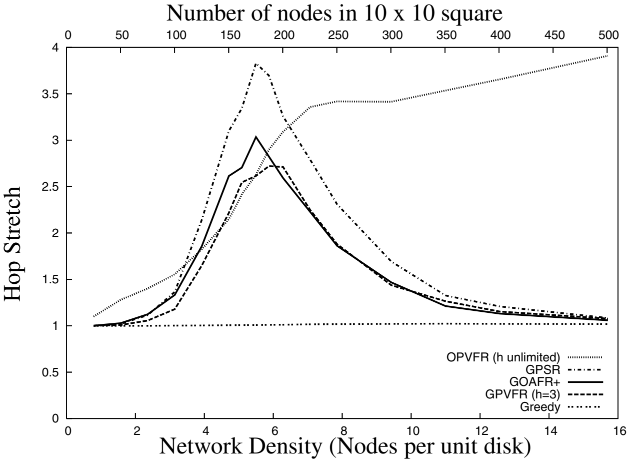

**Figure 8. Hop stretch for OPVFR (unlimited $h$), GPSR, GOAFR+, GPVFR, and Greedy for UDGs of varying densities (under RNG planarization).**

## 4.2. Memory and Bandwidth Requirements

Since nodes in GPVFR networks have access to more routing state than the corresponding nodes for GPSR and GOAFR+ networks, it is not surprising that GPVFR outperforms GPSR and GOAFR+. The important question is: what is the performance/state storage tradeoff for GPVFR, compared to GPSR and GOAFR+? Figure 11 shows the amount of state maintained by PVEX (size of face set $F(v)$) for random networks with GG planarization for $h = 3$. As shown in Figure 11, the face set size is less than 15 nodes for at least 95% of the nodes and this *upper limit is independent of the network density*, which is comparable to the number of immediate neighbors (neighbor set size), which increases with network density and is roughly equivalent to the amount of state stored per node by both GPSR and GOAFR+. Storing additional information about 14 nodes at each node is certainly not a concern for existing sensor devices like the Mica2 [33], which has 128K of program memory and 512K of flash RAM.

Another important question is whether GPVFR consumes more bandwidth than other geographic routing algorithms. In terms of the amount of state stored in the routing packet headers, GPVFR stores about the same amount of state as GPSR[3] and less state than GOAFR+. Since GPVFR has better hop stretch than GPSR and GOAFR+, it is likely that GPVFR will consume less routing bandwidth than both these algorithms when amortized over a large number of packets.

Next, we consider the beacons: the beacons for GPVFR may contain path vector information and can be significantly larger than those for GPSR. However, path vectors are only propagated when there are changes in the planarized subgraph (i.e. when nodes join, leave or move). Under normal operating conditions where the network is quasi-static, GPVFR beacons (or more accurately, PVEX beacons) are only slightly larger than GPSR beacons because they only contain a small additional set of sequence numbers in each beacon. Our empirical experiments suggest that the number of faces at each node is at most 6 for GG planarization, the size of the sequence number set is also less than 6, and thus we know that the PVEX beacons will contain no more than $\leq 8 \times 6 = 48$ nodes worth of information for $h = 3$. Since RNG is known to be a subgraph of GG [34], PVEX beacons are smaller when RNG is used as the underlying planarization algorithm.

## 4.3. Understanding GPVFR Performance

We store extra face information at each node with the intention of helping nodes make better decisions on the forwarding direction when they switch packets from greedy to perimeter forwarding mode. Hence, one interesting question is how often the extra information is useful in helping nodes make better choices. In our experiments with GPVFR (with $h = 3$) over the entire range of network densities, we found that on an average, any node is likely to know 80% of its adjacent faces completely and therefore it is able to choose the best path; where complete face information is not available, the probability of picking a good forwarding direction is about 60% on average, as opposed to 50% for the *right-hand rule*.

[3] More specifically, GPVFR stores two additional bits of information in the packet compared to GPSR because we allow for both clockwise and anti-clockwise face traversal and we have three forwarding modes instead of two. The cost of provisioning for these two extra bits is negligible.

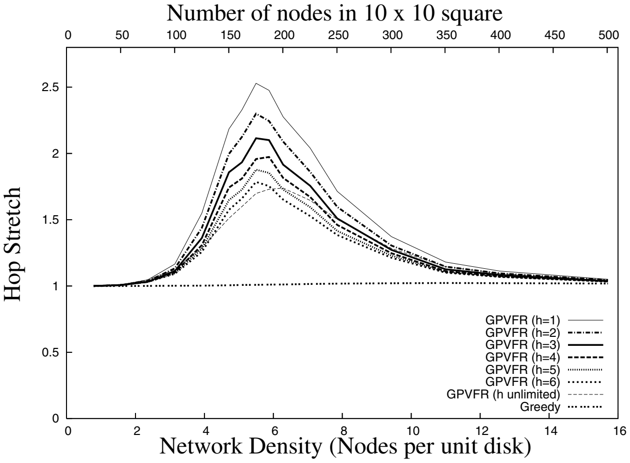

**Figure 9. The effect of varying $h$ on the hop stretch of GPVFR for UDGs of varying densities (under GG planarization).**

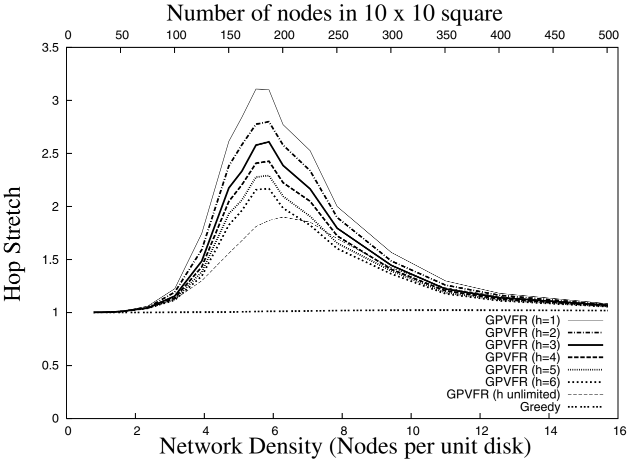

**Figure 10. The effect of varying $h$ on the hop stretch of GPVFR for UDGs of varying densities (under RNG planarization).**

As a side effect, local face information is also exploited by the OPVFR forwarding mode (which is equivalent to greedy-face forwarding) when regular greedy-neighbor forwarding fails. Hence, we also examined the distribution of the forwarding modes for GPVFR (with $h = 3$), GOAFR+ and GPSR. Figure 12 shows the proportion of hops that each packet will make in the various forwarding modes under GG planarization. The corresponding results for RNG planarization are similar. For the worst case in the critical region, about 65% of GPSR hops are made in greedy mode, which means that the remaining 35% of the hops are made in perimeter mode; in contrast, for GPVFR, some 83% of the hops are made in either greedy or OPVFR mode and only 17% are made in perimeter mode. Hence, our results suggest that a big win for GPVFR over GPSR is likely to come from being able to avoid perimeter forwarding mode most of the time. It was surprising for us that GOAFR+ packets actually spent a slightly smaller proportion of their time in greedy forwarding mode than corresponding GPSR packets.

## 5. Discussion

Our experiments suggest that geographic routing algorithms tend to perform better when the planarization has a higher density of connectivity, i.e., that GG planarization is preferred to RNG planarization. However, as highlighted by Karp and Kung [19], many MAC layers exhibit drastically reduced efficiency as the number of mutually reachable sending stations increases and using fewer links can help by reducing spatial diversity. It seems likely from our work that even though the underlying planarization algorithm affects the absolute efficiency of each routing algorithm, the relative efficiency across the range of routing algorithms is not dependent on the underlying planarization algorithm for randomized graphs. Although we have only studied the traditional GG and RNG planarizations, we believe that the relative performance of the studied algorithms will be similar for other planarizations even though there may be minor variations in the absolute performance. It would be interesting to evaluate the performance of these algorithms when a network is planarized with a different planarization algorithm like the recently-proposed Cross Link Detection Protocol (CLDP) [20] and also the effect of the MAC layer in practice.

As apparent from Figures 9 and 10, there are diminishing returns with increasing $h$ for both GG and RNG planarizations. However, increasing $h$ is more helpful for RNG planarization than GG planarization. This trend suggests a possible tradeoff: given a fixed bandwidth budget for PVEX maintenance, we can choose between GG planarization, or RNG planarization with a larger $h$.

Since GOAFR+ is the only asymptotically optimal algorithm among the algorithms we studied, it is possible to construct network topologies that will cause any of the remaining algorithms (including GPVFR) to perform arbitrarily badly. We note that GPVFR is compatible with AFR, i.e., GOAFR's technique of using an ellipse to bound the costs of a bad routing choices can be applied to GPVFR. It would be interesting to study a variant of GPVFR that incorporates bounded search. The fact that GOAFR+ packets spend more hops in perimeter forwarding mode than GPSR packets leads us to suspect that a new hybrid algorithm that combines GOAFR+ and GPVFR is likely to be able to further improve on GPVFR's current performance.

We have evaluated GPVFR in a mostly static scenario;

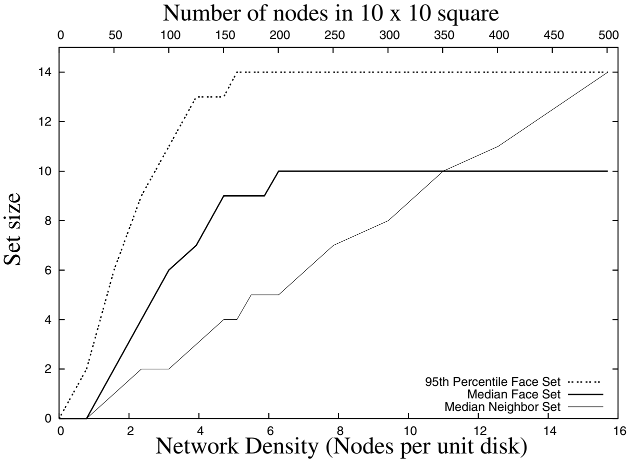

**Figure 11. Face and neighbor set sizes for random networks with nodes uniformly distributed in a square with 10 units side length (under GG planarization), when $h = 3$.**

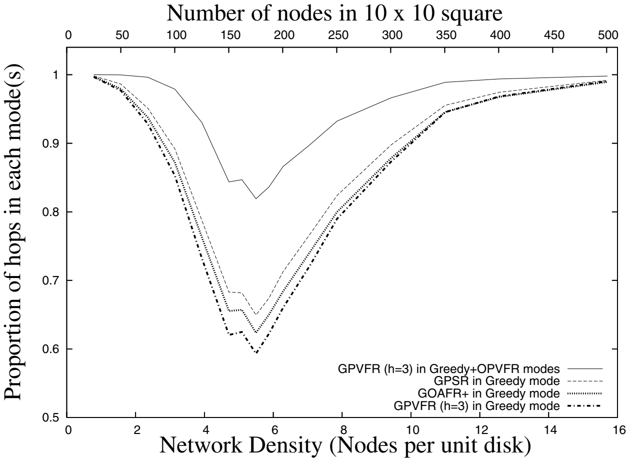

**Figure 12. Distribution of forwarding modes for GPVFR (with $h = 3$), GOAFR+ and GPSR.**

there are likely to be concerns that the bandwidth cost of maintaining face information may be prohibitive when there is rapid mobility. While we do have plans to evaluate the cost of GPVFR/PVEX in a fully mobile environment to address these concerns, we would like to highlight that we believe the bandwidth cost of PVEX is not likely to be a limiting factor for several reasons: First, if $h$ is limited to 3 hops, PVEX converges rapidly. Second, even though the problem of planarization for real radio networks has been solved, the current state-of-the-art planarization algorithm (CLDP) is significantly more complicated and will take longer to converge than PVEX [20]. If the planarization protocol converges, PVEX is almost certainly guaranteed to converge. In fact, if CLDP is employed as the planarization algorithm, nodes will learn about their adjacent faces during the planarization process and GPVFR can be applied without running PVEX. Finally, GPVFR can continue to work correctly when the planarization is correct even while face information is not yet consistent by simply skipping the OPVFR mode and switching directly from Greedy to Perimeter forwarding mode. To address bandwidth concerns or scarcity, GPVFR can also voluntarily trade off the routing performance with bandwidth consumption by reducing the length of the propagated path vectors. In the limit, if we set the length of the path vectors to zero, GPVFR is (almost) identical to GPSR.

## 6. Conclusion

In this paper, we demonstrate that by storing a small amount of local face information at each node, we can achieve better routing performance in terms of reduced path and hop stretch. The extra storage helps because the local face information can be exploited by a greedy-face forwarding mode using the available face information where regular greedy-neighbor forwarding fails to avoid switching to the costly perimeter forwarding mode. Also, where nodes have no choice but switch to perimeter forwarding, this extra information can marginally improve the probability of picking a good forwarding direction.

Our paper makes two main contributions: (i) we have shown that while it is possible to guarantee packet delivery with an oblivious algorithm in a network where nodes have full face information, it is impossible to do so when nodes are limited to knowing about nodes up to a fixed number of hops away on each face; and (ii) we developed Greedy Path Vector Face Routing (GPVFR), a non-oblivious algorithm that guarantees delivery even when nodes do not have complete face information. Through extensive simulations we have shown that GPVFR (with $h = 3$) achieves significantly better routing performance in terms of both path stretch and hop stretch than GPSR and somewhat better performance than GOAFR+ with only a small additional amount of routing state at each node.

## References

[1] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris. Link-level measurements from an 802.11b mesh network. In *Proceedings of ACM SIGCOMM Conference 2004*, August 2004.

[2] L. Barrière, P. Fraigniaud, and L. Narayanan. Robust position-based routing in wireless ad hoc networks with unstable transmission ranges. In *Proceedings of the 5th international workshop on Discrete algorithms and methods for mobile computing and communications*, pages 19–27. ACM Press, 2001.

[3] S. Basagni, I. Chlamtac, V. Syrotiuk, and B. Woodward. A distance routing effect algorithm for mobility (DREAM). In *Proceedings of the Fifth Annual International Conference on Mobile Computing and Networking (Mobicom 98)*, Dallas, Texas, October 1998.

[4] P. Bose, P. Morin, A. Brodnik, S. Carlsson, E. D. Demaine, R. Fleischer, J. I. Munro, and A. Lopez-Ortiz. Online routing in convex subdivisions. In *International Symposium on Algorithms and Computation*, pages 47–59, 2000.

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

[6] D. S. J. De Couto and R. Morris. Location proxies and intermediate node forwarding for practical geographic forwarding. Technical report, MIT, 2001.

[7] G. G. Finn. Routing and addressing problems in large metropolitan-scale internetworks. Technical Report ISI/RR-87-180, ISI, March 1987.

[8] S. Floyd and V. Jacoboson. The synchronization of periodic transmission messages. *IEEE/ACM Transactions on Networking*, 2(2):122–136, April 1994.

[9] K. Gabriel and R. Sokal. A new statistical approach to geographic variation analysis. *Systematic Zoology*, 18:259–278, 1969.

[10] J. Gao, L. Guibas, J. Hershberger, L. Zhang, and A. Zhu. Geometric spanner for routing in mobile networks. In *Proceedings of the 7th ACM International Conference on Mobile Computing and Networking (MobiCom '01)*, pages 45–55, July 2001.

[11] Z. J. Haas. A routing protocol for the reconfigurable wireless networks. In *IEEE International Conference on Universal Personal Communications (ICUPC'97)*, volume 2, pages 562–566, 1997.

[12] Z. J. Haas and M. R. Pearlman. The performance of query control schemes for the zone routing protocol. In *Proceedings of ACM SIGCOMM Conference 1998*, pages 167–177, August 1998.

[13] T. Hou and V. Li. Transmission range control in multihop packet radio networks. *IEEE Transactions on Communications*, 34(1):38–44, 1986.

[14] Q. Huang, C. Lu, and G.-C. Roman. Reliable mobicast via face-aware routing. In *Proceedings of IEEE Infocom '04*, March 2004.

[15] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353. Kluwer Academic Publishers, 1996.

[16] J. M. Kahn, R. H. Katz, and K. S. J. Pister. Next century challenges: Mobile networking for "smart dust". In *Proceedings of the International Conference on Mobile Computing and Networking (Mobicom '99)*, pages 271–278, August 1999.

[17] B. Karp. *Geographic Routing for Wireless Networks*. PhD thesis, 2000.

[18] B. Karp. Challenges in geographic routing: Sparse networks, obstacles, and traffic provisioning, May 2001.

[19] B. Karp and H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking (Mobicom 2000)*, pages 243–254, Boston, Massachusetts, August 2000.

[20] Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker. Geographic routing made practical. In *Proceedings of the 2nd Symposium on Networked Systems Design and Implementation (NSDI 2005)*, May 2005.

[21] E. Kranakis, H. Singh, and J. Urrutia. Compass routing on geometric networks. In *Proceedings of the 11th Canadian Conference on Computational Geometry*, pages 51–54, Vancouver, August 1999.

[22] F. Kuhn, R. Wattenhofer, Y. Zhang, and A. Zollinger. Geometric ad-hoc routing: Of theory and practice. In *Proceedings of the 22nd ACM Int. Symposium on the Principles of Distributed Computing (PODC)*, July 2003.

[23] F. Kuhn, R. Wattenhofer, and A. Zollinger. Asymptotically optimal geometric mobile ad-hoc routing. In *Proceedings of the of the 6th international workshop on Discrete algorithms and methods for mobile computing and communications (Dial-M)*, pages 24–33. ACM Press, September 2002.

[24] F. Kuhn, R. Wattenhofer, and A. Zollinger. Worst-Case Optimal and Average-Case Efficient Geometric Ad-Hoc Routing. In *Proceedings of the 4th ACM Int. Symposium on Mobile Ad-Hoc Networking and Computing (MobiHoc)*, June 2003.

[25] B. Leong. Geographic routing network simulator, 2004. http://web.mit.edu/benleong/www/netsim.

[26] J. Li, J. Jannotti, D. S. J. D. Couto, D. R. Karger, and R. Morris. A scalable location service for geographic ad hoc routing. In *Proceedings of the 6th annual international conference on Mobile computing and networking (MobiCom '00)*, pages 120–130, New York, NY, USA, 2000. ACM Press.

[27] C. Perkins. Ad-hoc on-demand distance vector routing. In *Proceedings of IEEE MILCOM '97*, November 1997.

[28] C. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *Proceedings of ACM SIGCOMM'94 Conference*, pages 234–244, August 1994.

[29] S. Ratnasamy, B. Karp, S. Shenker, D. Estrin, R. Govindan, L. Yin, and F. Yu. Data-centric storage in sensornets with ght, a geographic hash table. *Mobile Networks and Applications (MONET), Journal of Special Issues on Mobility of Systems, Users, Data, and Computing: Special Issue on Algorithmic Solutions for Wireless, Mobile, Ad Hoc and Sensor Networks*, 2003.

[30] T. J. Shepard. A channel access scheme for large dense packet radio networks. In *Proceedings of the ACM SIGCOMM '96 Conference*. ACM SIGCOMM, August 1996.

[31] I. Stojmenovic and X. Lin. Loop-free hybrid single-path/flooding routing algorithms with guaranteed delivery for wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 12(10):1023–1032, 2001.

[32] H. Takagi and L. Kleinrock. Optimal transmission ranges for randomly distributed packet radio terminals. *IEEE Transactions on Communications*, 32(3):246–257, 1984.

[33] C. Technlogies. Mica2 series wireless measurement system. http://www.xbow.com.

[34] G. T. Toussaint. The relative neighbourhood graph of a finite planar set. *Pattern Recognition*, 12:261–268, 1980.