Geographic Routing with Enhanced Local Information for Wireless Networks

Jung-Tsung Tsai, *Member, IEEE,* and Yunghsiang S. Han, *Fellow, IEEE*

Abstract

Geographic routing provides for network scalability through routing decisions made from local position information. This feature is essentially at the cost of routing path stretch particularly in networks having sparse nodes or holes. To mitigate the unappealing factor, we exploit information on the maximum angle between successive edges from a node, defined as the representative angle of a node that economically characterizes local topology and is also exchanged with neighbors. Besides analyzing its valuable information, we prove that nodes with the representative angle larger than or equal to $5\pi/3$, called corner nodes, can be logically removed from networks without disconnecting a given source from destination. We then propose routing schemes that thoroughly utilize representative angle information and associated design principles to reduce hop counts. In particular, we propose greedy network boundary traversal schemes which overlay the state of the art mechanism by the rotational sweep algorithm based on curved-stick to bypass network voids. Consequently, through excluding corner nodes as relays and taking into account the representative angle of a relay candidate by a weighting function in greedy mode operation and skipping unnecessary hops in recovery mode operation, our routing scheme achieves a remarkable saving of routing hop counts.

Index Terms

Geographic routing, greedy forwarding, network void, face routing, rotational sweep algorithm.

1 Introduction

GEOGRAPHIC routing [1] relies on position information to make routing decisions. It suffices that each node knows its own position and exchanges it with neighbors and that the source of a packet is aware of the packet destination position. Without the overheads of routing table maintenance or performing route discovery procedures, it is attractive particularly for wireless sensor networks and ad hoc networks.

The idea of geographic routing is to utilize information on neighbor positions and a packet destination to determine the next node to relay the packet. If there exists one neighbor closer to destination than the current node, in terms of Euclidian distance here, the one closest to destination is chosen. The method, called *greedy forwarding* (GF) [2], is repeated until destination is reached or such a neighbor does not exist. When the latter event occurs,

J.-T. Tsai is with the Department of Computer Science and Information Engineering, National Taiwan Normal University, Taipei, 10610 Taiwan (e-mail: jutsai@csie.ntnu.edu.tw).

Y. S. Han is with the Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, 10610 Taiwan (email: yshan@mail.ntust.edu.tw)

Part of this paper was presented in IEEE GlobeCom, Anaheim, CA, 3-7 Dec. 2012.

the packet is stuck at a node of so-called *local minimum*. This problem often arises in a network with sparse nodes, void regions, energy-loss nodes, or temporal topological holes due to node mobility.

Face routing on a *planar graph* [3] is a solution to the local minimum problem. Under the unit disk graph assumption that each edge is a symmetric perfect channel for a distance within a given range of radio coverage, reliable routing can be achieved through the joint operation that GF is employed whenever possible and when it fails, facing routing is invoked [4], [5], [6]. However, such a reliability and minimal information requirement is at the cost of path hop counts. Besides, the aggregate loading of relaying packets overall in a network increases. Both potentially raise end-to-end delay, an unappealing factor for time-sensitive traffic or saving node energy.

Instead of relying on face routing and well-known graph planarization techniques [7], [8], nongraph-based approaches have been proposed in [9], [10], [12], [13] to bypass network voids. As illustrated in Fig. 1(a), when GF fails, the BOUNDHOLE algorithm [9] uses a line of one transmission radius long to sweep a hole area according to the right-hand rule and chooses the first node hit by the line as the next relay. Unable to handle the issue of edge intersection, the method does not guarantee delivery and thus requires a remedy by flooding. The issue is resolved by Greedy Anti-void Routing (GAR) [10] where a rolling ball with a radius of half the radio transmission range is used instead. This method however increases routing hop counts. To further reduce visited nodes during traversing a convex boundary of a hole, a Curved Stick (CS) [13], a circular arc of the transmission radius, is used in place of the sweeping line to sweep a network hole. This scheme based on CS is exactly similar to the rotational sweep algorithm based on Twisting Triangle (TT) [12]. With a smaller curvature than the rolling ball and a chord length of the sweeping line, CS has been ingeniously designed to achieve both performance features of BOUNDHOLE and GAR. However, none of them would avoid selecting a relay on the concave boundary of a network void, such as node v_1 in Fig. 1(b) that can be practically skipped.

Apart from the position of a node, we exploit one extra item of local topology information for routing. It is the maximum angle of a partition of 2π formed by a node and all rays from the node to its neighbors. This angle, designated as *the representative angle* of a node, economically characterizes local topology and is exchanged with neighbors besides position information required normally. In essence, neighbor's representative angle information (RAI) provides partial information of the network topology bordering the node coverage area.

In this work, we focus on developing routing schemes capable of efficiently utilizing RAI to reduce routing path hop counts. We'll first resolve what topology information can be inferred from given RAI and then find how to correctly utilize the inference. This work has the following features:

- We prove that nodes with the representative angle larger than or equal to $5\pi/3$, called corner nodes, can be logically removed from a network graph without disconnecting a given source from destination.
- We analyze partial looking-ahead information from neighbor's RAI and utilize that information to design efficient greedy forwarding schemes. We also modify the perimeter, i.e. face, routing (PR) rule of greedy perimeter stateless routing (GPSR) [4] so that it walks on a planar graph with all corner nodes virtually trimmed off. The modified scheme significantly outperforms GPSR in terms of routing hop counts.
- We exploit RAI to design localized routing approaches overlaying the rotational sweep algorithm based on CS

[13] or TT [12] and create extremely efficient network boundary traversal schemes to bypass network voids. In essence, constrained by local available information our schemes trace the path of CS-sweeping and skip as many unnecessary hops as possible to locate the next relay. Consequently, the routing path hop count is remarkably reduced when they are combined with proposed GF variants particularly for networks having sparse nodes or voids.

1.1 Other related work

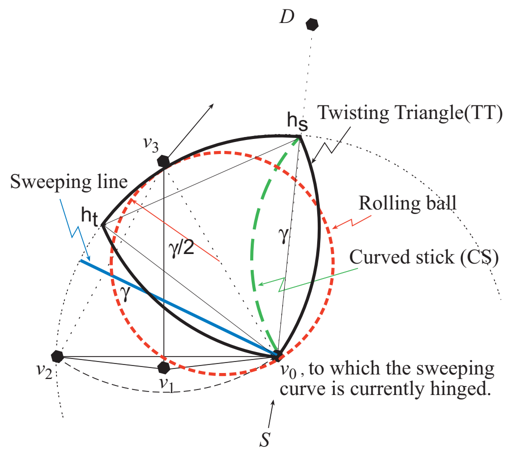
Early work of geographic routing was in the area of packet radio networks. In [1], [14], most forward progress routing (MFR) was considered, by which the neighbor with its projected position on the line from current node v to destination D closest to D is the next relay. The spirit of projection by MFR was viewed as an analytically convenient approximation of geographic greedy distance forwarding [15] when the distance $|vD|$ is sufficiently large. Cartesian routing in [2] considered routing based on geographic positions rather than network addresses in large scale internetworks.

In [3], compass routing II needs first to travel around an entire face boundary once to determine an edge intersected by the line from stuck node v to destination D and closest to D and then to travel the second time to a node of the edge where face routing proceeds to traverse the next face. As an improvement, FACE-2 in [5] employed the face traversal rule that routing changes to traverse the next face as soon as the first boundary edge intersected by vD is found. Greedy forwarding combined with face routing over a planar graph for guaranteed delivery was investigated in [4], [5]. To achieve a better bound of long routing paths, the GOAFR algorithm in [16] was shown to be asymptotically optimal in the worst-case scenarios through bounding searchable area adapted and restricted by an ellipse during face routing. GOAFR+ [16], [17] improved its average case efficiency by augmenting a counter technique for determining an instant of early fallback from face routing to GF.

To reduce path hop counts, one can resort to structuring a network area to contain subareas where efficient forwarding schemes may be stuck [3], [18] or subareas where greedy forwarding is allowed [19], [20], [21], [22]. Both require non-local information generally obtained at the price of message propagation. Low routing path stretch can be achieved if global structure information is used to assist inter-area routing [19], [20], [21].

In [23], location added routing (LAR) in MANET limits the search area of greedy flooding within a request zone during performing a route discovery. The request zone is defined from current node and destination locations and some margin around destination for uncertainty. Instead, the rectangular request zone was treated and built as an unsafe zone for some specific direction in [18] where routing for destination in the quadrant covering the direction should avoid choosing relay nodes in this unsafe zone.

Contention-based or beaconless routing [24], [25] did not require prior knowledge of neighborhood. The forwarder first broadcast, for example, a RTS message including local and destination position information. Each neighbor utilized the information in the message and its own position to determine an instant when contending for the relay by sending a CTS response. To recover from greedy failure, angular relaying algorithm [11] and rotational sweep algorithm [12] used angle-based contention by which each neighbor upon receiving a RTS message computed its



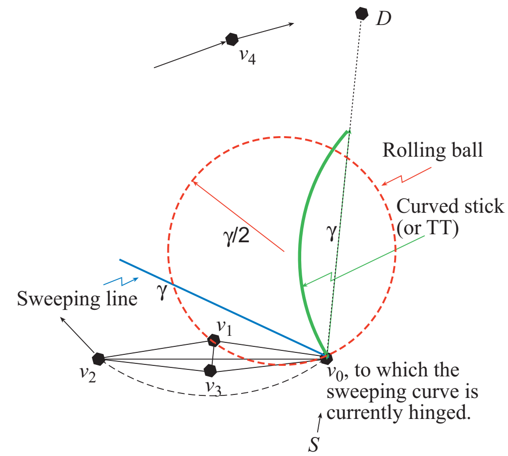


Fig. 1. Routing around network voids: (a) An edge intersection phenomenon arises when $|v_1v_3| < \gamma < |v_0v_3|$ and $\gamma < |v_2v_3|$. Node v_1 is on a convex boundary if $|v_0v_3| > \gamma$ and $|v_1v_3| > \gamma$. (b) Node v_1 is on a concave boundary.

contention delay proportional to the angle of rotating a given shape of sweep curve counterclockwise to hit it. This requires the previous hop position also included in the RTS message.

1.2 Outline

The rest of this paper is organized as follows. In Section 2 we describe the system model and basic components of geographic routing considered in this work. In Section 3 we present all design philosophies. Incorporating them, GF variants and a modified version of GPSR are presented. In Section 4, we propose greedy network boundary traversal schemes overlaying the CS-sweeping mechanism to bypass network voids. In Section 5 we show wireless network settings and present simulation results with discussions. We conclude in Section 6 with a brief discussion.

2 System model and elements

2.1 Networks and enhanced local information

Let the network graph of an ad hoc wireless or sensor network be denoted by $G(\mathcal{V}, E)$ in which $\mathcal{V}$ is the set of all nodes and E the set of edges. For each pair of nodes $u, v \in \mathcal{V}$, $u \neq v$, edge uv is included in E if and only if $|uv| \leq \gamma$ where γ is the radius of radio coverage and $|uv|$ the Euclidean distance between u and v. This follows the unit disk graph assumption. In other words, the wireless channel between each pair of nodes with distance less than or equal to γ is considered to be perfect and symmetric. This work doesn't involve issues arising from link impairments due to fading or node mobility.

Let $\mathcal{N}_v = \{v_i : |vv_i| \leq \gamma, v_i \in \mathcal{V}\}$ be the set of neighbors of node v, including node v itself. Here v_i is also used to represent its coordinate. Thus neighborhood set $\mathcal{N}_v$ furthermore represents the set of position information locally available to node v. It can be obtained by including position information in beacon messages as used in IEEE 802.11 provision for ad hoc networking. Besides, assume that there is a location service system through which the source S of a packet obtains the packet destination denoted by D. By including D in a packet, relay nodes will obtain it upon receipt of the packet.

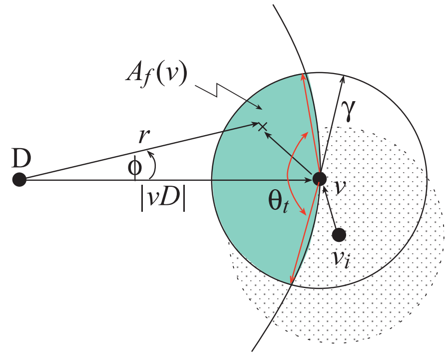


Fig. 2. A network graph shows the forwarding area of node v for destination D by $A_f(v)$, the SPN angle threshold from v to D by θ_t and the range of radio coverage γ.

Given $|\mathcal{N}_v| = n+1$ nodes, those n edges of vv_i partition a full angle 2π with vertex at v into n angles as denoted by θ_i, $i = 1, 2, \cdots, n$. Let $\theta_v^* = \max\{\theta_i : i = 1, 2, \cdots, n\}$. Also define $\theta_v^* = 2\pi$ for $|\mathcal{N}_v| = 2$ and $\theta_v^* = \infty$ for $|\mathcal{N}_v| = 1$ where node v is isolated. Here θ_v^* is the representative angle of node v. For routing schemes involving RAI, we assume that θ_v^* is also exchanged by beacon messaging. Thus a neighbor of v knows position v as well as representative angle θ_v^*. We will explore valuable information from θ_v^* in Section 3 and then utilize it for routing. For convenience we first define two terms:

- A corner node (CNN) is a node, say $v \in \mathcal{V}$, with representative angle $\theta_v^* \geq 5\pi/3$.
- A sure progress node (SPN) for position D is a node, say $v \in \mathcal{V}$, with representative angle $\theta_v^* < 2\cos^{-1}(\frac{\gamma}{2|vD|})$ and $|vD| > \gamma$.

As defined in [13], a **network hole**, or void, is a sequence of nodes $\{v_i, v_{i+1}, \cdots, v_{i+j}, v_i : v_{i+j} \neq v_i, j > 0\}$ so that the closed region bounded by this non self-intersecting polygonal sequence contains no node. A node is called a **boundary node** if it is located on the boundary of the network or of a hole inside the network. We define the boundary of a boundary node v_i being **locally concave** if interior angle $\angle v_{i-1}v_iv_{i+1} > \pi$ and **locally convex** if $\angle v_{i-1}v_iv_{i+1} \leq \pi$, where v_{i-1}, v_i and v_{i+1} are three successive boundary nodes. Accordingly, the boundary of node v_1 in Fig. 1-(a) is locally convex if $|v_1v_3| > \gamma$ and $|v_0v_3| > \gamma$ and in Fig. 1-(b) locally concave.

2.2 Function elements

Three fundamental routing schemes GF, PR, and TT are considered for further development. The latter two are each used for recovery from greedy failure. PR is referred to the face routing method on a relative neighborhood graph (RNG) [8] considered in GPSR, a renowned scheme. Because of its simplicity, *TT is here referred to the rotational sweep algorithm based on CS [13] instead of TT [12]* which is a Reuleaux triangle formed by the intersection of three circles placed on the corners of an equilateral triangle with side length of one transmission radius, as shown in Fig. 1(a). Scheme TT represents the state of the art technique based on local state information to bypass network voids.

2.2.1 Greedy forwarding (GF)

By GF, the forwarding area of node v to destination D for $|vD| > \gamma$, as shown in Fig. 2, is defined as

$$A_f(v) = \{x \in \Re^2 : |xv| \leq \gamma, |xD| < |vD|\}. \tag{1}$$

When generating or receiving a packet destined to D, node v_i based on information $\mathcal{N}_{v_i}$ searches a node in $A_f(v_i)$ closest to D as the next relay for the packet. That is, choose v^* , where

$$v^* = \arg \min_{v_j \in A_f(v_i)} |v_j D|. \tag{2}$$

If v^* is empty, v_i is a node of local minimum for D and referred to as a *stuck node*.

2.2.2 Perimeter routing (PR)

We follow the solution of face routing over a planar graph [4] with minor changes. Function components include the construction method of RNG, the righthand rule for traversing face boundaries, and an anchor v_A for switch-back test as well as for loop-back test. They are addressed in the following.

A graph without edge intersections is planar. RNG is a planar graph in which edge $vv_i \in E$ exists if

$$\begin{aligned} |vv_i| \leq \max\{|v_j v|, |v_j v_i|\} \qquad & \forall v_j \in \mathcal{V}, v_j \notin \{v, v_i\}, \\ & \text{and } v_i v_j, v v_j \in E. \end{aligned} \tag{3}$$

Under unit disk graphs, the search in (3) for witness node v_j between v and v_i is reduced to $\forall v_j \neq v_i \in \mathcal{N}_v$ and therefore, whether an edge from v exists in RNG can be determined locally.

Let v be a stuck node for D. By the righthand rule, rotate ray vD counterclockwise to reach and select the first node, say v_i, in $\mathcal{N}_v$ with edge vv_i in RNG as the next relay. At v_i, rotate ray $v_i v$ counterclockwise to reach and select the first node, say v_j, in $\mathcal{N}_{v_i}$ with edge $v_i v_j$ in RNG as the next relay. This rule recursively applies in traversing face boundaries.

Anchor v_A is the stuck node position where operation mode GF is switched to PR. In PR, v_A is carried in a packet until a node, say v_i, where the condition $|v_i D| < |v_A D|$ is satisfied and the operation is immediately switched from PR to GF. If v_i is determined to be another local minimum, the operation is then switched from GF to PR and v_i is the new anchor v_A to be carried in the packet. Upon such an occurrence, the operation may have performed a face change. On the other hand, if a packet from node v arrives at node $v_i = v_A$ and ray $v_A v$ is rotated counterclockwise to first reach the intersection of $v_A D$ and the circle of transmission range from v_A before any other node, then node v_A detects and stops the loop routing.

2.2.3 Scheme TT

The sweep curve of CS [13], or one side of TT [12], is an arc of a circle of radius γ with chord length γ, as shown in Fig 1. Let C_i be the circle of radius γ from center v_i that delimits the area within the transmission range of node v_i. When the arc is swung counterclockwise to complete one turn with one end of the arc hinged to node v_i and with its center of curvature trailing behind, the swept area is exactly that of C_i and each node in $\mathcal{N}_{v_i} - \{v_i\}$ is hit

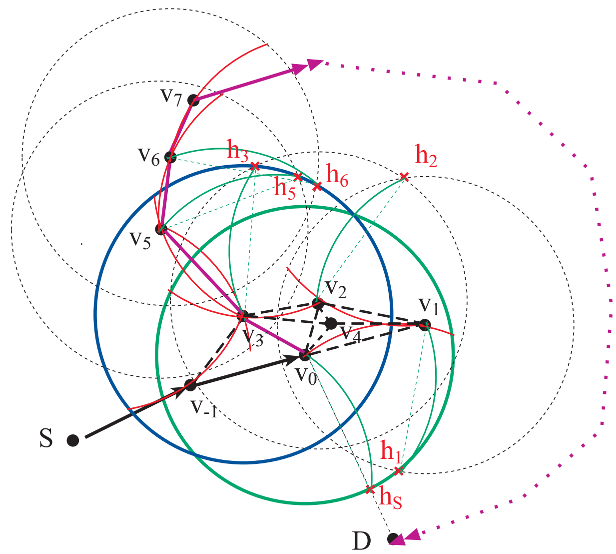


Fig. 3. Route a packet from S to D. By GFTT, it reaches v_{-1}, v_0 by GF, $v_1, v_2, v_3, v_5, v_6, v_7$ by TT, etc. By GFeTT, it reaches v_{-1}, v_0 by GF, v_3, v_5, v_6, v_7 by eTT, etc.

once. Such a sweeping is performed by v_i to locate the first hit node, denoted by v_{i+1}, as the next relay. Besides counterclockwise sweeping, five keys to correct operations are addressed in the following, with reference to Fig. 3.

1) At node v_i, position the sweep curve with both ends on (v_i, h_i) and swing it counterclockwise. Here h_i is the start point of sweeping located on C_i. If v_i is a stuck node to initialize CS sweeping, h_i is the intersection point of line segment v_iD and C_i which is also denoted as h_s; otherwise it is the intersection point of C_i and C_{i-1} on the righthand side of edge $v_{i-1}v_i$ where v_{i-1} is the upstream node that performs CS sweeping to hit and select node v_i. Note that shown in Fig. 1 (a), the start point of sweeping from a stuck node is h_s by CS in [13] and h_t by TT in [12], which is their main difference.
2) When the sweep curve is hinged to node v_i, it sweeps nodes in $\mathcal{N}_{v_i}$ located in the area of C_i.
3) If the sweep curve is hinged to v_i, the first hit node $v_{i+1} \in \mathcal{N}_{v_i}$ is selected as the next relay. In case that two or more nodes are first hit at the same time, the one with the largest distance to the CS-hinged node is chosen.
4) Switch to mode GF immediately when node v_i under recovery mode operation satisfies $|v_iD| < |v_AD|$ where v_A is the anchor (i.e., stuck) node.
5) A routing loop and hence destination-unreachable event is detected when the sweep curve is hinged to v_A again and swung to first hit h_s.

An example of joint GF and TT operation is illustrated in Fig. 3, where a packet destined to D is routed along path $S, v_{-1}, v_0, v_1, v_2, v_3, v_5, v_6, v_7$, etc. On the path, v_0 is a stuck node and the segment $v_1, v_2, v_3, v_5, v_6, v_7$ is constructed through a sequence of six CS sweepings, swung from (v_0, h_s) to hit next hinged node v_1, from (v_1, h_1) to v_2, from (v_2, h_2) to v_3, from (v_3, h_3) to v_5, from (v_5, h_5) to v_6, and from (v_6, h_6) to v_7.

2.3 Routing schemes and operations

In general, two routing methods with different purposes are employed and alternate in use for delivery guarantee. Fig. 4 shows the transition diagram of them for a packet from source S to destination D or to a node where D is

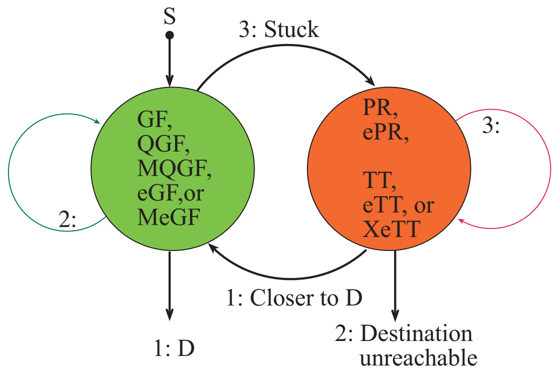


Fig. 4. Operation mode transition graph. The combination eGF+eTT, eGF+XeTT, QGF+eTT, or QGF+XeTT is not allowed.

found unreachable. In the figure, eGF, MeGF, QGF, and MQGF are modified from GF, ePR modified from PR, and eTT and XeTT modified from scheme TT, all by further involving RAI. Their designs will be presented in next two sections. We'll use xGF, xTT, or xPR to denote any one in each related class of methods, respectively.

In Fig. 4, a node upon receiving a packet invokes one of two available routing methods according to the operation *mode* carried in a packet header. When xGF is stuck, the operation mode is immediately switched to xTT or xPR. During mode xTT or xPR, a popular criterion for changing to xGF is the distance from current position to D shorter than from the stuck node. For the purpose, the stuck node position v_A is carried in a packet header after operation mode xGF is switched to xTT or xPR. Position v_A is also used in determining whether a routing loop or destination-unreachable event occurs.

Because of remarkable performance by xGF, the conventional wisdom is that the operation state should be kept in xGF as often as possible. In other words, the operation state should be changed from xPR or xTT to xGF as early as possible. However, this is no longer invariably true for our design of eTT and XeTT in Section 4.

3 Design principles, xGF and eGFePR

3.1 Characteristics and use of θ_v^*.

Assume that nodes are Poisson distributed in 2-dimensional space and that node $v \in \mathcal{V}$ has at least one neighbor, the upstream node from which it receives a packet. Let $|\mathcal{N}_v| = N + 1$, where $N - 1$ is the number of nodes other than v and its upstream node distributed in a circle area of $\pi\gamma^2$. Thus $\Pr\{N \geq 1\} = 1$ and representative angle $\theta_v^* = \max\{\theta_i; i = 1, 2, \cdots, N\}$ is a random variable. Presented in our conference version [26] the following proposition is given here for completeness.

Proposition 1: 1) If node v is a SPN for D, then it is not a node of local minimum for greedy forwarding.

2) $\Pr\{\theta_v^* > \phi | N = n\} \geq \Pr\{\theta_v^* > \phi | N = n + 1\}\ \forall \phi \in (0, 2\pi)$, $n \geq 1$.

Angle threshold $2\cos^{-1}\frac{\gamma}{2|vD|}$ used in defining SPN in Section 2.1 represents the angle formed by vertex v and the two intersection points of the circle of radius γ centered at v and that of radius $|vD|$ centered at D, as shown by θ_t in Fig. 2. If $\theta_v^* < 2\cos^{-1}\frac{\gamma}{2|vD|}$ holds, $A_f(v)$ cannot be empty of nodes. Statement 1) simply addresses this fact. Therefore a packet will reach destination if each visited node successfully finds a SPN for relay.

It is important to note that although varying with $|vD|$, angle threshold $2\cos^{-1}\frac{\gamma}{2|vD|}$ can be computed locally, for example, by current relay v_i in Fig. 2 which has already had position v and packet destination D. If node v_i has θ_v^*, it can determine whether v is a SPN for D. The angle threshold ranges between $2\pi/3$ and π since $2\cos^{-1}(\gamma/2|vD|) \to 2\pi/3$ if $\gamma/|vD| \to 1$ and $2\cos^{-1}(\gamma/2|vD|) \to \pi$ if $\gamma/|vD| \to 0$.

Let $\theta_{v|n}^*$ denote the size of θ_v^* conditioning on $N = n$. The expression of 2) shows that $\theta_{v|n}^*$ is stochastically larger than $\theta_{v|n+1}^*$ [27]. Intuitively, the higher the number of nodes uniformly distributed in the neighborhood of v, the smaller the representative angle θ_v^* stochastically. Obviously $\theta_{v|n}^*$ only takes values in $[2\pi/n, 2\pi]$, $n \geq 1$. Statement 2) reveals that to avoid the local minimum problem, preference should be given to a relay candidate with a smaller representative angle.

Consider that node v_i has representative angle θ_v^* of neighbor v. Let θ denote the angle from axis vv_i to the starting edge of an angular area delimited for θ_v^*. Assume that θ is uniformly distributed in $[0, 2\pi - \theta_v^*)$. Suppose that a node is randomly tossed according to the uniform distribution into the two angular areas of $(0, \theta]$ and $(\theta + \theta_v^*, 2\pi]$ within a circle of radius γ centered at v. Let ϕ denote the angle of the tossed node from polar axis vv_i.

Proposition 2: Conditioning on available θ_v^*, the tossed node is located in the angular area from angle ϕ to $\phi + d\phi$ with probability $f(\phi|\theta_v^*)d\phi$, where $f(\phi|\theta_v^*) =$

$$\begin{cases} \frac{2\pi-\theta_v^*-\phi}{(2\pi-\theta_v^*)^2}, & 0 < \phi \ \leq \min\{\theta_v^*, 2\pi - \theta_v^*\} \\ \frac{2(\pi-\theta_v^*)}{(2\pi-\theta_v^*)^2}, & \theta_v^* < \phi \leq 2\pi - \theta_v^* \\ 0, & 2\pi - \theta_v^* < \phi \leq \theta_v^* \\ \frac{\phi-\theta_v^*}{(2\pi-\theta_v^*)^2}, & \max\{2\pi - \theta_v^*, \theta_v^*\} < \phi \leq 2\pi\ . \end{cases} \tag{4}$$

Proof: See Appendix A. □

To see what can be inferred from available information θ_v^* and (4) by node v_i (see Fig. 2), we should focus on large θ_v^*. Since $f(\phi|\theta_v^*) = 0$ for $\phi \in (2\pi - \theta_v^*, \theta_v^*)$ and $\theta_v^* > \pi$, the angle range void of relay nodes is $2\theta_v^* - 2\pi$ increasing with $\theta_v^* > \pi$ and the void region is located around the direction of v_iv. In particular, the void region generally covers $A_f(v)$ entirely when $\theta_v^* > \pi$ is sufficiently large and the direction of v_iv is close to that of v_iD. On the other hand, if node v in $A_f(v_i)$ is far away from v_iD the condition for the void region to entirely cover $A_f(v)$ requires θ_v^* much larger than π. In such a case, preference is no longer given to node v by forwarding schemes. In short, under both situations node v has been a node of local minimum or can be ignored. This gives rise to classifying nodes into CNN and non-CNN by large threshold $5\pi/3$ and the following development.

Lemma 1: Let node v_i, with representative angle $\theta_{v_i}^*$, be a CNN in network graph G.

1) If $\theta_{v_i}^* = \infty$, v_i is an isolated node in G.
2) If $\theta_{v_i}^* = 2\pi$, v_i is a protruding node with only one neighbor, say v_j, in G. Besides, nodes v_i and v_j form an isolated network fragment in G if node v_j is also a CNN.
3) If $5\pi/3 \leq \theta_{v_i}^* < 2\pi$, node v_i has at least two neighbors in G, say v_j and v_k, which are in the neighborhood of each other. Moreover, nodes v_i, v_j and v_k are the vertices of an equilateral triangle network fragment in G if both v_j and v_k are also CNNs, and furthermore there is no network fragment of more than three successive CNNs.

Proof: Statement 1) is simply a fact of definition. To complete proof for 2) and 3), we need the following property. If $5\pi/3 \leq \theta^*_{v_i} < 2\pi$, v_i must have two neighbors v_j and v_k with rays v_iv_j and v_iv_k, $|v_iv_j| \leq \gamma$ and $|v_iv_k| \leq \gamma$, to form the two sides of angle $\theta^*_{v_i}$. Let $\theta = 2\pi - \theta^*_{v_i}$. Then,

$$\begin{aligned} |v_jv_k| &= \sqrt{|v_iv_j|^2 + |v_iv_k|^2 - 2|v_iv_j||v_iv_k|\cos(\theta)} \\ &\leq \sqrt{|v_iv_j|^2 + |v_iv_k|^2 - |v_iv_j||v_iv_k|} && (5) \\ &\leq \gamma && (6) \end{aligned}$$

implying that v_j and v_k are in the neighborhood of each other according to the unit disk graph assumption. Since all neighbors of v_i are within the angular area limited by v_iv_j and v_iv_k, they are in the neighborhood of each other.

The first half of statement 2) is also a fact of definition. To show the second half, suppose that v_j is a CNN with $5\pi/3 \leq \theta^*_{v_j} < 2\pi$. Then, there must exist a node v_k in the neighborhood of v_i by the above argument for (6). This however violates the hypothesis that v_i is a protruding node with $\theta^*_{v_i} = 2\pi$. Since v_j is not an isolated node of case 1), the only circumstance for v_j to be a CNN is $\theta^*_{v_j} = 2\pi$. Thus, v_j has only one neighbor v_i by the first half of 2) and therefore fragment $\{v_i, v_j, v_iv_j\}$ is isolated.

The previous argument for (6) has validated the first half of 3). Since v_j and v_k are neighbors, nodes v_i, v_j, and v_k form a triangle network fragment in which the explementary angle of each internal one adds up to 5π. If all of them are CNNs, $\theta^*_{v_i} = \theta^*_{v_j} = \theta^*_{v_k} = 5\pi/3$ is the only solution. Now suppose that v_i, v_j and v_k are all CNNs in a network fragment of more than three successive CNNs. Then there must exist a CNN, say v_ℓ, $\ell \neq i, j, k$, in the neighborhood of v_i, v_j or v_k. However, this is impossible since a) if v_ℓ is outside the triangle formed by v_i, v_j and v_k, then any of them neighboring v_ℓ is no longer a CNN and b) if v_ℓ is inside the triangle, it is in the neighborhood of every v_i, v_j and v_k and hence not a CNN. The proof of 3) is thus complete. □

Let $G^{(1)}$ be the resultant network subgraph after all CNN $v \in \mathcal{V} - \{S, D\}$ and all edges incident at those CNNs are removed from $G(\mathcal{V}, E)$.

Proposition 3: For any path from source S to destination D in G, there exists a path from S to D in $G^{(1)}$.

Proof: See Appendix B. □

After CNNs and their associated edges are removed from G, some of non-CNNs in G may turn into CNNs in $G^{(1)}$. Recompute θ^*_v for all v in $G^{(1)}$ and then remove all CNNs except S and D to obtain another trimmed subgraph $G^{(2)}$ from $G^{(1)}$. Repeat the procedure to shape a network graph and update representative angles until it is impossible to further trim a subgraph, denoted by $G^{(n)}$, where except S and D, no more CNNs and associated edges can be further removed. Applying Proposition 3 for $i = 2, \cdots n$, we have

Corollary 1: For any path from S to D in G there exists a path from S to D in $G^{(n)}$.

To recursively trim G and get $G^{(n)}$ in a distributed manner, it needs that each node in G computes its representative angle without taking into account the existence of any CNN, except S and D, and keeps updating the angle with neighbors. This demands disseminating state information extensively, which has been beyond the scope of locally available information considered in this work.

3.2 Forwarding schemes eGF, MeGF, QGF, and MQGF

3.2.1 eGF

Recall that by GF, node v_i utilizes position $v_j \in A_f(v_i)$ in cost function $|v_jD|$ in (2). When representative angle $\theta^*_{v_j}$ is also available, our approach first excludes CNNs from the set of relay candidates as suggested by Proposition 3 and adds to the cost function a linear and normalized extra cost increasing with $\theta^*_{v_j}$ as revealed by 2) of Proposition 1. Specifically, by eGF the way to exploit RAI is that for $|v_iD| > \gamma$, node v_i chooses next relay node v^* for a packet destined to D by

$$v^* = \arg \min_{v_j \in A_f(v_i) \wedge \theta^*_{v_j} < 5\pi/3} |v_jD| + \omega \frac{\theta^*_{v_j} - \theta_t(|v_jD|)}{5\pi/3 - \theta_t(|v_jD|)} \gamma \tag{7}$$

where $\omega \geq 0$ is an enhanced factor accounting for the impact of representative angles and $\theta_t(|v_jD|) = 2\cos^{-1}(\frac{\gamma}{2|v_jD|})$ the SPN angle threshold. If v^* is empty, the packet is stuck at node v_i.

Condition $\theta^*_{v_j} < 5\pi/3$ in the search domain of (7) is imposed to precisely realize Proposition 3. Note that $\frac{\theta^*_{v_j} - \theta_t(|v_jD|)}{5\pi/3 - \theta_t(|v_jD|)} < 1$ and that $\frac{\theta^*_{v_j} - \theta_t(|v_jD|)}{5\pi/3 - \theta_t(|v_jD|)} < 0$ for $\theta^*_{v_j} < \theta_t(|v_jD|)$. This extra weight actually decrements the cost of routing over SPNs for D. Also note that even by setting $\omega = 0$ in (7), the scheme is no longer GF since CNNs are excluded.

3.2.2 MeGF

When a packet is stuck at node v_i by eGF, there may still exist some CNN $v_j \in A_f(v_i)$ with $|v_jD| < |v_iD|$. Thus v_i may not be on a network boundary. However, it is imperative for scheme eTT or XeTT, to be presented in Section 4, to start at a network boundary node in order to ensure correct operations. For the purpose, eGF is modified to perform one more search for a relay by GF according to (2) when v^* by (7) is empty.

Algorithm 1 shows a C-like MeGF where line 1 checks whether current position is D, line 3 checks whether destination D is in transmission range, line 6 performs eGF to find the next relay, if none has been found line 9 tries GF to find one, if there is one then send the packet to the next relay in line 12, and otherwise switch to a chosen network boundary traversal scheme in line 17. When lines 8–10 are removed, the algorithm is reduced to eGF. Lines 14–16 prepare parameters required for operation in recovery mode xTT. If xPR is chosen to escape from a stuck node, lines 16 is replaced by mode $\leftarrow$ {PR or ePR}.

3.2.3 QGF and MQGF

The adaptive quasi-greedy forwarding scheme in [26] simply divides neighbors in forwarding area $A_f(v_i)$ into two subsets. It determines a relay by GF first from the subset consisting of SPN neighbors and then from the second consisting of the rest otherwise. Now, with Proposition 3 we further divide the second subset by removing CNNs from it and collecting them into the third one and likewise exclude the third subset from relay candidates. We call this approach QGF for short. Like MeGF, MQGF tries a GF step by (2) attempting to route a packet to a CNN in the third subset when QGF fails. QGF or MQGF doesn't require enhance factor ω used in (7) but loses some performance. This will be shown in Section 5.

Algorithm 1: Modified Enhanced GF (MeGF)

Require: A packet with mode *MeGF* for D at node v_i.
{If v_i originates a packet for D, it starts with mode *MeGF*.}
Ensure: Send node v_{i+1} the packet with mode *MeGF* or switch to {TT, eTT, XeTT}.

if v_i is D **then**
  STOP
else if $|v_i D| \leq \gamma$ **then**
  $v_{i+1} \longleftarrow D$.
else
  Find node $v_{i+1} \longleftarrow v^*$ by (7).
end if
if v_{i+1} is empty **then**
  Find node $v_{i+1} \longleftarrow v^*$ by (2).
end if
if v_{i+1} is not empty **then**
  Send the packet to v_{i+1}. {Remain in MeGF mode}
else
  $v_A \longleftarrow v_i$
  Compute h_s
  mode $\longleftarrow$ {TT, eTT or XeTT}
  Switch to operation mode
end if

3.3 Routing scheme eGFePR

With RAI, ePR follows PR described in Section 2.2.2 while realizing Proposition 3 with the constraint that each relay searched and chosen by the righthand rule in RNG should be a non-CNN. That is, if a relay candidate chosen by PR is a CNN in RNG, skip this one and try the right hand rule to find the next one. This procedure is repeated until a non-CNN in RNG is selected. In effect, ePR walks on a virtually trimmed RNG.

Combined for routing, both eGF and ePR do not select any CNN as a relay. The correctness of eGFePR completely follows that of GPSR (i.e. GFPR) except that a remedy is required to correct a possible flaw that may occur at source S. The flaw and remedy are addressed next.

Suppose that S is in a connected network subgraph $G_s \subset G$, D not in G_s, and $|v_i D| > |SD| > \gamma$ for all $v_i \in G_s, v_i \neq S$. Then, S is a stuck node and the one in G_s closest to D. Suppose furthermore that S is a CNN and the next node chosen according to the righthand rule is v_B. By PR, S is the anchor node carried in a packet. When the constraint of selecting non-CNNs in RNG is imposed by ePR, v_B, if not empty, is definitely not a CNN. After sending to v_B a packet for D, S won't be selected as a relay thereafter. An infinite loop thus occurs since the

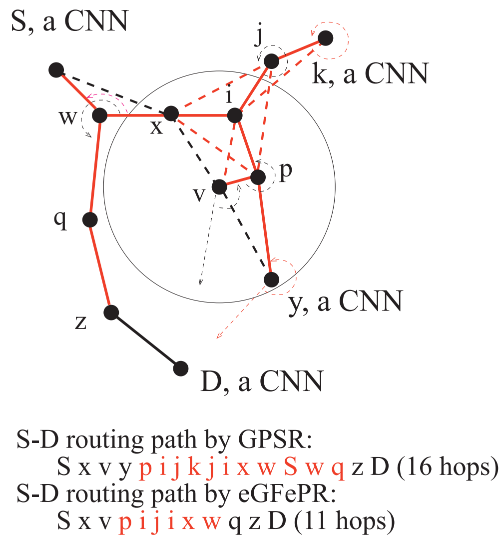


Fig. 5. The routes from S to D that are chosen by GPSR (i.e. GFPR) and by eGFePR with $\omega = 0$, respectively. Dashed links do not exist in RNG. GPSR switches to mode PR at node y and takes 16 hops totally to reach D. eGFePR switches to ePR at node v and only takes 11 hops to reach D by omitting CNNs y, k and S.

traversal over trimmed RNG will never return to S where a routing loop can be detected.

We fix the above problem by using node v_B as the anchor node instead as if v_B started an ePR session. The way is that node S in ePR mode leaves the anchor node field of a packet header empty if it is a CNN and sends it to node v_B; When node v_B receives a packet in ePR mode with empty anchor node, it sets its own position as the anchor.

Similarly the previous flaw arises when QGF is combined with ePR and possibly at many stuck nodes when GF, MeGF, or MQGF is combined with ePR for routing. The above remedy can be still applied.

The example shown in Fig. 5 illustrates two routing paths separately created by GPSR and by eGFePR with $\omega = 0$. The network graph consists of 12 nodes and 19 edges. Among them, nodes S, D, y and k are CNNs and eight dashed links do not exist in RNG. GPSR starts with GF, switches to PR at node y and returns to GF at node z, totally taking 16 hops from S to D. Scheme eGFePR starts with eGF, switches to ePR at node v, omits CNNs y, k and S during the course of ePR and returns to eGF at node q, for a total of only 11 hops.

4 GREEDY NETWORK BOUNDARY TRAVERSAL

Every node first hit by the sweep curve of CS [12], [13] is definitely selected as the next relay and hence the next CS hinged node. With RAI available we'll here develop two network boundary traversal schemes that may perform a sequence of sweepings and then select as the next relay the CS hinged node that performs the last sweeping in the sequence. This decouples the above relationship that every CS hinged node is also a selected relay. The spirit can be illustrated by the example shown in Fig. 3 where a packet destined to D is stuck at node v_0, a sequence of sweepings by the sweep curve successively hinged to v_0, v_1, v_2, v_3 is performed to sweep nodes in $\mathcal{N}_{v_0}$ but located in C_0, C_1, C_2, C_3, respectively (i.e. in $\mathcal{N}_{v_0}, \mathcal{N}_{v_1}, \mathcal{N}_{v_2}, \mathcal{N}_{v_3}$, respectively, but located in C_0), and the packet is then sent directly to node v_3 after the sequence of sweepings that lastly hits node v_{-1}. Two hops through nodes v_1 and v_2 required by scheme TT are now skipped over through routing computations at node v_0 from locally available information. Besides the fixed sweeping area C_0 restricted by local information for such a sequence of sweepings, a key to correct operations is determining where to stop and what information to be carried with the packet to the stop node where a new session of successive sweepings may be invoked.

In Section 4.1 we'll establish the condition, in Proposition 4, under which the hop to a concave node, like v_1 in Fig. 1-(b), is unnecessary. In Section 4.2 we'll establish the condition, in Proposition 5, under which even the hop to a convex node, like v_1 in Fig. 1-(a) and $|v_1v_3| > \gamma$, is unnecessary.

4.1 Enhanced TT (eTT) scheme

Let v_i in $\mathcal{N}_{v_{i-1}}$(or C_{i-1}) and v_{i+1} in $\mathcal{N}_{v_i}$ (or C_i) be two successive nodes hit and selected by the CS hinged to v_{i-1} and v_i and swung from start points h_{i-1} and h_i, respectively. Define $\phi_{tt}(v_i) = \angle v_{i-1}v_iv_{i+1}$. It has been shown in Property 4 [13] that any interior angle in the polygram generated by the relay nodes chosen by sweeping is larger than $\pi/3$. That is,

Lemma 2:

$$\phi_{tt}(v_i) > \pi/3. \tag{8}$$

Instead, let v_{i+1} be the first node hit by CS hinged to v_i and swung from start point h_i to sweep nodes in $\mathcal{N}_{v_{i-1}} \cap \mathcal{N}_{v_i}$ (or $C_{i-1} \cap C_i$). For such a modification of sweeping area, define $\phi_{gtt}(v_i) = \angle v_{i-1}v_iv_{i+1}$ to differentiate it from the previous one. Then,

Lemma 3: Suppose that $\theta^*_{v_i} > \pi$.

1) If $\phi_{tt}(v_i) > \pi$, then $\theta^*_{v_i} \leq \phi_{tt}(v_i) < \theta^*_{v_i} + \pi/6$.

2) If $\phi_{gtt}(v_i) > \pi$, then $\phi_{gtt}(v_i) \geq \theta^*_{v_i}$.

Proof: To verify 1), let $\phi_{tt}(v_i) > \pi$. Since $2\pi - \phi_{tt}(v_i) < \pi < \theta^*_{v_i}$, representative angle $\theta^*_{v_i}$ can not occur in the explementary angle of $\phi_{tt}(v_i)$ at node v_i. Thus, $\theta^*_{v_i} \leq \phi_{tt}(v_i)$ and $\theta^*_{v_i}$ is entirely in or on $\phi_{tt}(v_i)$. Let nodes $u_{i+1} \in \mathcal{N}_{v_i}$ and v_i with $|v_iu_{i+1}| \leq \gamma$ form one side of the representative angle $\angle v_{i-1}v_iu_{i+1} = \theta^*_{v_i}$. Also let z be the point on C_i with $|v_iz| = \gamma$ and $\angle v_{i-1}v_iz = \theta^*_{v_i}$. It is clear that u_{i+1} is on the line segment v_iz. If u_{i+1} is first hit by CS swung from (v_i, h_i) for nodes in $\mathcal{N}_{v_i}$, then u_{i+1} and v_{i+1} exactly refer to the same node and $\phi_{tt}(v_i) = \theta^*_{v_i}$. On the other hand, if $u_{i+1} \neq v_{i+1}$ then $\phi_{tt}(v_i) > \theta^*_{v_i}$ and the maximum of $\phi_{tt}(v_i)$ occurs when $u_{i+1} = z$ and the sweep curve hinged at v_i is swung to first hit v_{i+1} located on the tangent of the curve at v_i and near v_i immediately before hitting u_{i+1}. Under this circumstance, $\angle u_{i+1}v_iv_{i+1} = \pi/6$. Thus $\phi_{tt}(v_i) < \theta^*_{v_i} + \pi/6$. By similar arguments, $\theta^*_{v_i} > \pi$ can not occur in the explementary angle of $\phi_{gtt}(v_i)$ if $\phi_{gtt}(v_i) > \pi$ and thus Part 2 follows. □

Lemma 4:

$$\phi_{gtt}(v_i) \geq \phi_{tt}(v_i). \tag{9}$$

Proof: The relation (9) simply follows from the fact that both CS sweepings start from (v_i, h_i) to locate next node v_{i+1} but select the first hit node from different node sets $(\mathcal{N}_{v_{i-1}} \cap \mathcal{N}_{v_i})$ and $\mathcal{N}_{v_i}$, where $(\mathcal{N}_{v_{i-1}} \cap \mathcal{N}_{v_i}) \subseteq \mathcal{N}_{v_i}$. □

Proposition 4: If $\theta^*_{v_i} > \pi$ and $\phi_{gtt}(v_i) = \theta^*_{v_i}$, then $\phi_{gtt}(v_i) = \phi_{tt}(v_i)$.

Proof: Let $\theta^*_{v_i} > \pi$ and $\phi_{gtt}(v_i) = \theta^*_{v_i}$. By Lemma 4, $\phi_{gtt}(v_i) \geq \phi_{tt}(v_i)$. Suppose that $\phi_{tt}(v_i) < \phi_{gtt}(v_i)$. For $\phi_{tt}(v_i) < \theta^*_{v_i}$, $\theta^*_{v_i}$ cannot occur in the angular sector of $\phi_{gtt}(v_i)$ but explementary angle $2\pi - \phi_{gtt}(v_i) < \pi$ cannot contain $\theta^*_{v_i} > \pi$. Therefore it is only true that $\phi_{gtt}(v_i) = \phi_{tt}(v_i)$. □

Proposition 4 implies that through exploiting available information $\theta^*_{v_i}$ at node v_{i-1}, the modified CS sweeping scheme although operating at node v_{i-1} is able to find the same CS-hit node v_{i+1} as would be chosen as the next relay by the TT scheme operating at v_i if conditions $\theta^*_{v_i} > \pi$ and $\phi_{gtt}(v_i) = \theta^*_{v_i}$ are satisfied. Under such a circumstance, node v_i is on a locally concave network boundary and node v_{i+1} is directly accessible from v_{i-1}. Being on the network boundary constructed by CS sweeping and hence free of the edge intersection problem, relay candidate v_i can be skipped over. This raises the spirit of successive CS sweepings before packet forwarding, as revealed in the following corollary which is self-evident.

Corollary 2: Suppose that the condition of Proposition 4 is satisfied. Let $\angle v_i v_{i+1} v_{i+2}$ be denoted by $\phi_{tt}(v_{i+1})$ and by $\phi_{gtt}(v_{i+1})$ when v_{i+2} is the first hit node by CS swung from (v_{i+1}, h_{i+1}) to sweep nodes in $\mathcal{N}_{v_{i+1}}$ and in $\mathcal{N}_{v_{i+1}} \cap \mathcal{N}_{v_{i-1}}$ (or $C_{i+1} \cap C_{i-1}$), respectively. Then, Lemma 4 and Proposition 4 remain true for node v_{i+1}, $\phi_{gtt}(v_{i+1})$, $\phi_{tt}(v_{i+1})$ and $\theta^*_{v_{i+1}}$.

Our first modification of scheme TT, called eTT, is to compute and trace as far as possible a sequence of CS-hit nodes that would be selected by scheme TT but can be practically skipped, all based on locally available information. Under successive computations for locating CS-hit and -hinged nodes, the violation of $\theta^*_{v_j} > \pi$ and $\phi_{gtt}(v_j) = \theta^*_{v_j}$ of Proposition 4 determines a *stop place*, say node v_j, from which the node first hit by the modified CS sweeping is unsure to be the same one selected by original scheme TT. Since the computed path fragment to the stop place is what would be exactly followed by CS-sweeping, its correctness for bypassing network voids is ensured as validated in [12], [13]. Upon locating a stop at v_j, node v_{i-1} sends the packet including the next CS sweeping start point h_j to node v_j where a new sequence of modified CS sweepings will be initiated. Note that the stop place cannot be beyond the transmission range of the current relay. Also note that it is necessary to include start point h_j in the packet under operation mode eTT so that node v_j has the idea of where to start CS sweeping.

Algorithm 2 shows eTT in C-like code. Some operational points are as follows.

- After selecting the first hit node v_i in $\mathcal{N}_{v_{i-1}}$ (line 9) current relay v_{i-1} continues to perform CS sweeping from (v_{i+j}, h_{i+j}) for first hit and hinged node in $\mathcal{N}_{v_{i-1}}$ and in C_{i+j}, $j \geq 0$ (line 19 in the while loop) until condition $\phi_{gtt}(v_{i+j^*}) = \theta^*_{v_{i+j^*}} > \pi$ does not hold for some CS hinged node v_{i+j^*} (line 33 in the while loop). Node v_{i-1} then includes information (v_A, h_{i+j^*}) in the packet and forwards it to node v_{i+j^*} (lines 36–37).
- During above successive sweeping computations at node v_{i-1}, if CS is hinged at node v_A and swept to first hit h_A before any node, stop computations immediately for destination D unreachable (lines 11 and 27); and if any CS hinged node $v_{i+j^o}, j^0 \leq j^*$, to D is first found to be within the distance γ, stop computations immediately, set operation mode xGF in the packet and send it to v_{i+j^0} (lines 14 and 22).
- Upon receiving a packet of mode eTT, node v_{i+j^*} switches to greedy mode xGF immediately if $|v_{i+j^*}D| < |v_A D|$ (lines 4–5). Except the special case in line 14 or 22 mentioned previously, it implies that only the CS-hinged

node that has been selected as a relay is allowed to switch back to xGF. Before reaching this CS-hinged node, any previous CS-hit node even with a distance to D shorter than $|v_A D|$ is denied as a relay that can perform the mode switching. In Section 5, we'll also apply the conventional way by which the operation of CS-sweeping is immediately switched to xGF whenever possible and study its effect.

- Line 1 checks whether v_0 is isolated. This check (lines 1–3) can be moved and merged into lines 9–12. Line 26 is checked for STOP only when next hit node v_{i+1} is confirmed on the route of original Scheme TT in line 25.

4.2 XeTT, eTT with an extra tracing capability

By Proposition 4, eTT allows to track and bypass a sequence of relay nodes selected by CS-sweeping but surely fails and stops when a convex network boundary node is encountered. This raises a question on how to augment the capability of eTT to skip it. In fact, the shape of CS has been designed to bypass some of convex boundary nodes that would be selected by GAR. It has been shown in Property 1 of [13] that $\phi_{tt}(v_i) > \pi - \sin^{-1}\frac{|v_{i-1}v_{i+1}|}{2\gamma}$ if v_{i+1} is in the transmission range of CS-hinged node v_{i-1} (or in C_{i-1}). Thus $\phi_{tt}(v_i) > 5\pi/6$ if $|v_{i+1}v_{i-1}| \leq \gamma$, a more restricted case of Lemma 2. We'll follow previous notations $\phi_{gtt}(v_i)$ and $\phi_{tt}(v_i)$ in developing XeTT.

Let $\theta^e_{v_i}$ be the maximum of the angles between adjacent edges from vertex v_i to nodes in $\mathcal{N}_{v_{i-1}} \cap \mathcal{N}_{v_i}$ (or $C_{i-1} \cap C_i$) and in the circular sector delimited by explementary angle $2\pi - \phi_{gtt}(v_i)$, rays $v_i v_{i-1}$ and $v_i v_{i+1}$, where v_{i+1} is the node in $\mathcal{N}_{v_{i-1}} \cap \mathcal{N}_{v_i}$ first hit by CS hinged at v_i. Thus, $\theta^e_{v_i} \leq 2\pi - \phi_{gtt}(v_i)$.

Proposition 5: If $\phi_{gtt}(v_i) = \theta^*_{v_i}$ and $\theta^e_{v_i} < \theta^*_{v_i}$, then $\phi_{gtt}(v_i) = \phi_{tt}(v_i)$.

Proof: Let $\phi_{gtt}(v_i) = \theta^*_{v_i}$ and $\theta^e_{v_i} < \theta^*_{v_i}$. By Lemma 4, $\phi_{gtt}(v_i) \geq \phi_{tt}(v_i)$. Suppose that $\phi_{gtt}(v_i) > \phi_{tt}(v_i)$. By the definition of representative angle, either $\theta^*_{v_i} \leq \phi_{tt}(v_i) < \phi_{gtt}(v_i)$ or $\theta^*_{v_i}$ occurs in the explementary angle of $\phi_{gtt}(v_i)$ and satisfies $\theta^*_{v_i} \leq \theta^e_{v_i}$. Both violate the hypothesis. □

When the condition of Proposition 5 is satisfied, v_{i+1} is the one that would be selected by CS-sweeping and can be reached directly by v_{i-1} to bypass node v_i. XeTT further utilizes this condition in routing computations to track and skip CS-hit nodes for $\pi/3 < \theta^*_{v_i} \leq \pi$ besides the condition in Proposition 4 for $\theta^*_{v_i} > \pi$. To reduce computation complexity, $\theta^e_{v_i}$ is computed only after $\pi/3 < \theta^*_{v_i} \leq \pi$ and then $\phi_{gtt}(v_i) = \theta^*_{v_i}$ are found to hold. Here, the limit $\pi/3$ is due to Lemma 2.

XeTT completely follows the eTT in Algorithm 2 except that lines 18 and 25 in the while loop of the algorithm are replaced with the following **while** and **if** lines, respectively.

while ($\theta^*_{v_i} > \pi/3$ **and** *PseudoWalk* is **true**) **do**

  ......

  if $\theta^*_{v_i}$ is $\angle v_{i-1}v_iv_{i+1}$ and (($\theta^*_{v_i} > \pi$ or ($\theta^*_{v_i} \leq \pi$ and $\theta^e_{v_i} < \theta^*_{v_i}$)) **then**

    .....

  end if

end while

Algorithm 2: Enhanced TT (eTT) Scheme

Require: A packet with $(\text{mode } eTT, v_A, h_s)$ destined to D at node v_0.
$\{h_A$ is the intersection point of $v_A D$ and C_A. $\}$
Ensure: Send node v_i the packet with mode xGF or with $(\text{mode } eTT, v_A, h_s)$, switch to xGF, or STOP due to destination unreachable.
1: **if** $\theta^*_{v_0}$ is ∞ **then**
2: STOP {Destination unreachable}
3: **end if**
4: **if** $|v_0 D| < |v_A D|$ **then**
5: $Mode \longleftarrow$ xGF and switch to scheme xGF.
6: **end if**
7: $h_0 \longleftarrow h_s$
8: $i \longleftarrow 1$
9: Find $v_i \in \mathcal{N}_{v_0}$ first hit by CS counterclockwise from $v_{i-1}h_{i-1}$.
10: **if** v_{i-1} is v_A and h_A is hit by CS before v_i **then**
11: STOP {Destination unreachable}
12: **end if**
13: **if** $|v_i D| < \gamma$ **then**
14: $Mode \longleftarrow$ xGF and goto step 37.
15: **end if**
16: Find h_i, the intersection of C_{i-1} and C_i on the right-hand side of $v_{i-1}v_i$.
17: *PseudoWalk* $\longleftarrow$ **true**;
18: **while** ($\theta^*_{v_i} > \pi$ **and** *PseudoWalk* is **true**) **do**
19: Find $v_{i+1} \in \mathcal{N}_{v_0}$ first hit by CS counterclockwise from $v_i h_i$.
20: **if** $|v_{i+1} D| < \gamma$ **then**
21: $i \longleftarrow i+1$
22: $Mode \longleftarrow$ xGF, break and goto step 37.
23: **end if**
24: Compute $\angle v_{i-1}v_iv_{i+1}$.
25: **if** $\theta^*_{v_i}$ is $\angle v_{i-1}v_iv_{i+1}$ **then**
26: **if** v_i is v_A and h_A is hit by CS before v_{i+1} **then**
27: break and STOP {destination unreachable.}
28: **else**
29: $i \longleftarrow i+1$
30: Find h_i, the intersection of C_{i-1} and C_i on the righthand side of $v_{i-1}v_i$.
31: **end if**
32: **else**
33: *PseudoWalk* $\longleftarrow$ **false**
34: **end if**
35: **end while**
36: $h_s \longleftarrow h_i$
37: Send node v_i the packet with *Mode* and with (v_A, h_s) if *Mode* is eTT.

4.3 Combine xGF with TT, eTT or XeTT

When CS is hinged at a network boundary node to start sweeping, all CS-hinged nodes actually or computationally visited are located on network boundaries. The guarantee of detecting a routing loop thus relies on whether stuck node v_A is on a network boundary and of local minimum. Accordingly, GF, MeGF or MQGF can be combined with any of TT, eTT and XeTT to achieve correct routing operations. Any of such combinations {eGF, QGF} + {eTT, XeTT} for routing may fail to detect a routing loop because stuck node v_A by eGF or QGF may not be on a network boundary and because eTT and XeTT do not insist on switching to xGF immediately whenever a CS-hit node is found closer to D than stuck node v_A. However, eGFTT (or QGFTT) is able to provide the guarantee but takes a longer routing path to detect a routing loop. This is because CS-sweeping will stop and the operation mode immediately changes to eGF(or QGF) whenever a CS-hit node, say v_j, satisfies $|v_jD| < |v_AD|$. When v_j has been on a network boundary and of local minimum and the operation switches back to TT again, anchor v_A is now v_j and hence a routing loop if existing will be detected.

To illustrate the previous points, we apply three routing schemes in a connected network subgraph of seven nodes without connection to D as shown in Fig. 6. When eGFeTT is employed, the packet created at S for D will visit v_4 and then be stuck at v_1 by eGF. With stuck node information $v_A \leftarrow v_1$, the packet routed by eTT will visit v_2, v_3 and v_4, skip CS-hit nodes S and v_4, arrive at v_5, skip CS-hit nodes v_0 and v_3 and arrive at v_4. Then, the packet will be sent back and forth between v_4 and v_5 in eTT mode because anchor node v_1 is not on network boundary and no longer hit by any successive CS sweeping. When MeGFeTT is employed, the packet will visit v_4 and v_1 and be stuck at v_0 by MeGF. With stuck node information $v_A \leftarrow v_0$, the packet routed by eTT will visit v_3 and v_4, skip CS-hit nodes S and v_4, and arrive at v_5 where the event of destination unreachable is detected upon computing the next CS-hit node from (v_0, h_0). When eGFTT is employed, the packet will visit v_4 and be stuck at v_1. With stuck node information $v_A \leftarrow v_1$, the packet routed by TT will visit $v_2, v_3, v_4, S, v_4, v_5$, arrive at v_0 where routing operation switches from TT to eGF, updates anchor node $v_A \leftarrow v_0$ and switches back to TT again, then visit v_3, v_4, S, v_4, v_5 and finally stop at v_0 where a routing loop is detected.

5 SIMULATIONS AND DISCUSSIONS

Let transmission radius $\gamma = 1$ for all network scenarios. We consider two types of network topology of the same size 40×40, type I having no artificial void and type II having two as shown in Fig. 7. Each node is uniformly distributed in the network area except the two voids in type II. The number of nodes distributed in network type I is $2400, 2800, 3200, 4000, 4800, 5600$ or 6400 and that in network type II is $1920, 2240, 2560, 3200, 3840, 4480$ or 5120. For each network type and node density, 2×10^5 random network graphs are created for simulations. This represents a sample size of $m = 2 \times 10^5$.

In each network graph as numbered by i, the node nearest to position $(3, 3)$ is designated as source S_i while the one nearest to $(37, 37)$ is destination D_i. When a packet is sent from S_i to D_i, it will stop and find destination unreachable at a stuck node v_i or reach D_i after a number of transmission hops denoted by L_i. For loop-free routing protocols, $L_i < \infty$ for all i.

Let m_s be the number of successful routings among m network graphs and L_i the number of hops taken by the i^{th} successful routing from S_i to D_i. The average path hop count of the successful transmissions, denoted as H_s, is

$$H_s = \frac{\sum_{i=1}^{m_s} L_i}{m_s} \ . \tag{10}$$

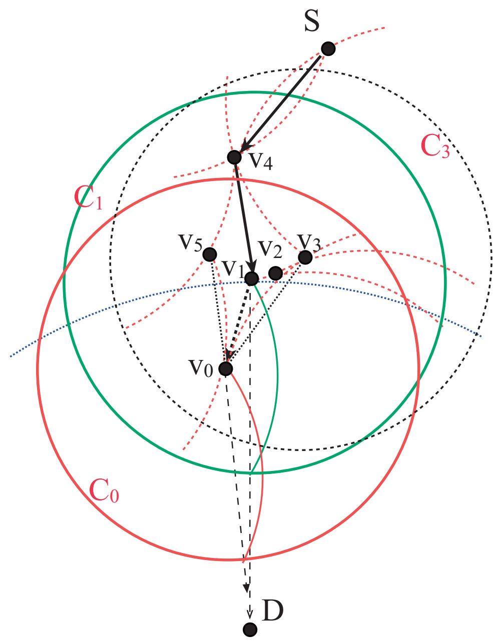


Fig. 6. A network graph consists of a connected subgraph of seven nodes and an isolated node D. For eGFeTT, a packet created at S for D will visit v_4 and v_1 by eGF and $v_2, v_3, v_4, v_5, v_4, v_5, \cdots$ by eTT. A loop routing infinitely thus occurs. For MeGFeTT, the packet will visit v_4, v_1 and v_0 by MeGF and v_3, v_4 and v_5 by eTT where a routing loop is detected. For eGFTT, the packet will visit v_4 and v_1 by eGF, $v_2, v_3, v_4, S, v_4, v_5$ and v_0 by TT and v_3, v_4, S, v_4, v_5 and v_0 by TT where a routing loop is detected.

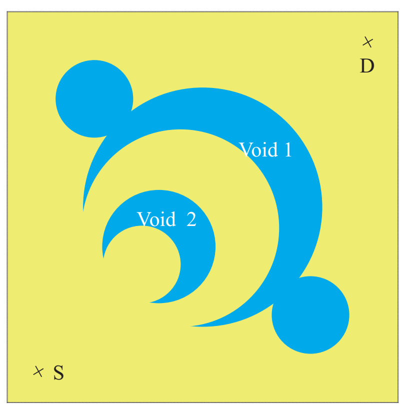


Fig. 7. A network area of 40×40 in the first quadrant has two artificial voids. Void 1 includes the circular areas of radius $r_1 = 12$ centered at $(20, 20)$, radius $r_3 = 4$ centered at $(31, 9)$ and radius $r_4 = 4$ centered at $(9, 31)$, but not in the circular area of radius $r_2 = 10$ centered at $(18, 18)$. Void 2 includes the circular area of radius $r_5 = 6$ centered at $(16, 16)$ but not in the circular area of radius $r_6 = 4$ centered at $(14, 14)$. The two voids occupy roughly 20% of the total network area.

Among m network graphs, the number of graphs where S_i and D_i, $1 \leq i \leq m$, are connected is listed in the second and fifth columns, denoted by m_s, of Table 1 for network types I and II, respectively. Each entry of the two columns also represents the number of successful routings achieved by every routing scheme presented here in those m network graphs for a given number of nodes listed in the first or fourth column. The H_s by GFTT for

TABLE 1

Number of nodes in a network graph, number of successful routings m_s in $m = 2 \times 10^5$ network graphes, and the average routing path hop count H_s by GFTT.

No artificial void (type I)			Two artificial voids (type II)		
Nodes	m_s	H_s	Nodes	m_s	H_s
2400	31103	1972.38	1920	265	2083.24
2800	137076	975.73	2240	21197	1653.96
3200	180428	434.62	2560	113163	1036.36
4000	197990	124.87	3200	195412	389.16
4800	199723	79.87	3840	199647	251.47
5600	199950	69.77	4480	199946	218.92
6400	199988	65.60	5120	199989	206.28

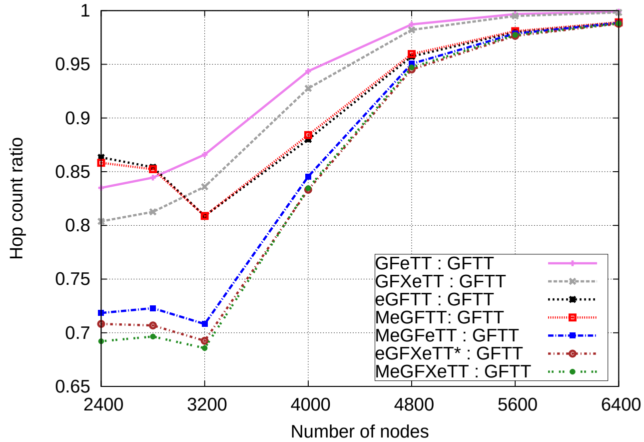
the two types of network are listed in columns 3 and 6, respectively.

5.1 The benefit of RAI

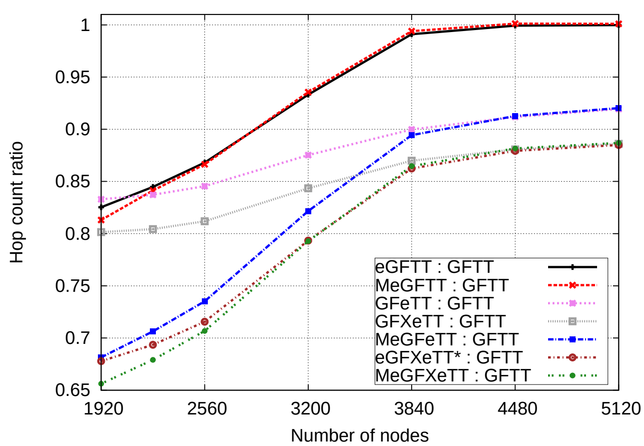
The mean successful routing path length H_s defined in (10) for routing scheme GFeTT, GFXeTT, eGFTT, MeGFTT, MeGFeTT, eGFXeTT*, or MeGFXeTT is all normalized relative to the H_s of GFTT and plotted in Figs. 8(a) and 8(b) for network types I and II, respectively. Here eGFXeTT* represents a scheme combining eGF with a modified XeTT by which operation mode XeTT is immediately switched to eGF whenever a CS-hit node is found closer to destination D than stuck node v_A, a conventional approach stated in Sec 4.1 for Algorithm 2. True values of H_s for each scheme can be obtained by multiplying the corresponding baseline data of H_s by GFTT listed in Table 1. In the figures, those for schemes involving enhance factor ω have been optimized.

From curves eGFTT and MeGFTT in both figures, it can be seen that the effect of exploiting RAI by forwarding scheme eGF or MeGF on shortening routing hop count H_s is significant when network node densities are low but less significant when node densities are high. This is mainly due to the number of ignorable CNNs increasing with decreasing network densities and due to the efficient way by (7) to utilize RAI. Comparing eGFTT and MeGFTT, we could hardly notice that the former performs slightly better than the latter when network densities are high and vice versa. This is because eGF potentially saves one hop to a CNN before stuck in high density networks in which each random network void tends to be small. For low density networks in which random network voids are statistically larger, MeGF is potentially stuck at an anchor node v_A much closer to D than eGF. This allows scheme TT to capitalize on its efficient way to traverse network boundary longer. However, eGFTT will generally consume more hops to detect a routing loop as addressed in Section 4.3.

We can see from curves GFeTT and GFXeTT in both figures that the impact of scheme eTT or XeTT on reducing routing path length H_s is remarkable when networks have low node densities to create more or longer void boundaries statistically. However, the saving of hop counts remains significant when network densities are high and network voids are obviously located between S and D, as can be seen from Fig. 8(b). Comparing GFeTT and GFXeTT in both figures, we see that there is roughly a saving of 3 percentages of hop counts by XeTT than by eTT



(a)



(b)

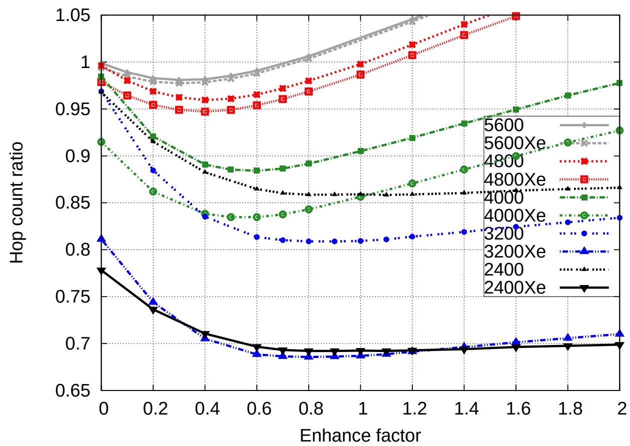
Fig. 8. Routing path hop count (H_s) ratio versus numbers of nodes (network densities) for networks with (a) no artificial void and (b) two artificial voids.

when network densities are low or network voids are obviously located on the way from S to D. This saving, due to skipping some locally convex boundary nodes, decreases with network densities for network type I as shown in Fig. 8(a) because less and smaller voids exist in high density networks.

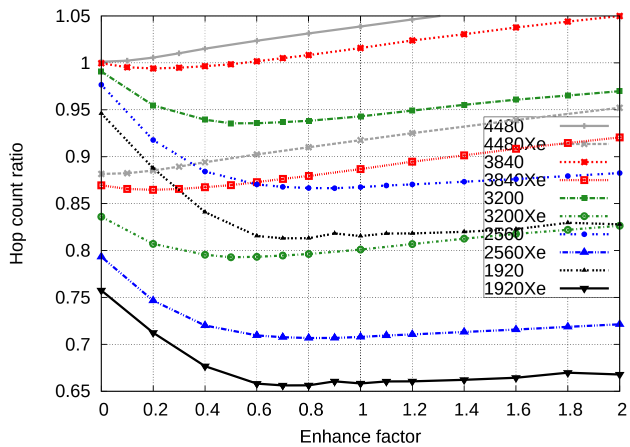
Curves MeGFeTT and MeGFXeTT in both figures illustrate the combined effect of utilizing RAI for greedy forwarding and greedy network boundary traversal on shortening routing hop counts. We can see that the percentage of saving by MeGFXeTT (or MeGFeTT) is almost equal to the sum of savings by MeGFTT and by GFXeTT (or GFeTT).

Comparing eGFXeTT* and MeGFXeTT in both figures, we can see the effect of the conventional way versus our way about the instant recovery mode operation is changed to greedy one. In fact, eGFXeTT* presents an unnoticeable performance advantage over MeGFXeTT when network densities are high. This is partly because of a similar reason for eGFTT and MeGFTT mentioned previously and partly because XeTT* is able to avoid the event of skipping a potential stuck node and then consuming hops to fall back.

Figs. 9(a) and 9(b) illustrate the impact of enhance factor ω on the ratios of H_s by MeGFTT and by MeGFXeTT



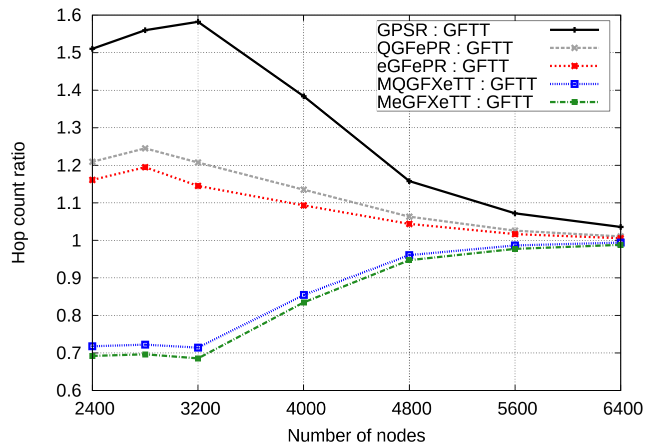
(a)



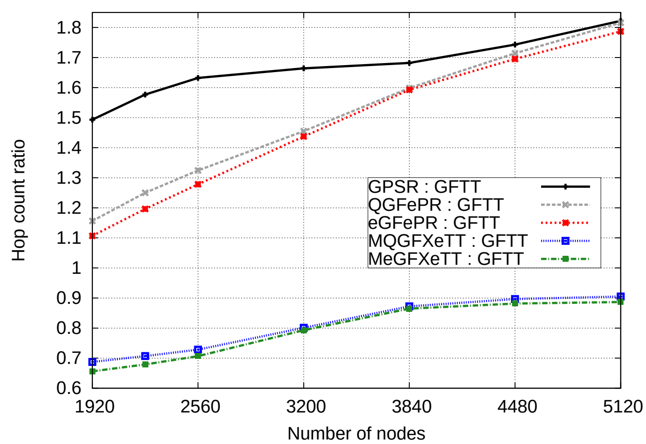
(b)

Fig. 9. Path hop count ratios $\frac{H_s(\mathrm{MeGFXeTT})}{H_s(\mathrm{GFTT})}$, indicated by suffix "Xe" to each number of nodes, and $\frac{H_s(\mathrm{MeGFTT})}{H_s(\mathrm{GFTT})}$ versus enhanced factor ω for networks with (a) no artificial void and (b) two artificial voids.

to that by GFTT for network types I and II, respectively. All curves are convex except some irregularity due to simulations. For each given number of network nodes, curves for MeGFTT and MeGFXeTT in both figures appear to keep a constant performance gap and go in the same fashion as ω increases. As can be seen from both figures that the ratios of H_s are more sensitive to selected values of ω when the density of network nodes is high. It can be seen that optimal ω for minimum H_s ratio decreases with network node densities. This is because RAI becomes less important in high density networks where greedy forwarding is generally able to proceed far at optimal per-hop forwarding distance besides very few CNNs existing in such networks. For low density networks, we can simply set $\omega = 0.8$ for MeGFTT and MeGFXeTT to achieve a remarkable saving of path hop counts. For network type II, those curves in Fig. 9(b) appear to have smaller curvature at the minimum of H_s ratios as compared to those in Fig. 9(a) but exhibit a similar fact on optimal ω for minimum H_s ratio. In simulations, both MeGFTT and MeGFXeTT achieve the minimum H_s for networks of 4480 and 5120 nodes at $\omega = 0$.



(a)



(b)

Fig. 10. Path hop count H_s ratio versus number of nodes for networks with (a) no artificial void and (b) two artificial voids. Here GPSR is GFPR.

5.2 The efficiency of bypassing network voids

Figs. 10(a) and 10(b) show the ratio of average path hop count H_s by GPSR, QGFePR, eGFePR, MQGFXeTT and MeGFXeTT to that by GFTT for network types I and II, respectively. Those for eGFePR and MeGFXeTT have been optimized over enhance factor ω. Note that the H_s ratio by GFTT is always 1.

We can see from Fig. 10 (a) that all H_s ratios tend to converge when networks have high node densities and no artificial voids. This is because each greedy forwarding scheme in such network scenarios rarely encounters CNNs or stuck nodes and is almost good enough to reach destination in a short path stretch. However, MQGFXeTT and MeGFXeTT exhibit their efficient way by XeTT to bypass network voids when networks have artificial voids or low node densities, as can be seen from both figures in Fig. 10.

Without involving enhance factor ω, QGFePR and MQGFXeTT suffer some performance loss as compared with eGFePR and MeGFXeTT, respectively.

Comparing curves GPSR and eGFePR in Figs. 10(a) and 10(b), we see that eGFePR significantly outperforms GPSR when network densities are low. This reflects the benefit of virtually omitting CNNs in a network graph besides that of selecting rule (7). However, by the evidence that MeGFXeTT or MQGFXeTT outperforms eGFePR in all network scenarios, particularly significant when networks tend to have void regions, we can conclude that XeTT based on non-graphical CS sweeping is a much better approach to bypassing network voids than ePR based on planar graph RNG.

6 Concluding remarks

We have utilized the maximum angle between successive edges from a node as vital local state information for making routing decisions to reduce routing path hop counts. For greedy forwarding, such neighbor's maximum angle information allows a partial vision about next two hops deterministically, in that a CNN neighbor is definitely a candidate of local minimum that should be excluded and that a SPN neighbor definitely entails further advancement. Furthermore, the size of a neighbor's maximum angle reveals stochastic topology information on the number of next two hop relay candidates. They have been embodied in forwarding variants eGF and MeGF. We have also shown that simply by further restricting selection of non-CNN in RNG for relay, eGFePR, modified from GPSR, is able to substantially save hop counts for networks of low node densities.

Overlaying the rotational CS-sweeping mechanism, eTT and XeTT utilize the maximum angle of the CS hinged node to determine whether the next node hit by CS sweeping performed by the current node in lieu of the CS hinged one is on the network boundary constructed by original scheme TT. This allows them to take advantage of correctness and efficiency from scheme TT and skip unnecessary hops. The study has revealed that XeTT presents genuinely favored features of greedy network boundary traversal for saving hop counts and that MeGFXeTT is extremely efficient for routing over networks having voids or sparse nodes. An alternative to the scheme, MQGFXeTT has the benefit of no enhance factor parameter although losing some performance.

Appendix A

Proof of Proposition 2

Let $f(\theta|\theta_v^*)$ and $f(\phi|\theta_v^*,\theta)$ be the conditional probability density functions given θ_v^* and (θ_v^*,θ), respectively. By assumption, θ is uniformly distributed in $(0, 2\pi-\theta_v^*]$ and hence has the distribution

$$f(\theta|\theta_v^*) = \frac{1}{2\pi-\theta_v^*}\ , 0<\theta\leq 2\pi-\theta_v^*. \tag{11}$$

For a node uniformly tossed into angular areas limited in $(0,\theta]$ and $(\theta+\theta_v^*, 2\pi]$, its angle ϕ has the distribution

$$f(\phi|\theta_v^*,\theta) = \begin{cases} \frac{1}{2\pi-\theta_v^*}, & 0<\phi\leq\theta \\ 0, & \theta<\phi\leq\theta+\theta_v^* \\ \frac{1}{2\pi-\theta_v^*}, & \theta+\theta_v^*<\phi\leq 2\pi. \end{cases} \tag{12}$$

Apply the chain rule and use (11) to obtain for $\phi\in(0,2\pi]$

$$\begin{aligned} f(\phi|\theta_v^*) &= \int_0^{2\pi-\theta_v^*} f(\phi|\theta_v^*,\theta) f(\theta|\theta_v^*) d\theta \\ &= \int_0^{2\pi-\theta_v^*} f(\phi|\theta_v^*,\theta)\frac{d\theta}{2\pi-\theta_v^*}. \end{aligned} \tag{13}$$

The remaining step is to apply (12) to (13). Through careful manipulation of two dimensional domain of θ and ϕ, we have (4).

APPENDIX B
PROOF OF PROPOSITION 3

Let f_{SD} be a path from S to D and node $v_i \neq S, D$ be a CNN on the path. We need to prove that, after removing all CNNs (except S and D if they are CNNs) and all edges connected to them, there is still a path between S and D. We consider three cases: $\theta^*_{v_i} = \infty$, $\theta^*_{v_i} = 2\pi$, and $5\pi/3 \leq \theta^*_{v_i} < 2\pi$. By 1) in Lemma 1, since node v_i is not isolated we have $\theta^*_{v_i} \neq \infty$. Now assume that $\theta^*_{v_i} = 2\pi$. Let node v_j be the only neighbor of v_i. By 2) in Lemma 1, v_j is not a CNN; otherwise v_i and v_j form an isolated network fragment. It is clear that there exists a path segment $(v_j, v_i),\ (v_i, v_j)$ on path f_{SD}. Hence, node v_i and all edges connected to it can be removed without breaking S to D.

Now we consider $5\pi/3 \leq \theta^*_{v_i} < 2\pi$. Let v_j and v_k be neighbors of v_i and v_jv_i and v_iv_k be two edges on path f_{SD}. By 3) in Lemma 1, v_j and v_k are connected. Assume that both v_j and v_k are not CNNs. Then we can remove path segment $(v_j, v_i), (v_i, v_k)$ from path f_{SD} and replace it with (v_j, v_k) such that S and D remain connected on the modified path $f_{SD} - \{(v_j, v_i), (v_i, v_k)\} + \{(v_j, v_k)\}$. If v_j and v_k are the same node, we can remove (v_j, v_i), (v_i, v_j) from path f_{SD}. Hence v_i and all edges connected to it can be removed without breaking S to D.

Assume that both v_j and v_k are CNNs. First we consider the case that v_j is the first node on path f_{SD}, i.e. v_j is S. If v_k is D, then we can reroute S directly to D and remove v_i and all edges connected to it. If v_k is not D, by 3) in Lemma 1, there exists a non-CNN v_ℓ inside the triangle formed by S, v_i, and v_k such that v_kv_ℓ is on f_{SD}. We can reroute S directly to v_ℓ since they are connected, and remove v_i, v_k, and all edges connected to them. Now assume that v_j is not the first node on f_{SD}. By 3) in Lemma 1, there exists a non-CNN v_ℓ inside the triangle formed by v_i, v_j, and v_k such that $v_\ell v_j$ is on f_{SD}. If v_k is D, then we can reroute the path from v_ℓ to D and remove v_i, v_j, and all edges connected to them. Assume that v_k is not D. Then there exists a non-CNN node v_m inside the triangle formed by v_i, v_j, and v_k such that v_kv_m is on f_{SD}. Then we can reroute v_ℓ directly to v_m since they are connected. Hence, we can remove v_i, v_j, v_k, and all edges connected to them without disconnecting S to D. The argument also holds for $v_l = v_m$.

Next we assume that v_j is a CNN but v_k is not. Since v_i and v_k are connected to v_j, obviously $5\pi/3 \leq \theta^*_{v_j} < 2\pi$. If v_j is the first node on path f_{SD}, then v_j is S and we can reroute S directly to v_k. Removing v_i and all edges connected to it does not disconnect S to D. Now assume that v_j is not the first node on the path. Let v_ℓ be the node on the path f_{SD} that directly connects to v_j. If v_ℓ is a CNN, then v_i, v_j, and v_ℓ are CNNs. By the argument in previous paragraph we can reroute S to D. Now assume that v_ℓ is not a CNN. By (6), we have $|v_\ell v_k| < \gamma$, i.e. v_ℓ and v_k are connected. Then we can reroute v_ℓ directly to v_k to form a new path from S to D, or take a shortcut at v_ℓ if $v_\ell = v_k$. Hence, we can remove v_i, v_j (If v_j is not S), and all edges connected to them. Note that even if v_j is S,[1] it is clear that the above reroute is still valid. Similar argument can be applied to the case when v_k is a CNN

1. In this case, S appears on the path f_{SD} at least twice.

but v_j is not. The only difference in this case is that we need to consider whether v_k is D or not.

Therefore any path between S and D can be modified by removing CNNs and adding edges by the above procedure recursively until it involves no CNN, except S and D and such modifications do not break them.

REFERENCES

[1] H. Takagi and L. Kleinrock, "Optimal transmission ranges for randomly distributed packet radio terminals," *IEEE Tran. Commun.*, vol. 32, pp. 246–257, Mar. 1984.

[2] G. G. Finn, "Routing and addressing problems in large metropolitan-scale internetworks," Univ. Southern California, Tech. Rep. ISI/RR-87-180, Mar. 1987.

[3] E. Kranakis, H. Singh, and J. Urrutia, "Compass routing on geometric networks," in *Proc. 11th Canadian Conf. Computational Geometry,* Vancouver, BC, Canada, Aug. 1999, pp. 51-54.

[4] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. IEEE/ACM Mobicom*, Boston, MA, Aug. 2000, pp. 243–254.

[5] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia,"Routing with guaranteed delivery in ad hoc wireless networks," in *Proc. 3rd Int. Workshop Discrete Algorithms and Methods for Mobile Computing and Commun.*, 1999, pp. 48–55.

[6] B. Karp, "Geographic routing for wireless networks," Ph.D. thesis, Harvard Univ., Cambridge, MA, 2000.

[7] K. R. Gabriel and R. R. Sokal, "A new statistical approach to geographic variation analysis," *Systematic Zoology*, vol. 18 (3), pp. 259–278, 1969.

[8] G. T. Toussaint, "The relative neighborhood graph of a finite planar set," *Pattern Recognition*, vol. 12, no 4, pp. 261–268, 1980.

[9] Q. Fang, J. Gao, and L. Guibas,"Locating and bypassing routing holes in sensor networks," in *Proc. IEEE INFOCOM*, 2004, vol. 4, pp. 2458-2468.

[10] Wen-Jiunn Liu, Kai-Ten Feng,"Greedy routing with anti-void traversal for wireless sensor networks", *IEEE Trans. on Mobile Computing,* vol. 8, no. 7, pp. 910-922, 2009.

[11] S. Rührup, H. Kalosha, A. Nayak, and I. Stojmenović, "Message-efficient beaconless georouting with guaranteed delivery in wireless sensor, ad hoc, and actuator networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 1, pp. 95-108, Feb. 2010.

[12] S. Rüehrup and I. Stojmenović, "Optimizing communication overhead while reducing path length in beaconless georouting with guaranteed delivery for wireless sensor networks," *IEEE Tran. Comput.,* vol. 62, no. 12, pp. 2440-2453, Dec. 2013.

[13] A. Mostefaoui, M. Melkemi, and A. Boukerche, "Localized routing approach to bypass holes in wireless sensor networks," *IEEE Tran. Comput.,* vol. 63, no. 12, pp. 3053-3065, Dec. 2014.

[14] T.-C. Hou and V. O.K. Li, "Transmission range control in multihop packet radio networks," *IEEE Trans. Commun.*, vol. 34, no. 1, Jan. 1986.

[15] L. Kleinrock and J. A. Silvester, "Optimum transmission radii for packet radio networks or why six is a magic number," in *Conf. Rec. Nat. Telecommun. Conf.,* Dec. 1978, pp. 4.3.1-4.3.5.

[16] F. Kuhn, R. Wattenhofer, and A. Zollinger, "An algorithmic approach to geographic routing in ad hoc and sensor networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 1, pp. 51–62, Feb. 2008.

[17] F. Kuhn, R. Wattenhofer, Y. Zhang, and A. Zollinger,"Geometric routing: Of theory and practice," in *Proc. 22nd ACM Symp. Principles of Distrib. Computing (PODC),* 2003, pp. 63-72.

[18] Z. Jiang, J. Ma, W. Lou, and J. Wu,"An information model for geographic greedy forwarding in wireless ad-hoc sensor networks," in *Proc. IEEE INFOCOM,* 2008, pp. 1499–1507.

[19] Q. Fang, J. Gao, L. Guibas, V. de Silva, and L. Zhang, "GLIDER: Gradient landmark-based distributed routing for sensor networks," in *Proc. IEEE INFOCOM*, 2005, vol. 1, pp. 339-350.

[20] G. Tan, M. Bertier, and A.-M. Kermarrec, "Visibility-graph-based shortest-path geographic routing in sensor networks," in *Proc. IEEE INFOCOM*, 2009, pp. 1719-1727.

[21] G. Tan and A.-M. Kermarrec, "Greedy geographic routing in large-scale sensor networks: a minimum network decomposition approach," *IEEE/ACM Trans. Netw.*, vol. 20, no. 3, pp. 864–877, June 2012.

[22] S. S. Lam and C. Qian, "Geographic routing in *d*-dimensional spaces with guaranteed delivery and low stretch," *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, pp. 663–677, April 2013.

[23] Y.B. Ko and N.H. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks," in *Proc. ACM/IEEE MobiCom,* 1998, pp. 66-75.

[24] M. Zorzi and R. R. Rao, "Geographic random forwarding (GeRaF) for Ad Hoc and sensor networks: multihop performance," *IEEE Trans. Mobile Computing,* vol. 2, no. 4, pp. 337-348, Oct.-Dec. 2003.

[25] J. A. Sanchez, P. M. Ruiz, and R. Marin-Perez,"Beacon-less geographic routing made practical: challenges, design guidelines, and protocols," *IEEE Commun. Mag.,* vol. 47, no. 8, pp. 85V91, Aug. 2009.

[26] J.-T. Tsai and Y.-C. Li, "Quasi-greedy geographic routing in wireless networks," in *Proc. IEEE GlobeCom*, Anaheim CA, Dec. 2012, pp. 26-31.

[27] S. Ross, *Stochastic Processes*, 2^{nd} Ed. New York: John Wiley & Sons, 1996.

[28] I. Stojmenovic and X. Lin, "Loop-free hybrid single-path/flooding routing algorithms with guaranteed delivery for wireless networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 12, pp. 1023–1032, Oct. 2001.

[29] D. Chen and P.K. Varshney,"A survey of void handling techniques for geographic routing in wireless networks," *IEEE Comm. Surveys and Tutorials,* vol. 9, no. 1, pp. 50-67, 2007.

[30] T. Aguilar, S.-J. Syue, V. Gauthier, H. Afifi, and C.-L. Wang, "CoopGeo: A beaconless geographic cross-layer protocol for cooperative wireless ad hoc networks," *IEEE Trans. Wireless Commun.,* vol. 10, no. 8, pp. 2554–2565, Aug. 2011.