Virtual Surrounding Face Geocasting with Guaranteed Message Delivery for Ad Hoc and Sensor Networks

Jie Lian[†], Kshirasagar Naik[†], Yunhao Liu[‡], and Lei Chen[‡]

[†]Department of E&CE, University of Waterloo, Waterloo, ON, Canada. {jlian, knaik}@swen.uwaterloo.ca
[‡]Department of CSE, Hong Kong University of Science and Technology, Hong Kong. {liu, leichen}@cse.ust.hk

Abstract

Geocasting in wireless sensor networks and ad hoc networks is the delivery of a message from a source to all the nodes in a given geographical region. The objectives of a geocasting protocol are two-folds: guaranteed message delivery and low transmission cost. Most of the existing protocols do not guarantee message delivery, and those that do incur a high transmission costs. In this paper, we introduce the idea of a Virtual Surrounding Face (VSF), and present a geocasting protocol based on VSF. By using mathematical analyses and simulation studies, we show that the proposed protocol guarantees message delivery and has a significant lower transmission cost than the existing approaches.

I. INTRODUCTION

A wireless sensor network can be treated as a distributed sensor database system that supports various types of query services. One type of query, called a *zone-based* query, requires the participation of all the sensors within a geographical region, called a query zone, to build a query response. One example of this type of queries might be, to say, *locate all the wheeled vehicles in a specific sub-area for the next two hours*. To support such a query, a monitoring center should transmit the query message to all sensors in the query zone. The idea of sending a message to all the nodes in a given geographic area is called *geocasting* [18].

An important objective of a geocasting algorithm is to achieve guaranteed message delivery while maintaining a low cost (i.e., a lower number of transmissions). Guaranteed delivery ensures that, every sensor in a query zone receives a copy of the geocasting message. Since sensors are generally powered by batteries, the limited energy of sensors requires geocasting to consume as little energy as possible. Many algorithms have been proposed in literature [9-18] to achieve geocasting. However, the approaches presented in [9-16] do not guarantee message delivery and incur high transmission costs. Of the existing approaches, four algorithms—one in reference [17] and three in reference [18]—guarantee message delivery in continuous geocasting regions. However, these algorithms have high transmission costs.

In this paper, we propose a geocasting algorithm based on the idea of *Virtual Surrounding Face* (VSF) and we refer to this algorithm as VSF Geocasting (VSFG). We prove that VSFG guarantees message delivery to the nodes within a geocasting region. Also, the transmission cost of VSFG is significantly lower than those of the existing approaches and its cost is close to the optimal number of transmissions.

We consider the geocasting problem in connected sensor networks and the network connectivity can be found in [19]. In VSFG, each network topology graph is converted into a *planar graph* where no two edges cross one another. The network area is then partitioned into a set of faces, where a *face* is a continuous area enclosed by a sequence of edges. In VSFG, all the faces intersecting with a geocasting region R are merged into a unique virtual surrounding face containing R. Then VSFG works as follows. First, a source node delivers a geocasting message to a node on the boundary of the VSF, called a boundary node. Second, the boundary node initiates a traversal process in which all the nodes on the boundary of the VSF receive a copy of the message. Finally, during the traversal process, nodes within R that overhear the traversal message perform restricted flooding within R. We summarize the major contributions of this paper as follows.

1) We introduce the concept of VSF, and present an algorithm (VSFG) based on VSF to achieve geocasting with guaranteed message delivery. We show that VSFG is fully distributed; in which each node in a network only needs to maintain the information of its one-hop neighbors.

2) The RFIFT (Restricted Flooding with Intersected Face Traversal) [18] has the lowest transmission cost among all the previous algorithms. We contrast the upper bounds of the message complexities of RFIFT and VSFG. In RFIFT, the number of transmissions required to traverse the faces and to perform restricted flooding is bounded by $3n + k$, where n is the number of nodes on the boundary of the faces intersecting a geocasting region R but not in R, and k is the total number of nodes within R. In proposed VSFG, this bound has been reduced to $2n + k$.

3) We evaluate VSFG through comprehensive simulations in different network environments. We show that VSFG achieves up to a 40% reduction in the total number of messages required by RFIFT.

4) VSFG is designed to be used in static sensor networks, while it can also be employed in mobile ad hoc networks under an assumption made in RFIFT, in which the nodes involved in a geocasting task do not change their position significantly during the geocasting task.

The paper is structured as follows. In Section II, we review related work. We define some terms and describe the concept of VSF in Section III. We present VSFG in Section IV. The performance of VSFG is evaluated in Sections V and VI. We conclude this work in Section VII.

II. RELATED WORK

Geocasting algorithms [4, 17, 18] reduce transmission costs by using location-based routing to deliver a message to a node in a geocasting region. The node performs restricted flooding within the region. Hence, we review two categories of related work: location-based routing and geocasting algorithms.

A. Location-based Routing

Location-based routing techniques have been extensively studied in literature [1-7]. In these techniques, every node in a network knows its geographic location and the locations of all its neighbors. When a source node transmits a message to a destination node with a known location, the source and all intermediate forwarding nodes make their routing decisions based solely on their destination locations and the locations of their neighbors.

Finn [1] proposed the first formal location-based routing based on a *greedy principle*. In greedy routing, each node chooses the neighbor with the minimum distance to the destination as its next forwarding node. Such an algorithm fails if a *void* (a large area without nodes) exists in the routing direction, that is, the message reaches a node that is closer to the destination than any of its neighbor nodes.

To ensure message delivery, face routing was introduced in [2]. In face routing, a planar graph derived from the network topology is used, and the network area is partitioned into a set of faces. To transmit a message from a source s to a destination t, the message traverses the face intersecting the line segment st from s to t. If an edge e on the boundary of the traversed face intersects with st and the intersecting point is closer to t than to s, the face, which is next to e and closer to t than the currently traversed face, is traversed. This process is repeated until t is found. Face routing ensures message delivery, but it might use long forwarding paths [3, 4].

To find a routing path close to the optimal path, the Greedy-Face-Greedy (GFG) algorithm, combining greedy routing and face routing, was proposed [3, 4]. In GFG, nodes perform greedy routing whenever it is possible. In the case when a void exists in the forwarding direction, face routing is used to send the message around the void. Hence, GFG guarantees delivery and significantly reduces the hop lengths of forwarding paths. For dense networks, the average length of forwarding paths is approximately equal to that of the shortest hop path. An alternative implementation of GFG, called Greedy Perimeter Stateless Routing (GPSR), was presented in [5] by including the IEEE 802.11 MAC protocol. However, both of these algorithms are not asymptotically optimal [6].

Adaptive Face Routing (AFR) [6] is known as the first algorithm that combines face routing and the greedy principle,

and achieves asymptotically optimal of routing path lengths. In a follow up paper [7], GOAFR+ was proposed with an asymptotic optimality and average case efficiency.

B. Geocasting Algorithms

Geocasting can be easily achieved by flooding the network, thereby achieving guaranteed message delivery. However, flooding is not energy efficient since it requires at least N transmissions, where N is the total number of nodes in the network. Three classes of geocasting algorithms have been studied in the literature to reduce the flooding cost.

In the first class of algorithms, a restricted forwarding zone, covering both the source node and the geocasting region, is used to limit the scope of flooding [9, 12, 13, 16]. In Location-Based Multicast (LBM) [9], the minimum rectangular area containing both the source node and the geocasting region is chosen as the forwarding zone. Next, restricted flooding is performed by nodes within the forwarding zone. Two later approaches [12, 13, 16] using forwarding zones were proposed to improve the performance. These two approaches differ from LBM in the ways a forwarding zone is selected. Even though the three algorithms reduce the flooding area, they still incur high flooding costs since the forwarding zone may be much larger than the geocasting region. Moreover, these algorithms do not guarantee message delivery [18].

The second class of algorithms reduces the high flooding cost by using restricted forwarding zones and intelligent flooding techniques [8, 11]. In intelligent flooding, a sub-set of nodes, called the *connected dominating set* (CDS), is selected to perform flooding. An important property of a CDS is that each node in the zone is either in the CDS or has a neighbor in the CDS. In dense networks, the size of a CDS is much smaller than the number of nodes in the forwarding zone, and therefore, the number of required transmissions is reduced. Even in a connected network, however, these algorithms do not ensure the delivery of messages [18].

In the third class, a geocasting is divided into two phases: location-based unicasting and restricted flooding. In the first phase, location-based routing is used to route a message from a source node to a node in the geocasting region. In the second phase, restricted (or intelligent) flooding is performed by the nodes in the region. Generally, this approach minimizes the total number of nodes involved in geocasting. There is, however, no guaranteed message delivery if the topology graph in the geocasting region is not connected.

Various algorithms that combine location-based unicasting and restricted flooding with face traversal, and also guarantee message delivery, have been proposed in literature [4, 17, 18].

The first algorithm, called Depth-First Face Tree Traversal (DFFTT), was presented in [4] and formalized later in [18]. In the first phase, DFFTT uses GFG (or other location-based routing) to deliver a geocasting message to a node in a geocasting region R. Then, a face tree covering all the faces that intersect with R is constructed. By traversing every node on the face tree, the message is delivered to all nodes in R.

The second algorithm, called Restricted Flooding with

Intersected Face Traversal (RFIFT), was proposed separately in [17, 18]. Since the algorithm in [18] is just an improved version of the algorithm in [17], we treat both of them as RFIFT. The first phase of RFIFT is identical to that of DFFTT. In the second phase, RFIFT performs restricted flooding within the geocasting region R and traverses all the faces intersecting R. Each face traversal is determined by a pair of nodes: an internal border node and an external border node. An *internal border node* is defined as a node located in R with a planar neighbor outside of R. Here, two nodes are *planar neighbors* if an edge connecting these two nodes belongs to the planarized network graph. Similarly, an *external border node* is a node outside R, but with a planar neighbor in R. In RFIFT, each internal border node performs face traversal by using the left-hand rule with respect to all of its planar neighbors that are external border nodes.

The third algorithm [18], namely Entrance Zone Multicasting-based Geocasting (EZMG), sub-divides the surrounding area of a geocasting region R into a set of entrance zones. An important property of entrance zones is that any message entering into R must pass through at least one node in an entrance zone. In EZMG, the source multicasts a message to all entrance zones. Each node in entrance zones receiving the message broadcasts the message, and all nodes in R that hear the message perform restricted flooding in R.

The preceding three algorithms guarantee message delivery. However, these algorithms incur high transmission costs. Due to the multicasts used to check the emptiness of entrance zones, EZMG has the highest transmission cost among these algorithms. The additional cost associated with face tree construction makes DFFTT having the second highest transmission cost. Even though RFIFT has the lowest transmission cost of the three, the cost is still high.

III. Terminology and Virtual Surrounding Face

In this section, we present a network model and propose the concept of Virtual Surrounding Face (VSF).

A. Preliminary

Unit Disk Graph (UDG): UDGs are generally accepted models of sensor and ad-hoc networks in which all nodes have an identical transmission range [4, 6, 7, 17, 18]. Let $G_U = (V, E_U)$ denote a UDG where V is a set of *nodes*, and E_U is a set of *edges*. The radius is treated as a unity (normalized to 1). An edge e_{uv} between nodes u and v exists *if and only if* (*iff*) the Euclidean distance between u and v is not larger than 1. For e_{uv}, u and v are called *UDG neighbors*.

Planar Graph and Gabriel Graph (GG): Face routing plays an important role in unicasting and geocasting with guaranteed message delivery. Face routing can only be applied on a *planar graph* which is defined as a graph with no two edges crossing one another. To planarize a UDG $G_U = (V, E_U)$, a deduced sub-graph of G_U, called a Gabriel graph (GG), is normally employed. A Gabriel graph on $G_U = (V, E_U)$ is defined as a graph $G_{GG} = (V, E_{GG})$ so that for each edge $e_{uv} \in$

E_U, $e_{uv} \in E_{GG}$ *iff* the circle with e_{uv} as a diameter does not contain any nodes other than u and v. For $e_{uv} \in E_{GG}$, u and v are called *Gabriel neighbors*. A localized algorithm to find G_{GG} has been presented in [4] with an important property of G_U and the G_{GG}: if G_U is connected, then G_{GG} is connected.

Border Nodes of Geocasting Regions: For a geocasting region R, let V_R be the set of nodes within R. For an edge $e_{uv} \in E_{GG}$ intersecting with the boundary of R, u is called *an external border node* if $u \notin V_R$, or u is called *an internal border node* if $u \in V_R$. An edge e_{uv} is called *a crossing edge* if $e_{uv} \in E_{GG}$, e_{uv} intersects with the boundary of R, and one of u and v is an external border node.
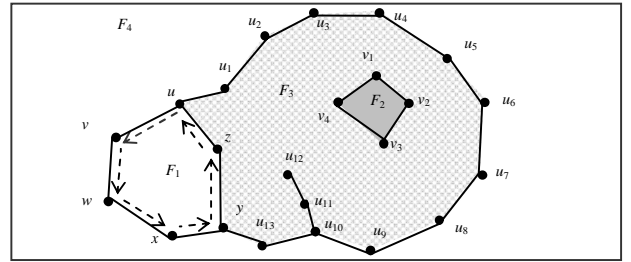

Figure 1 Face partition and traversal

Faces in Planar Graphs: The edges in a planar graph partition the network area into a set of faces [2, 3]. There are two types of faces: *interior faces* and *exterior faces*. An interior face is the continuous area bounded by one or more closed sequences of edges. An exterior face is the unbounded area outside the boundary of a network graph. For example in Figure 1, the network area is partitioned into four faces, F_1, F_2 (dark grey area), F_3 (light grey area), and F_4, where F_4 is an exterior face. It may also be noted that face F_3 is bounded by two sequences of edges: an *outer edge sequence* (or *outer boundary*) and an *inner edge sequence* (or *inner boundary*). The outer boundary is specified by the sequence of endpoints: $u \rightarrow u_1 \rightarrow u_2 \rightarrow \ldots \rightarrow u_{10} \rightarrow u_{11} \rightarrow u_{12} \rightarrow u_{11} \rightarrow u_{10} \rightarrow u_{13} \rightarrow y \rightarrow z \rightarrow u$. And, the inner boundary is: $v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_1$.

Face Traversal Rule: In VSFG, face traversal visits all nodes on the boundary of a face to guarantee delivery. We employ both the *Right-Hand Rule* [1, 3] and the *Left-Hand Rule* to traverse a face. In the former, a person explores a face by keeping her right hand on the walls (edges) and she will eventually visit all edges on the face. Similarly, in the later, a person explores a face by keeping her left hand on the walls.

To precisely specify face traversal, we define a face traversal method as follows and illustrated in Figure 1. Starting from u, to traverse F_1 by the Right-Hand Rule, u will send a traversal message to v in the form of *trav(source, destination, rule)*, where the source is the message sender, the destination is the message recipient, and the rule is either Right- or Left-Hand Rule. For node u, the message is *trav(u, v, Right)*. When v receives this message, v sends the message *trav(v, w, Right)* to node w. Repeated applying of these steps allows the message to traverse F_1 counterclockwise. Similarly, u can use *trav(u, z, Left)* to traverse F_1 clockwise.

In face traversal, some nodes may be visited more than once, which occurs when a face contains a dead-end. A *dead-end* of a face is a sub-path such that entering and exiting the sub-path can only be done through the same node. For example, node u_{10} in Figure 1 is an entering node of a dead-end $u_{10} \rightarrow u_{11} \rightarrow u_{12}$. To traverse face F_3, the traversal path is: $\ldots \rightarrow u_9 \rightarrow u_{10} \rightarrow u_{11} \rightarrow u_{12} \rightarrow u_{11} \rightarrow u_{10} \rightarrow u_{13} \rightarrow \ldots$, in which u_{10} and u_{11} are each visited twice.

B. Virtual Surround Face

For any two faces that share one edge, if the shared edge is ignored, the two faces are merged into one face with a larger area. For a geocasting region R, if we repeatedly merge all faces intersecting with R, we will eventually find a face large enough such that the boundary of the face contains R. This face is called a *virtual surrounding face* (VSF) of R. The objective of defining a VSF is as follows. To deliver a message to all the nodes in R, the message can be sent to one node on the boundary of the VSF. The message traverses the boundary of the VSF and each internal border node hearing the traversal message performs restricted flooding within R. Then all the nodes in R will eventually receive the message.
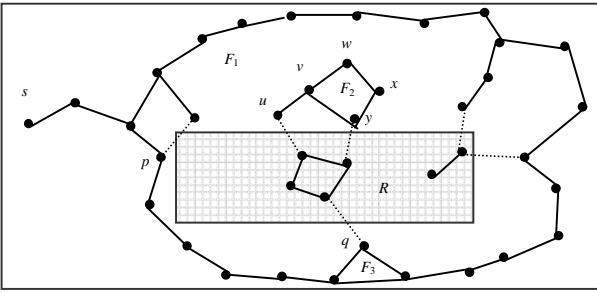

Figure 2 Interior virtual surrounding face F_1 of geocasting region R
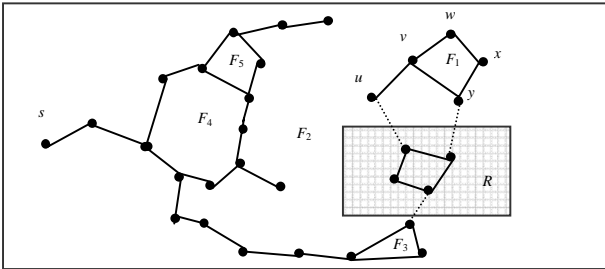

Figure 3 Exterior virtual surrounding face F_2 of geocasting region R

Consider the Gabriel graphs shown in Figures 2 and 3, where s is the source node of a geocasting task. If all crossing edges of R (dotted lines) are ignored, all nodes in V_R are disconnected from the rest of nodes in $V - V_R$, where "$-$" denotes the set difference operator. Therefore, the area immediately outside the boundary of R is continuous and belongs to one face, which is the VSF of R.

A node on the boundary of a VSF is called a *VSF node* and an edge on the VSF boundary is called a *VSF edge*. A VSF node which is an external border node is called a *VSF border node*. For a VSF edge e_{uv}, nodes u and v are called *VSF neighbors*. A VSF can be either an interior face (F_1 in Figure

2) or an exterior face (F_2 in Figure 3). Additionally, a VSF boundary may not be connected along VSF edges. For the VSF F_1 in Figure 2, the boundary of F_1 consists of not only the outer boundary, but the boundary specified by the sequence of endpoints $uvwxyvu$.

IV. DISTRIBUTED GEOCASTING ALGORITHM

Based on the concept of a VSF, we propose a VSF geocasting algorithm, VSFG, with guaranteed message delivery. VSFG consists of three components. (1) **VSF Forwarding**: A source node s transmits a geocasting message to a node u on the boundary of the VSF by using location-based routing; (2) **VSF Traversal**: Node u starts VSF traversal. In this task, each VSF node must transmit the geocasting message at least once; (3) **Restricted Flooding**: During VSF traversal, each node in R overhearing the traversal message for the first time performs restricted/intelligent flooding within R.

Let MSG(s, R, [*option*], *data*) denotes a geocasting message containing the source s and a geocasting region R. The *option* field contains the task-related information. Let $G_U = (V, E_U)$ represent a network and $G_{GG} = (V, E_{GG})$ be the Gabriel graph of G_U. Each node u in VSFG maintains the following local information. Let $N_U(u)$ denotes the set of UDG neighbors of u. u knows its own geographic location and the geographic locations of all nodes in $N_U(u)$. Node u also maintains the set of Gabriel neighbors, denoted by $N_{GG}(u)$, of u. In addition, we assume that all nodes involved in a geocasting task do not change their locations during the task.

A. VSF Forwarding

VSF forwarding uses location-based routing to deliver a message MSG(s, R, [*option*]) to a VSF node. Similar to the existing approaches [17, 18], we select a destination reference point r to guide VSF forwarding. The point r is chosen as the geographic point in R with the shortest distance to s. The forwarding message is MSG(s, R, r). Whether or not a node u is a VSF node is determined by the following proposition.

***Proposition* 1.** A node u is a VSF border node if u is an end point of a crossing edge of R.

Proof: This is proved by the construction process of VSF, where all faces intersecting with R are merged into a VSF by ignoring all crossing edges of R. When all the crossing edges of R connecting u are ignored, u must be located on the boundary of the constructed VSF.

In Figure 2, using Proposition 1, node p is selected to begin the VSF traversal of a message from the source s. VSF forwarding can be implemented by modifying a face routing or GFG-like routing algorithm [2, 3, 4, 5, 6, 7]. To illustrate the VSF forwarding process, we modify GFG [3]. The modified GFG algorithm given in Algorithm 1 guarantees to find a VSF node in a connected network. In VSF forwarding, note that in the Greedy Mode [3], a node chooses the next forwarding node from its UDG neighbors, and in the Face Mode, a node chooses its next node from its Gabriel neighbors.

Algorithm 1 VSF Forwarding

```
1:  Input:    MSG(s, R, r);     //the geocasting message
2:  BEGIN
3:      u ← s
4:      for (true)
5:          if (u has a crossing edge of R) then
6:              return u; // u is a VSF node of R; terminate.
7:          end if
8:          if (GFG fails to discover a node located at r) then
9:              return null; // s is disconnected from all nodes in R.
10:         end if
11:         u ← the next forwarding node of u towards r in GFG;
12:     end for
13: END
```

B. VSF Traversal

In VSF traversals, every VSF node must be visit at least once to guarantee message delivery in VSFG. All the VSF nodes may not be fully connected by VSF edges. When this situation occurs, the geocasting message must go through some nodes in the geocasting region to a disconnected component of the VSF. There are two cases associated with this situation, and VSF traversal handles them as follows.

Case 1: The VSF nodes are connected via a crossing edge of R that connects an internal border node and an external border node. Figure 4 illustrates this case, where VSF boundary $uvwx$ is connected to the outer face boundary via the crossing edge e_{tu} of R. When node u which does not receive a traversal message overhears the flooding message from t, u starts its own face traversal.

Case 2: The VSF nodes are connected via a crossing edge that connects two external border nodes. Figure 5 illustrates this case, where VSF boundary uvw is connected to the outer face boundary via e_{tu}, which is ignored during VSF construction. In this case, when u overhears the traversal message that is sent from node t and is designated to another node for the first time, if e_{tu} intersects R, u starts its own face traversal.
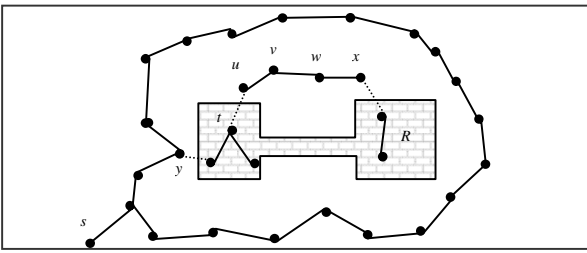

Figure 4 VSF boundary connected via a node in R

The VSF node selected during VSF forwarding stops VSF forwarding and starts VSF traversal. VSF traversals use a traversal message MSG(s, R, ini, $trav(…)$), where $trav(…)$ is the traversal method defined in Section III.A, and ini is the entrance node initiating this VSF traversal. A VSF node u is called *an entrance node* if u initiates a face traversal before u receives any traversal messages designated to u. It is possible for more nodes to become entrance nodes. For example, nodes u and y in Figure 4 are two entrance nodes.
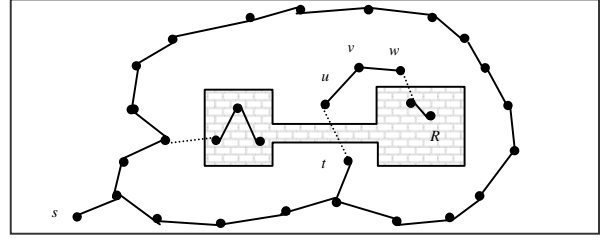

Figure 5 VSF boundary connected via a crossing edge of R

1) Initiation of VSF Traversal

To reduce traversal time and guarantee delivery, each entrance node simultaneously initiates a VSF traversal in two directions by using the Right-Hand and the Left-Hand Rule. Two possible starting cases are shown in Figures 6 and 7.

In Figure 6, the entrance node u has two VSF neighbors v_1 and w_1. Since u knows that it itself is a VSF node, u can find the next traversal node on the virtual surrounding face by ignoring the crossing edge e_{ux} of R. Then, u sends MSG(s, R, u, $trav(u, v_1, Left)$) to v_1 and MSG(s, R, u, $trav(u, w_1, Right)$) to w_1. When v_1 receives the traversal message, v_1 knows itself to be a VSF node and v_1 forwards MSG(s, R, u, $trav(v_1, v_2, Left)$) to v_2. Similar steps are repeated until the termination condition, to be given later in this section, is satisfied.

In Figure 7, the entrance node u has only one neighbor v which is a VSF node. In this case, u sends v a message MSG(s, R, u, $trav(u, v, Left$-$Right)$), where $Left$-$Right$ indicates to apply both the Left- and Right-Hand Rules. When v receives the message, since v has only one traversal node w, v modifies the message to MSG(s, R, u, $trav(v, w, Left$-$Right)$) and sends it to w. Once w receives the message, due to the Left-Right instruction in the message, and w having two VSF neighbors z and y, w sends two messages MSG(s, R, u, $trav(w, z, Left)$) and MSG(s, R, u, $trav(w, y, Right)$) to z and y, respectively.
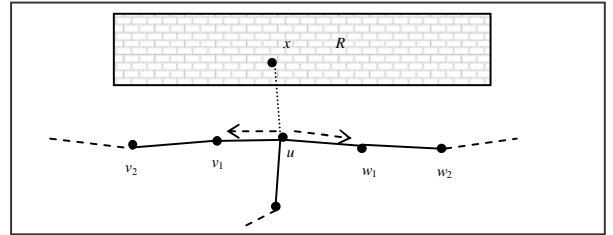

Figure 6 Case 1: two VSF neighbors of the entrance node u
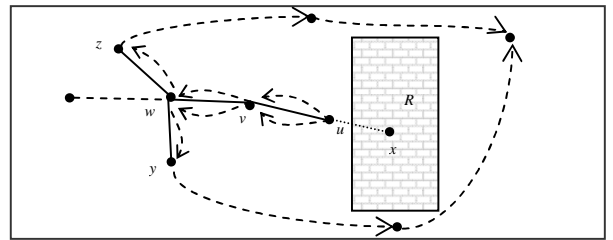

Figure 7 Case 2: one VSF neighbor of the entrance node u

2) Termination of VSF Traversal

To prevent from having the messages traverse a VSF many times, each VSF node uses a *termination condition*, which decides if the received traversal message can be discarded. This condition is stated as follows. Since an entrance node performs VSF traversal in two directions, intermediate VSF nodes may receive more than one traversal message. For two VSF neighbors u and v receiving two traversal messages, if the message received by u will traverse v next, and the message received by v will traverse u next, these two messages are discarded by u or v depending on which node transmits its message first. If u transmits its message to v first, then v discards both messages and stops its transmission to u. On the other hand, if v transmits first, u discards the messages.

Let the function *next*(q, MSG) return the node which will be traversed next when node q receives a traversal message MSG. For example in Figure 7, if w receives $MSG(s, R, u, trav(v, w, Left)$ from v, then $next(w, MSG(s, R, u, trav(v, w, Left))) = z$. The termination condition is as follows.

Termination Condition: For each VSF node u related to a geocasting task, let $M = \{MSG_1, MSG_2, \ldots, MSG_k\}$ denote the set of traversal messages that have been received by u but not been forwarded to other nodes by u yet. Once u receives a new message $MSG_{new}(s, R, w_{new}, trav(v_{new}, u, Rule_{new}))$, if there exists a message $MSG_j(s, R, w_j, trav(v_j, u, Rule_j)) \in M$ such that $next(u, MSG_{new}) = v_j$ and $next(u, MSG_j) = v_{new}$, u discards MSG_j and MSG_{new}, and stops the VSF traversal initiated by w_{new} and w_j. In addition, u removes MSG_j from M.

Algorithm 2 VSF traversal

```
1:  For each VSF node u;
2:  BEGIN
3:    state(u) ← IDLE; //initial state of the node u
4:    repeat
5:      if ((state(u) = = IDLE) && (receiving a message MSG )) then
6:        if ((MSG is a flooding MSG) && (the sender is a Gabriel
          neighbor of u)) then
7:          state(u) ← ENTRANCE;  // u is a entrance node
8:          u starts traversing in two directions;
9:        else if (MSG is a traversing message) then
10:          if ((the MSG recipient is not u) && (the edge from u to
            the MSG sender is a crossing edge of R)) then
11:            state(u) ← ENTRANCE;  // u is a entrance node
12:            u starts traversing in both direction;
13:          else if (the MSG recipient is u)
14:            state(u) ← TRVERSING;
15:            forward the MSG to the node v = next(u, MSG);
16:          end if
17:        end if
18:      else if ((state(u) ≠ IDLE) && (receiving a message MSG)) then
19:        if (Termination condition is satisfied by u) then
20:          if (u has not forwarded any traversing message and u is an
            external border node of R) then
21:            u broadcasts the content of MSG to its Gabriel
            neighbors;    // not forward MSG
22:          end if
23:          u discards the received message MSG;
24:        else
25:          u forwards the received MSG to node v = next(u, MSG);
26:        end if
27:      end if
28:    until (true)
29:  END
```

The detailed VSF traversal algorithm is given in Algorithm 2. In some situations of Algorithm 2, a node u may receive one or more MSGs which satisfy the termination condition before u forwards any traversal message. In Algorithm 2, u will discard all MSGs without further traversal. However, if u is an external border node of R, u must broadcast the contents of one of the received MSGs once. In this paper, the term "broadcast" means that a node transmits a message to all its neighbors, thus not flooding the network. The broadcasted message is called *VSF broadcasting* in the form of MSG(s, R, u, $data$), where u is the message sender. When an internal border nodes v of u receives this message for the first time, v performs VSF restricted flooding (Section IV.C). This special treatment is described in lines 19-22 of Algorithm 2.

C. Restricted Flooding

VSF restricted flooding has one requirement: all internal border nodes of R must broadcast the message once to ensure that the disconnected components of VSF can begin VSF traversal. Many existing flooding techniques can be modified to fulfill this requirement. In intelligent flooding, not all nodes in R need to broadcast, resulting in reduced message overheads. In this paper, we use the simplest restricted flooding. For each node in R which receives a geocasting message for the first time, the node broadcasts the message. Since all the duplicated messages received by a node in R are discarded, each node in R has exactly one transmission.

V. PERFORMANCE ANALYSIS

In this section, we prove guaranteed message delivery for VSFG, analyze the asymptotical bound of VSFG, and compare the performance of VSFG with existing approaches.

A. Guaranteed Message Delivery of VSFG

Guaranteed message delivery of VSFG is justified due to three VSF properties as follows.

***Property* 1**: For every node outside a VSF transmitting a message to a node in R, the message must go through at least one VSF node.

Proof: This property can be proved by contradiction. Assume that there is a path through which a message can be transmitted to a node in R without passing any VSF nodes. Then one edge on the path must cross one VSF edge, which contradicts the definition of a planar graph.

***Property* 2**: On a connected Gabriel graph, every node in R has a multi-hop path connecting to at least one VSF node. This property can be obtained immediately from Property 1.

***Property* 3**: let $\Psi = \{F_1, F_2, \ldots, F_k\}$ be the set of faces that intersect R on G_{GG} and let F_R be the VSF of R. Then $F_R = F_1 \cup F_2 \cup \ldots \cup F_k$, where $\cup$ is the union operator of two areas.

Proof: For each face $F_i \in \Psi$ and $k \geq 2$, there exists a face $F_j \in \Psi$ such that F_i and F_j share one edge e_{ij} intersecting with the boundary of R. Ignoring e_{ij} results in F_j and F_i being merged into one face. If this step is repeated unit all edges intersecting with the boundary of R are ignored, F_1, F_2, …, and F_k

eventually merge into one face. It may be noted that these steps are exactly the same as the steps used to build VSF F_R, and therefore, Property 3 holds.

According to Properties 1-3 and VSFG, the boundary nodes of all faces intersecting R are traversed, which is proved in [4] to be a sufficient condition to ensure message delivery.

B. Performance Analysis of VSFG

Similar to the existing approaches [4, 17, 18], the total cost C of VSFG is subdivided into two phases. (1) Unicasting phase: a geocasting task is delivered from a source to a node in the geocasting region. The cost C_u in this phase is measured by the number of transmissions required. (2) Face traversal with restricted flooding phase, which guarantees message delivery. Let C_r denote the cost (the total number of transmissions) associated with restricted flooding. Similarly, let C_f denote the face traversal cost which is measured by the total number of transmissions required to traverse all faces.

According to the preceding definitions, we have $C = C_u + C_f + C_r$. In the first phase, a location-based routing algorithm is modified to find the first entrance node. The best known algorithm is GOAFR$^+$ [7], which is both asymptotically worst case optimal and average case efficient. Using GOAFR$^+$, the total number of transmissions C_u required by VSFG to find an entrance node is bounded by $O(c^2)$, where c is the length of the shortest hop path from the source node to the entrance node.

In the second phase of restricted flooding, each node in the geocasting region broadcasts once, and therefore,

$$C_r = k, \tag{1}$$

where k is the total number of nodes in the region. We give the face traversal cost C_f in Proposition 2.

***Proposition* 2**: The total number of transmissions C_f required in VSF traversal is bounded by $C_t \leq 2n$, where n is the total number of VSF nodes.

Proof: According to VSF traversal, each VSF node might be visited once, twice, or more than twice. Figure 8 illustrates the first case in which all VSF nodes transmit the traversal message once and VSF traversal is terminated at node v. Figure 9 shows the second case in which some VSF nodes transmit the traversal message twice. This occurs if a VSF contains a dead-end. For example, node v on the path $v \rightarrow w \rightarrow x$ will be traversed twice. Hence, the complete traversal path is: $\ldots \rightarrow u \rightarrow \ldots \rightarrow v \rightarrow w \rightarrow x \rightarrow w \rightarrow v \rightarrow \ldots \rightarrow y \rightarrow \ldots$, where the nodes v and w are traversed twice.

Figure 10 illustrates the third case in which some VSF nodes transmit the traversal message more than twice. This situation occurs when a VSF node is the entering point of two or more dead-ends. For example, the node w is the entering point of the dead-end $w \rightarrow x \rightarrow y$ and $w \rightarrow z \rightarrow t$. The traversal path in Figure 10 is: $\ldots \rightarrow u \rightarrow v \rightarrow w \rightarrow x \rightarrow y \rightarrow x \rightarrow w \rightarrow z \rightarrow t \rightarrow z \rightarrow w \rightarrow s \rightarrow \ldots$, where the node w transmits three times. It is clear that the number of transmissions of w is equal to $j + 1$, where j is the total number of dead-ends related to w. However, for each dead-end, there exists a node which only transmits once, such as the nodes y and t in Figure 10. In addition, all the other nodes on a dead-end transmit twice, e.g. nodes x and z in Figure 10. Hence, on average, all nodes on the dead-end associated with w transmit at most twice.
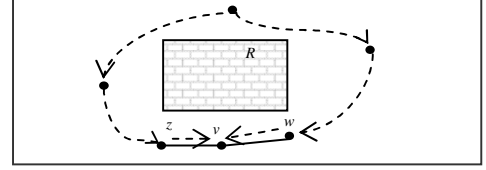

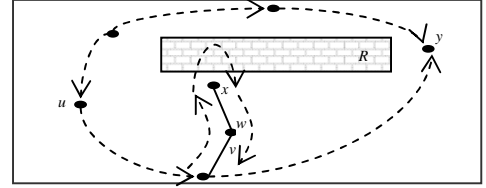Figure 8 VSF traversal: VSF nodes transmit once

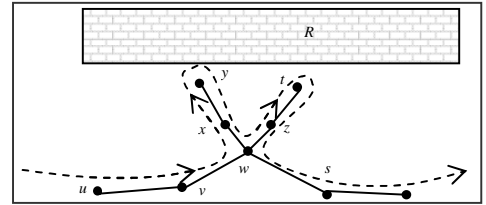Figure 9 VSF traversal: VSF node v transmits twice

Figure 10 VSF traversal: VSF node w transmits more than twice

Summarizing the three cases, on average, each VSF node transmits at most twice. Hence, we have $C_f \leq 2n$.

Combining the results of C_r and C_f, the total cost of VSFG in the second phase is given as follows.

***Proposition* 3**: The number of transmissions in VSFG in the second phase is upper bounded by $C_r + C_f \leq 2n + k \leq 2N$, where N is the number of connected nodes in the network.

Proof: According to the definition of VSF nodes, the set of VSF nodes and the set of nodes in R are disjoint. The total number of nodes in the VSF node set and the set of nodes in R is no more than the total number of nodes in the network. Hence, $n + k \leq N$. Combining the results in (1) and Proposition 2, we have $C_f + C_r \leq 2n + k \leq 2N$.

C. Performance Comparison

We compare VSFG with EZMG [18] and RFIFT [17, 18].

1) Comparison of VSFG and EZMG

The operations of EZMG are illustrated in Figure 11 based on a rectangular geocasting region R. In EZMG, the area surrounding R is partitioned into a set of entrance zones in two layers. As shown in Figure 11, each entrance zone is a square area (enclosed by dashed lines) with a width equal to the half the length of the transmission radius of a node. Hence, any message to be delivered into R must go through a node in an entrance zone. EZMG consists of two basic steps. First, EZMG multicasts a message toward the centers of all entrance zones by using a location-based routing. Second, all nodes in

the entrance zones receiving the message perform restricted flooding or intelligent flooding [8, 11] within R. In this way, EZBM guarantees message delivery. It may be noted that some entrance zones may contain no node. The emptiness of an entrance zone can be determined in EZMG using location-based routing with face traversal. As shown in Figure 11, the first step of EZMG is to construct a multicasting tree toward a set of destinations specified by all entrance zones. Figure 11 does not show the actual forwarding nodes, but shows the multicasting paths from a source s. In contrast, VSFG delivers the message via a single path. Obviously, VSFG uses a much smaller number of transmissions than EZMG in this step.

Moreover, as discussed in [18], the worst case of EZMG presents an excessive transmission cost related to the potential face traversal that is used to check the emptiness of entrance zones. The worst case scenario of EZMG occurs when only one entrance zone contains nodes and all other entrance zones are empty. As shown in Figure 12, only one entrance zone contains a node u and the solid curve outside R denotes the boundary of a face containing R, in which nodes on the face boundary are omitted. In this scenario, every empty entrance zone needs a face traversal to verify its emptiness. Since there is only one non-empty entrance zone, the face traversed for verifying emptiness of an entrance zone is roughly identical to VSF of R. Hence, the total number of transmissions required to verify all empty entrance zones is approximately equal to ($m -$ 1)n, where m is the number of zones, and n is the number of VSF nodes of R. Even though for a small R with a width and a height less than the radios range of a node, m is at least 4. For a large R, m can be very large. In contrast, VSFG requires at most $2n$ number of transmissions to verify the emptiness of entrance nodes in a similar scenario.

For the second step of EZMG, since VSFG can use the same restricted/intelligent flooding technique in EZMG, VSFG and EZMG require approximately identical transmissions. Thus, we conclude that VSFG performs superior to EZMG.
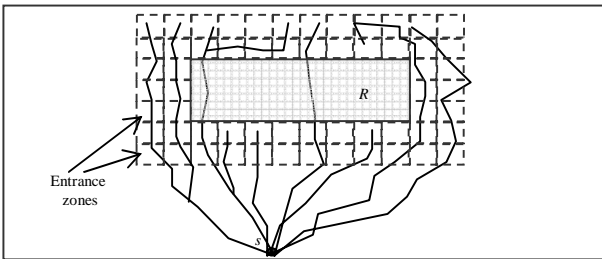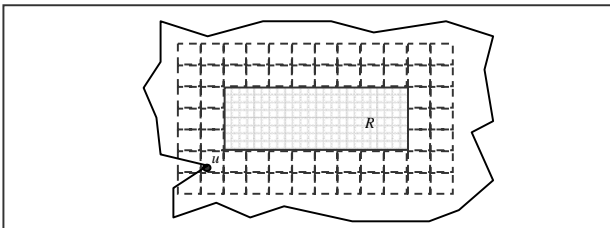
2) Comparison of VSFG and RFIFT

Since RFIFT and VSFG use two similar phases, we discuss the total number of transmissions involved in these two phases separately. In the first phase of location-based unicasting, it is fair to assume that RFIFT and VSFG use the same location-based routing algorithm. The following two conditions make VSFG slightly better than RFIFT in conserving the cost.

First, for a geocasting region R, RFIFT chooses the center point of R as the destination reference point. In contrast, VSFG uses a point in R with the shortest distance to the geocasting source as the reference point. Let d_{center} and d_{min} denote the distances from the source to the destination reference point in RFIFT and VSFG, respectively. Generally, the longer the distance between two nodes is, the longer the path between these two nodes. Since $d_{min} < d_{center}$, the path discovered in VSFG is shorter than that in RFIFT.

Second, for each region R, RFIFT terminates the destination searching when a node that is an internal border node of R is found. On the other hand, VSFG relaxes this condition to find a node that has an edge intersecting R, and it does not care if the other end point of the edge is located in R. Hence, VSFG will find the destination node by traversing a shorter path than RFIFT or at least at the same path as RFIFT does.

In the second phase, it is also fair to assume that RFIFT and VSFG use the same restricted flooding techniques in the geocasting region R. Referring to the results shown in [18], the total number of transmissions in this phase is constrained by $3n' + k \leq 3N$, where N is the number of nodes in the network, k is the number of nodes in R, and n' is the number of nodes that are on the faces intersecting R but not located within R. According to Property 3 in Section V.A, it is easy to show that $n < n'$, where n is the total number of VSF nodes in VSFG. Therefore, VSFG reduces the upper bound of the cost in this phase from $3n + k \leq 3N$ in RFIFT to $2n + k \leq 2N$.

VI. SIMULATION RESULTS

Since RFIFT is the most known efficient algorithm with guaranteed message delivery, we compare the performance of VSFG and RFIFT by using simulations. Due to the approximately identical costs C_u and C_r in VSFG and RFIFT, we do not show these two costs individually. Instead, we use the total cost C and the face traversal cost C_f as two performance metrics in the simulation. Two sets of experimental results are presented in various network topologies. In the first experiment, we compare VSFG and RFIFT in networks with randomly distributed nodes. In the second experiment, we compare these two algorithms in networks with randomly inserted voids, which are more realistic than random networks in practical applications. For simplicity, in all experiments, we use networks with stationary nodes. Under the assumption made in RFIFT in which the locations of VSF nodes and their neighbors involved in geocasting are not significantly changed during geocasting, VSFG can be used in mobile ad hoc networks too.

Figure 11 Entrance zones of a geocasting region R in EZMG

Figure 12 Worst case scenario of EZMG with many empty entrance zones

A. Simulation Results for Base Networks

The first experiment is done by using a routing-level simulator, based on a set of randomly generated networks. Nodes are randomly distributed throughout a 20×20 unit square area and the average degree (the average number of neighbors for all nodes) is g. We vary the value of g to observe the impact of the network density on the number of transmissions. All nodes have an identical transmission radius of 1 unit. These sample networks are called *base networks*, and for each fixed average degree, 10 base networks are generated and used. For each base network, we randomly generate 10 rectangular geocasting regions with width W and height H. We also vary the values of W and H to observe the impact of sizes of geocasting regions on the transmission costs. For simplicity, we omit the collisions involved in data transmissions.

Figures 13(a) and 13(b) show the costs C and C_f for geocasting regions with $W = 3$ and $H = 1.5$. In the following sections, the curve labeled with IFT denotes the result generated by using the RFIFT algorithm and the curve labeled with VSF denotes the results of VSFG. The x-axis denotes the average degree of sample networks. Similarly, Figures 14(a) and 14(b) show the costs C and C_f for geocasting regions with $W = 5$ and $H = 2.5$. According to the results shown in Figures 13 and 14, we have the following observations.
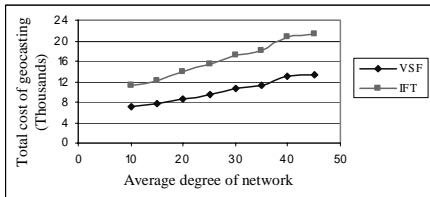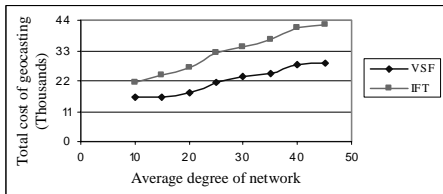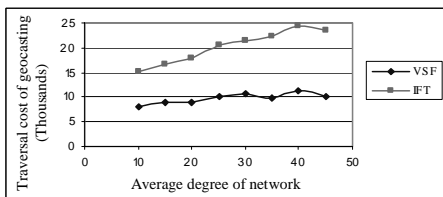

(a) Total cost of geocasting
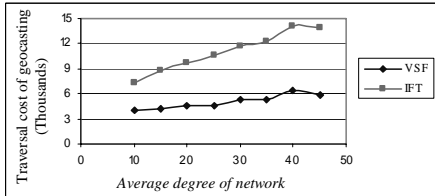
(b) Face traversal cost of geocasting

Figure 13 Costs for base networks with 3×1.5 geocasting regions

(a) Total cost of geocasting

(b) Face traversal cost of geocasting

Figure 14 Costs for base networks with 5×2.5 geocasting regions

First, VSFG reduces the total cost C of RFIFT by 25% to 40%. For fixed geocasting regions, when the network density increases, the reduction percentage of the total cost in VSFG decreases slightly comparing with RFIFT. This is because that the number of nodes in a region increases with the increase of network densities, resulting in an increase in the cost C_r of restricted flooding and C_r having a higher impact on the total cost. Since C_r in VSFG and RFIFT are identical, the reduction percentage of VSFG decreases compared with RFIFT.

Second, VSFG uses approximately 50% of transmissions for face traversal (C_f) compared with that of RFIFT. VSFC reduces more face traversal costs in higher density networks.
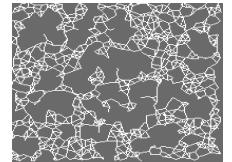
Third, when the size of geocasting regions increases, the reduction percentage of the total cost and the face traversal cost in VSFG decreases slightly. This is because for large regions, the cost C_r of restricted flooding has a higher impact on the total cost, and C_r in VSFG and RFIFT are identical.

B. Simulation Results for Void Networks

We then evaluate the performance of VSFG and RFIFT in networks with randomly inserted voids. For each base network, we randomly place a number of 1.5×1.5 square voids within the network area, and all the nodes in the voids are removed. The value of the void number is varied from 15 and 30. Figure 15 shows two graphs of sample void networks with 15 and 30 voids, generated from two base networks with $g = 10$. In practical applications, due to node mobility and the existence of obstacles, the networks shown in Figure 15 are more realistic than networks with uniformly distributed nodes.
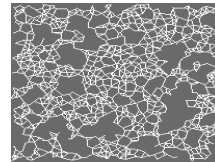
(a) Void network with 15 voids   (b) Void network with 30 voids

Figure 15 Void networks generated from base network with $g = 10$

Figures 16-19 plot the C and C_f of using VSFG and RFIFT in void networks. The curves labelled by VSF-C and IFT-C denote the total costs C for VSFG and RFIFT, respectively. Similarly, the curves labelled by VSF-Cf and IFT-Cf denote the costs C_f for VSFG and RFIFT, respectively. Based on the Figures 16-19, we have the following observations.

First, VSFG reduces 25% to 33% of the total cost involved in RFIFT. For a fixed geocasting region and a fixed void number, the reduction percentage of the total cost in VSFG decreases with the increase of network densities.

Second, for a fixed geocasting region and a fixed network density, the reduction percentage of the total cost decreases slightly when the number of voids in networks increases.

Third, VSFG uses approximately 50% to 60% of the face traversal cost C_f in RFIFT. The lower the network density, the higher the reduction percentage of C_f can be achieved. Comparing with RFIFT, VSFG can achieve a higher performance gain in base networks than in void networks.
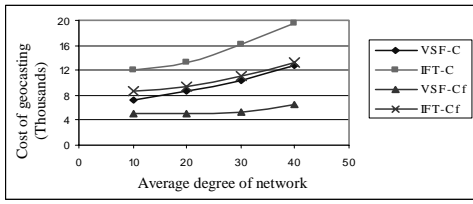
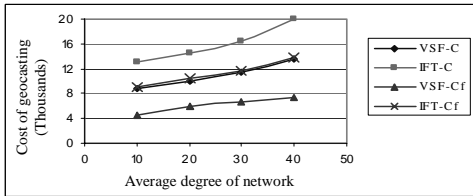Figure 16 Costs for void networks with 15 voids and 3×1.5 regions

Figure 17 Costs for void networks with 30 voids and 3×1.5 regions
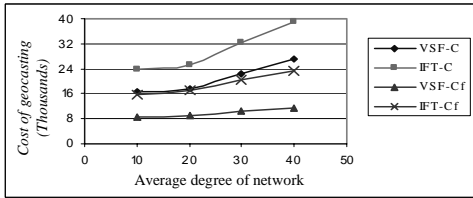

Figure 18 Costs for void networks with 15 voids and 5×2.5 regions
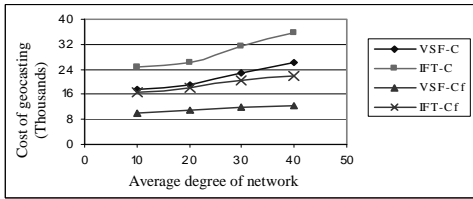

Figure 19 Costs for void networks with 30 voids and 5×2.5 regions

VII. CONCLUSIONS

Geocasting with guaranteed message delivery and low transmission cost has been extensively studied in literature [4, 9-18]. Three algorithms, namely Depth-First Face Tree Traversal (DFFTT) [4], Restricted Flooding with Intersected Face Traversal (RFIFT), and Entrance Zone Multicasting-based Geocasting (EZMG) [17, 18], guarantee message delivery. However, these algorithms are associated to high transmission costs.

In this paper, we present a geocasting algorithm VSFG with guaranteed message delivery and low transmission cost. In VSFG, a VSF of a geocasting region is constructed by ignoring the edges intersecting the geocasting region. By traversing all the boundary nodes of VSF and performing restricted flooding within the geocasting region, all nodes are guaranteed to receive the message.

We evaluate the proposed design VSFG through theoretical analyses and comprehensive simulations. Among all the existing algorithms, RFIFT has the lowest transmission cost, in which the cost for face traversal with restricted flooding is limited to $3n + k \leq 3N$. We show that our VSFG reduces this bound to $2n + k \leq 2N$. The simulation results also demonstrate that VSFG reduces up to 40% of the total transmissions required in RFIFT.

REFERENCES

[1] G.G. Finn, "Routing and addressing problems in large metropolitan inter-networks," ISI Research Report ISU/RR-87-180, 1987.

[2] E. Kranakis, H. Singh, and J. Urrutia, "Compass routing on geometric networks," In *Proc. Canadian Conference on Computational Geometry*, 1999, pp. 51–54.

[3] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, "Routing with Guaranteed Delivery in Ad Hoc Wireless Networks," in *Proc. ACM Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, 1999, pp. 48–55.

[4] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks," *Wireless Networks*, vol. 7, no. 6, pp. 609–616, 2001.

[5] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proc. ACM/IEEE MOBICOM*, 2000, pp. 243–254.

[6] F. Kuhn, R. Wattenhofer, and A. Zollinger, "Asymptotically optimal geometric mobile ad-hoc routing," in *Proc. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, *ACM Press*, 2002, pp.24–33.

[7] F. Kuhn, R. Wattenhofer, Y. Zhang, and A. Zollinger, "Geometric ad-hoc routing: of theory and practice," in *Proc. ACM Symposium on Principles of Distributed Computing*, 2003, pp. 63–72.

[8] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying for flooding broadcast messages in mobile wireless networks," in *Proc. Hawaii Intl. Conference on System Sciences*, 2002, pp. 3898–3907.

[9] Y. B. Ko and N. H. Vaidya, "Geocasting in mobile ad hoc networks: location-based multicast algorithms," in *Proc. Workshop on Mobile Computer System and Applications*, 1999, pp. 101–110.

[10] S. Basagni, I. Chlamtac, and V. R. Syrotiuk, "Geographic messaging in wireless ad hoc networks," in *Proc. VTC'99*, 1999, pp 1957–1961.

[11] W. H. Liao *et al.*, "Geogrid: a geocasting protocol for mobile ad hoc networks based on grid," *Journal of Internet Technology*, vol. 1, no. 2, pp. 23–32, 2000.

[12] I. Stojmenovic, A. P. Ruhil, and D. K. Lobiyal, "Voronoi Diagram and Convex Hull-Based Geocasting and Routing in Wireless Networks," in *Proc. IEEE Symposium on Computer and Communication*, 2001, pp. 51–56.

[13] Y. B. Ko and N. H. Vaidya, "Flooding-based geocasting protocols for mobile ad hoc networks," *Mobile Networks and Applications*, vol. 7, no. 6, pp. 471–480, 2002.

[14] C. Schwingenschlogl and T. Kosch, "Geocast enhancements of AODV for vehicular networks," *ACM Mobile Computing and Communication Review*, vol. 6, no. 3, pp. 96–97. 2002

[15] B. An and S. Papavassiliou, "Geomulticasting: architectures and protocols for mobile ad hoc networks," *Journal of Parallel and Distributed Computing*, vol.63, pp.182–195, 2003.

[16] T. Camp and Y. Liu, "An adaptive mesh-based protocol for geocast routing," *Journal of Parallel and Distributed Computing*, vol. 63, pp. 196–213, 2003.

[17] K. Seada and A. Helmy, "Efficient geocasting with perfect delivery in wireless networks," in *Proc. Wireless Communications and Networking (WCNC)*, 2004, pp. 2551–2556.

[18] I. Stojmenovic, "Geocasting with Guaranteed Delivery in Sensor Networks," *IEEE Wireless Communications,* vol.11(6), 2004, pp. 29–37.

[19] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T. Lai, "Deploying wireless sensors to achieve both coverage and connectivity," in *Proc. ACM MobiHoc*, 2006, pp. 131–142.