Three-Dimensional Greedy Anti-Void Routing for Wireless Sensor Networks

Wen-Jiunn Liu, *Student Member, IEEE,* and Kai-Ten Feng, *Member, IEEE*

***Abstract*—Due to the low-cost design nature of greedy-based routing algorithms, it is considered feasible to adopt this type of schemes within the three-dimensional (3D) wireless sensor networks. In the existing research work, the unreachability problem (i.e., the so-called void problem) resulting from the greedy routing algorithms has not been fully resolved, especially under the 3D environment. In this letter, a three-dimensional greedy anti-void routing (3D-GAR) protocol is proposed to solve the 3D void problem by exploiting the boundary finding technique for the unit ball graph (UBG). The proposed 3D rolling-ball UBG boundary traversal (3D-RUT) scheme is employed to guarantee the delivery of packets from the source to the destination node. The correctness proofs, protocol implementation, and performance evaluation for the proposed 3D-GAR protocol are also given in this letter.**

***Index Terms*—Greedy routing, void problem, unit ball graph, three-dimensional, wireless sensor network.**

I. Introduction

In recent years, three-dimensional (3D) routing has gained attention in the wireless sensor networks (WSNs). For example, the applications for underwater sensor networks have become more popular in the field of oceanographic engineering, including data collection, water monitoring, pollution control, and ocean surveillance. Previous work on the routing protocols for the 3D WSNs can be found in [1]. Due to the limited available resources, efficient design of localized routing protocols becomes a crucial subject within the 3D WSNs. How to guarantee delivery of packets is considered an important issue for the localized routing algorithms. The well-known greedy forwarding (GF) protocol [2] is proposed as a superior scheme with its low routing overheads and the adaptability to the 3D-routing environment. However, the unreachability problem (i.e., the so-called void problem [3]) occurring within the GF algorithm will fail to guarantee the delivery of data packets. In order to alleviate the void problem, the 3D-ABLAR protocol [4] employs the heuristic next-hop selection techniques that forward packets to additional two neighbor nodes located in separated regions so as to gain more chance to escape from the void. The projection from two-dimensional (2D) face routing to 3D space is also proposed in [5] as another technique to deal with the void problem.

Manuscript received February 12, 2009; revised June 28, 2009; accepted August 31, 2009. The associate editor coordinating the review of this letter and approving it for publication was F.-N. Pavlidou.

The authors are with the Department of Communication Engineering, National Chiao Tung University, Hsinchu, Taiwan (e-mail: jiunn.cm94g@nctu.edu.tw; ktfeng@mail.nctu.edu.tw).

This work was in part funded by the Aiming for the Top University and Elite Research Center Development Plan, NSC 96-2221-E-009-016, the MediaTek research center at National Chiao Tung University, and the Universal Scientific Industrial (USI) Co., Taiwan.

Digital Object Identifier 10.1109/TWC.2009.12.090220

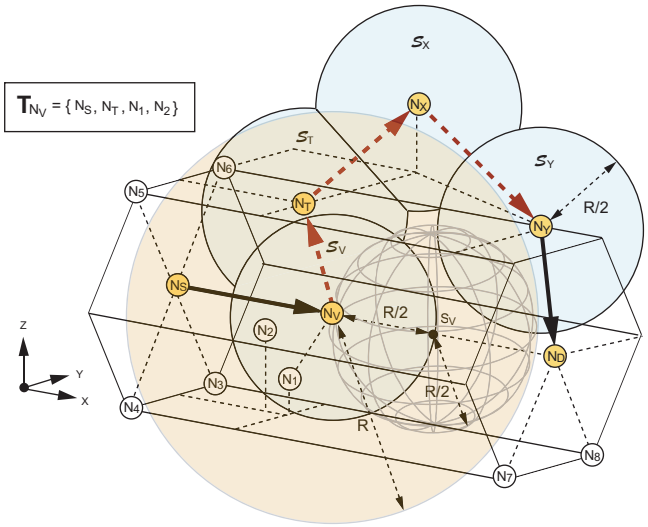


Fig. 1. The example routing path constructed by using the 3D-GAR algorithm: (N_S, N_D) is the transmission pair, and N_V is the void node. N_S, N_T, N_1, and N_2 are within the light-yellow transmission range $\overline{\Theta}(P_{N_V}, R)$ of N_V, constructing the one-hop neighbor table $\mathbf{T}_{N_V}$. There exists no SN in $\mathbf{T}_{N_V}$ whose distance to N_D is smaller than that of N_V to N_D, which results in the void problem while N_S intends to deliver the packets to N_V by adopting the GF scheme. The 3D-GAR algorithm changes its routing strategy into the 3D-RUT scheme, forming the rolling ball $RB_{N_V}(\boldsymbol{s}_V, R/2)$ and the resulting constrained SP surfaces. In order to clearly visualize the 3D diagram, only the surfaces $\mathcal{S}_V$, $\mathcal{S}_T$, $\mathcal{S}_X$, and $\mathcal{S}_Y$ are illustrated, i.e., the light-blue surfaces. After the surface traversal of $\mathcal{S}_V$, $\mathcal{S}_T$, $\mathcal{S}_X$, and $\mathcal{S}_Y$ associated with the corresponding boundary nodes based on the 3D-RUT scheme, the GF scheme can be resumed at N_Y whose distance is smaller than that of N_V to N_D. Therefore, the entire path resulting from the 3D-GAR protocol can be obtained as $\{N_S, N_V, N_T, N_X, N_Y, N_D\}$.

However, the void problem resulting from the GF algorithm has not been fully resolved under the 3D environment. In this letter, a 3D greedy anti-void routing (3D-GAR) protocol is proposed to solve the void problem under the unit ball graph (UBG) settings. The associated 3D rolling-ball UBG boundary traversal (3D-RUT) scheme is exploited within the 3D-GAR algorithm with the assurance for packet delivery. Moreover, the proofs of correctness, protocol implementation, and performance evaluation for the proposed algorithms are also described in the end of this letter.

II. Network Model and Problem Statement

Considering a set of SNs $\mathbf{N} = \{N_i \,|\, \forall\, i\}$ within a 3D Euclidean space $\mathbb{R}^3$, the locations of the set $\mathbf{N}$ are represented by the set $\mathbf{P} = \{P_{N_i} \,|\, P_{N_i} = (x_{N_i}, y_{N_i}, z_{N_i}), \forall i\}$, which can be acquired by their own positioning systems. The set of closed balls defining the transmission ranges of $\mathbf{N}$ is denoted as $\overline{\mathbf{\Theta}} = \{\overline{\Theta}(P_{N_i}, R) \,|\, \forall\, i\}$, where $\overline{\Theta}(P_{N_i}, R) = \{\boldsymbol{x} \,|\, \|\boldsymbol{x} - P_{N_i}\| \leq R, \forall\, \boldsymbol{x} \in \mathbb{R}^3\}$. It is noted that P_{N_i} is the

center of the closed ball with R denoted as the radius of the transmission range for each N_i. Furthermore, a unit ball graph (UBG) is defined as the intersection graph of a group of unit spheres in $\mathbb{R}^3$. Therefore, the network model for the 3D WSNs can be represented by a 3D UBG as $G(\mathbf{P}, \mathbf{E})$ with the edge set $\mathbf{E} = \{E_{ij} \,|\, E_{ij} = (P_{N_i}, P_{N_j}), P_{N_i} \in \overline{\Theta}(P_{N_j}, R), \forall\, i \neq j\}$. The edge E_{ij} indicates the unidirectional link from P_{N_i} to P_{N_j} whenever the position P_{N_i} is within the closed ball region $\overline{\Theta}(P_{N_j}, R)$. Moreover, the one-hop neighbor table for each N_i is defined as $\mathbf{T}_{N_i} = \{[ID_{N_k}, P_{N_k}] \,|\, P_{N_k} \in \overline{\Theta}(P_{N_i}, R), \forall\, k \neq i\}$, where ID_{N_k} represents the designated identification number for N_k. In the greedy forwarding (GF) algorithm, it is assumed that the source node N_S is aware of the location of the destination node N_D. If N_S wants to transmit packets to N_D, it will choose the next hop from its $\mathbf{T}_{N_S}$ which (a) has the shortest Euclidean distance to N_D among all the SNs in $\mathbf{T}_{N_S}$ and (b) is located closer to N_D compared to the distance between N_S and N_D. The same procedure will be performed by the intermediate nodes (e.g., N_V as in Fig. 1) until N_D is reached. However, the GF algorithm will be inclined to fail due to the occurrences of voids even though some routing paths exist from N_S to N_D. The void problem is defined as follows:

Problem 1 (Void Problem). *The greedy forwarding (GF) algorithm is exploited for packet delivery from N_S to N_D. The void problem occurs while there exists a void node (N_V) in the network such that*

$$\{P_{N_k} \,|\, d(P_{N_k}, P_{N_D}) < d(P_{N_V}, P_{N_D}), \forall\, P_{N_k} \in \mathbf{T}_{N_V}\} = \emptyset, \quad (1)$$

where $d(x, y)$ represents the Euclidean distance between x and y. $\mathbf{T}_{N_V}$ is the one-hop neighbor table of N_V.

III. Proposed 3D Greedy Anti-void Routing (3D-GAR) Protocol

The 3D-GAR protocol is a hybrid scheme consisting of both the GF algorithm and the 3D rolling-ball UBG boundary traversal (3D-RUT) scheme. The 3D-RUT algorithm is utilized to determine the boundary node set within the networks under the occurrence of void nodes. As the GF algorithm fails due to the void nodes, the 3D-RUT scheme can be utilized to escape from the void nodes by traversing the boundary node set and finally restart the GF forwarding process again. The packet delivery from N_S to N_D can therefore be guaranteed.

A. 3D Rolling-ball UBG Boundary Traversal (3D-RUT) Scheme

The 3D-RUT scheme is adopted to solve the boundary finding problem and acquire the so-called boundary node set (which will be defined later in this subsection) within the networks. The definition of boundary and the problem statement are described as follows.

Definition 1 (Boundary). *A boundary is defined as a closed surface that partitions the set of SNs $\mathbf{N}$ into two disconnected groups.*

Problem 2 (Boundary Finding Problem). *Given a UBG $G(\mathbf{P}, \mathbf{E})$ and the one-hop neighbor tables $\mathbf{T} = \{\mathbf{T}_{N_i} \,|\, \forall\, N_i \in \mathbf{N}\}$, how can a boundary be obtained by exploiting the distributed computing techniques?*

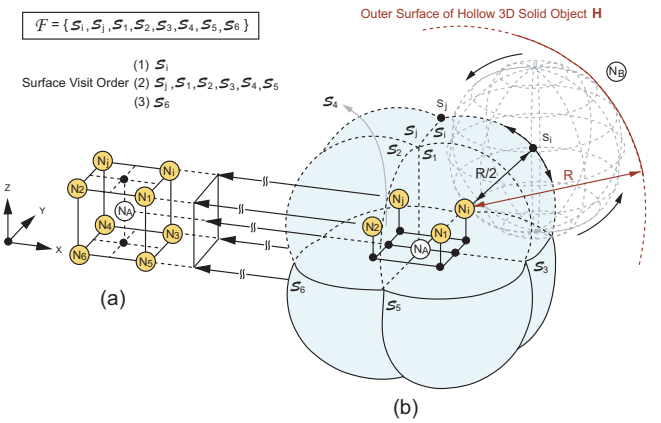


Fig. 2. The 3D rolling-ball UBG boundary traversal (3D-RUT) scheme: Fig. 2(a), the 3D scenograph of the internal part of Fig. 2(b), represents a cube with a node N_A at the center and $\{N_i, N_j, N_1, N_2, N_3, N_4, N_5, N_6\}$ at the eight corners. Given $\boldsymbol{s}_i$ and N_i, the 3D-RUT scheme freely rotates the rolling ball $RB_{N_i}(\boldsymbol{s}_i, R/2)$ hinged at N_i and constructs the eight constrained SP surfaces $\mathcal{S}_i$, $\mathcal{S}_j$, $\mathcal{S}_1$, $\mathcal{S}_2$, $\mathcal{S}_3$, $\mathcal{S}_4$, $\mathcal{S}_5$, and $\mathcal{S}_6$, which can be aggregated and considered as a boundary (i.e., the light-blue closed surface). The set $\mathbf{B} = \{N_i, N_j, N_1, N_2, N_3, N_4, N_5, N_6\}$ is established as a boundary node set by adopting the 3D-RUT scheme.

The concept of adopting the 3D-RUT scheme to resolve the boundary finding problem is briefly described as follows. Considering the cube formed by nine nodes as the vertices in Fig. 2(a), a 3D ball hinged at one of the vertex node with a radius of $R/2$ can be formed and freely rotated. It is noticed that the rolling ball is defined without any network node inside the ball. As in Fig. 2(b), it can be observed that the center points of the rotated balls can draw the closed blue surface, which is viewed as the boundary since this closed surface partitions the network into two disconnected parts, i.e., the nine nodes on the cube vertices as one network segment and node N_B as the other part. This type of rotated balls and their corresponding center points are formally defined as the rolling balls and the starting points (SPs) as follows:

Definition 2 (Rolling Ball). *Given $N_i \in \mathbf{N}$, a rolling ball $RB_{N_i}(\boldsymbol{s}_i, R/2)$ is defined as follows: (a) a closed ball hinged at P_{N_i} with its center point at $\boldsymbol{s}_i \in \mathbb{R}^3$ and the radius equal to $R/2$; and (b) there exists no node $N_k \in \mathbf{N}$ located inside the rolling ball.*

Definition 3 (Starting Point). *The starting point (SP) of N_i within the 3D-RUT scheme is defined as the center point $\boldsymbol{s}_i \in \mathbb{R}^3$ of $RB_{N_i}(\boldsymbol{s}_i, R/2)$.*

The detailed mechanism of the proposed 3D-RUT scheme is explained as follows. By means of computational geometry, each node N_i can verify if it has an SP or not by utilizing its one-hop neighbor table $\mathbf{T}_{N_i}$ since the rolling ball of N_i is always bounded by N_i's transmission range. Given $\boldsymbol{s}_i$ as an SP associated with its rolling ball $RB_{N_i}(\boldsymbol{s}_i, R/2)$ hinged at N_i, the rolling ball can freely rotate in all directions (i.e., 360°) within the 3D space as shown in Fig. 2. Based on the rotation of the rolling ball $RB_{N_i}(\boldsymbol{s}_i, R/2)$, an SP surface will be generated with the accumulation of the corresponding SPs.

However, according to Definition 2 that there should not exist any SN located inside the rolling ball, the resulting SP surface will become a constrained SP surface $\mathcal{S}_i$ since the rolling ball can be stuck by some of the SNs in the network. These SNs are denoted as the surface-adjacent nodes of N_i, i.e., N_j, N_1, N_2, N_3, N_4, and N_5 as shown in Fig. 2. It is noticed that these surface-adjacent nodes can be served as the next hopping nodes of N_i.

Subsequently, N_i will inform these surface-adjacent nodes to continue the 3D-RUT scheme by sending control packets that contain the information of their corresponding SPs in order to construct other constrained SP surfaces. For example, as shown in Fig. 2, the surface-adjacent node N_j can continue the 3D-RUT scheme by adopting the rolling ball $RB_{N_j}(\boldsymbol{s}_j, R/2)$ along with the newly assigned SP $\boldsymbol{s}_j$, which is located on the border of the constrained SP surface $\mathcal{S}_i$. Based on the same rotating procedure that is implemented by the rolling ball $RB_{N_j}(\boldsymbol{s}_j, R/2)$ hinged at N_j, the constrained SP surface $\mathcal{S}_j$ can therefore be constructed. Repeatedly, N_j's surface-adjacent nodes will be notified to continue the 3D-RUT scheme. As a result, all the constrained SP surfaces are established and identified as $\mathcal{S}_i$, $\mathcal{S}_j$, $\mathcal{S}_1$, $\mathcal{S}_2$, $\mathcal{S}_3$, $\mathcal{S}_4$, $\mathcal{S}_5$, and $\mathcal{S}_6$, which can be aggregated into a closed surface $\mathcal{F}$, i.e., the light-blue surface as in Fig. 2. For reader's clearness, the surface visit order starting from $\mathcal{S}_i$ is also summarized as in Fig. 2. This closed surface $\mathcal{F}$ is regarded as a boundary that is denoted in Definition 1; while the corresponding SNs that define the closed surface are represented as the elements in the boundary node set as follows:

Definition 4 (Boundary Node Set). *The boundary node set $\mathbf{B} \subseteq \mathbf{N}$ is defined as the SNs that construct the boundary based on the 3D-RUT scheme.*

Consequently, according to those eight constrained SP surfaces shown in Fig. 2, the boundary node set for this example can be obtained as $\mathbf{B} = \{N_i, N_j, N_1, N_2, N_3, N_4, N_5, N_6\}$. For preventing infinite recursion of the algorithm, each SN in $\mathbf{B}$ will only implement the 3D-RUT scheme once for the same boundary. Moreover, the reverse path of each SN in $\mathbf{B}$ to the original SN N_i can be obtained by referring to the previously visited constrained SP surfaces. For example, the surface traversing order can be obtained from $\mathcal{S}_i$, via $\mathcal{S}_j$, to $\mathcal{S}_6$ as in Fig. 2. Therefore, N_6 can construct the reverse path $N_6 \rightarrow N_j \rightarrow N_i$ in order to communicate with the original node N_i.

B. Detailed Descriptions of Proposed 3D-GAR Protocol

As shown in Fig. 1, the packets are intended to be delivered from N_S to N_D. N_S will select N_V as the next hop by adopting the GF algorithm. However, the void problem prohibits N_V to continue utilizing the same GF algorithm for packet forwarding. The 3D-RUT scheme is therefore employed by assigning an SP (i.e., $\boldsymbol{s}_V$) associated with the rolling ball $RB_{N_V}(\boldsymbol{s}_V, R/2)$ hinged at N_V. It is noticed that there should always exist an SP for each void node (N_V), which can be proved in Property 1 as follows:

Property 1. *There always exists an SP $\boldsymbol{s}_V$ for a void node N_V w.r.t. the destination node N_D.*

Proof: It is assumed that γ is denoted as the Euclidean distance between N_V and N_D. Based on the definition of a void node, there will not be any SN located inside the intersection area Υ of the two closed balls $\overline{\Theta}(P_{N_V}, R)$ and $\overline{\Theta}(P_{N_D}, \gamma)$. Considering a point $\boldsymbol{s}_V$ located on the connecting line between N_V and N_D with $R/2$ away from N_V, the closed ball $\overline{\Theta}(\boldsymbol{s}_V, R/2)$ will be situated inside the node-free intersection region Υ. According to Definitions 2 and 3, $\boldsymbol{s}_V$ will be an SP for N_V. It completes the proof. □

Based on Property 1, $\boldsymbol{s}_V$ can be chosen to locate on the connecting line between N_V and N_D with $R/2$ away from N_V as illustrated in Fig. 1. The corresponding constrained SP surfaces can be established by adopting the proposed 3D-RUT scheme. In order to clearly visualize the 3D diagram, only the constrained SP surfaces $\mathcal{S}_V$, $\mathcal{S}_T$, $\mathcal{S}_X$, and $\mathcal{S}_Y$ are depicted in Fig. 1 as the light-blue surfaces. Since N_T is one of the surface-adjacent nodes of N_V, N_T will be chosen by N_V as the next hopping node for continuing the 3D-RUT scheme. Due to the nature of the void node N_V, the distance from N_T to N_D should be not smaller than that from N_V to N_D, i.e., $d(P_{N_T}, P_{N_D}) \geq d(P_{N_V}, P_{N_D})$. Similar procedure will recursively be conducted by nodes N_T, N_X, and others. In the case that there exists a surface-adjacent node N_Y such that $d(P_{N_Y}, P_{N_D}) < d(P_{N_V}, P_{N_D})$, N_Y will inform N_V regarding the escape route from the void node based on the reverse path identified by the 3D-RUT scheme, e.g., $N_V \rightarrow N_T \rightarrow N_X \rightarrow N_Y$ as illustrated in Fig. 1. Consequently, the GF algorithm will be resumed at N_Y, and the route from N_S to N_D can therefore be constructed for packet delivery, e.g., $\{N_S, N_V, N_T, N_X, N_Y, N_D\}$ as in Fig. 1. Moreover, if there does not exist a node N_Y such that $d(P_{N_Y}, P_{N_D}) < d(P_{N_V}, P_{N_D})$, the 3D-RUT scheme will be terminated after completing the traversal of the boundary node set, e.g., the yellow nodes as depicted in Fig. 2. The result indicates that there is no routing path between N_S and N_D.

C. Proof of Correctness

Lemma 1. *A closed surface is established by all the SPs resulting from the 3D-RUT scheme.*

Proof: The relationship that "all the SPs within the 3D-RUT scheme form the surface of the resulting 3D solid object by overlapping the closed balls $\overline{\Theta}(P_{N_i}, R/2)$ for all $N_i \in \mathbf{N}$" is proven first. Based on Definitions 2 and 3, the set of SPs can be obtained as $\mathbf{S} = \mathbf{R_1} \cap \mathbf{R_2} = \{\boldsymbol{s}_i \mid \|\boldsymbol{s}_i - P_{N_i}\| = R/2, \exists N_i \in \mathbf{N}, \boldsymbol{s}_i \in \mathbb{R}^3\} \cap \{\boldsymbol{s}_j \mid \|\boldsymbol{s}_j - P_{N_j}\| \geq R/2, \forall N_j \in \mathbf{N}, \boldsymbol{s}_j \in \mathbb{R}^3\}$ by adopting the (a) and (b) rules within Definition 2. On the other hand, the surface of the resulting 3D solid object from the overlapped closed balls $\overline{\Theta}(P_{N_i}, R/2)$ for all $N_i \in \mathbf{N}$ can be denoted as $\mathbf{\Omega} = \mathbf{Q_1} - \mathbf{Q_2} = \bigcup_{N_i \in \mathbf{N}} K(P_{N_i}, R/2) - \bigcup_{N_i \in \mathbf{N}} \Theta(P_{N_i}, R/2)$, where $K(P_{N_i}, R/2)$ and $\Theta(P_{N_i}, R/2)$ represent the surface of a closed ball and the open ball centered at P_{N_i} with a radius of $R/2$ respectively. It is obvious to notice that $\mathbf{R_1} = \mathbf{Q_1}$ and $\mathbf{R_2} = \mathbf{Q_2'}$, which result in $\mathbf{S} = \mathbf{\Omega}$ and manifest the relationship in the beginning of this paragraph. Continuing the proof of this lemma, the surface of a 3D solid object will apparently result in a closed surface. Therefore, a closed surface is constructed by the combination of the SPs

resulting from the 3D-RUT scheme, e.g., the light-blue surface as in Fig. 2. It completes the proof of this lemma. □

Theorem 1. *The boundary finding problem (Problem 2) is resolved by the 3D-RUT scheme.*

Proof: Based on Lemma 1, a closed surface (denoted as $\mathcal{F}$) is constructed from the 3D-RUT scheme by rotating the rolling balls $RB_{N_i}(\boldsymbol{s}_i, R/2)$ hinged at P_{N_i} for all $N_i \in \mathbf{N}$. For example, as shown in Fig. 2, $\mathcal{F} = \{\mathcal{S}_i, \mathcal{S}_j, \mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3, \mathcal{S}_4, \mathcal{S}_5, \mathcal{S}_6\}$ is represented as the light-blue surface, and all SNs at which the corresponding rolling balls have been hinged are denoted by the set $\mathbf{U} = \{N_i, N_j, N_1, N_2, N_3, N_4, N_5, N_6\}$. Moreover, a hollow 3D solid object $\mathbf{H}$ is defined by the space that are traversed by those rolling balls, where the thickness of the object $\mathbf{H}$ will be equal to R since it is equivalent to the diameter of the rolling balls. The partial outer surface of $\mathbf{H}$ is illustrated as in Fig. 2(b). It is noticed that the closed surface $\mathcal{F}$ will become a layer of $\mathbf{H}$ situated at distance $R/2$ inward from the outer surface of $\mathbf{H}$.

Based on Definition 2, there is no SN located inside the rolling ball which consequently results in the case that there will be no SN within the 3D solid object $\mathbf{H}$. It can be observed that two disconnected regions can be derived as the inner and the outer spaces that are separated by $\mathbf{H}$ since, for all $N_A \in \mathbf{N}$ in the inner space, the smallest distance from N_A to N_B located in the outer space is greater than the SN's transmission range R. Consequently, the closed surface $\mathcal{F}$ situated within $\mathbf{H}$ can be considered as a boundary defined in Definition 1 that partitions $\mathbf{N}$ into two disconnected groups. As the example in Fig. 2(b), the set of nodes $\mathbf{U}$ and node N_A are located in the inner space; while node N_B is situated in the outer space. The set $\mathbf{U}$ can therefore be obtained as the boundary node set $\mathbf{B}$ based on Definition 4. It completes the proof. □

Theorem 2. *The void problem (Problem 1) is solved by the 3D-GAR protocol with guaranteed packet delivery.*

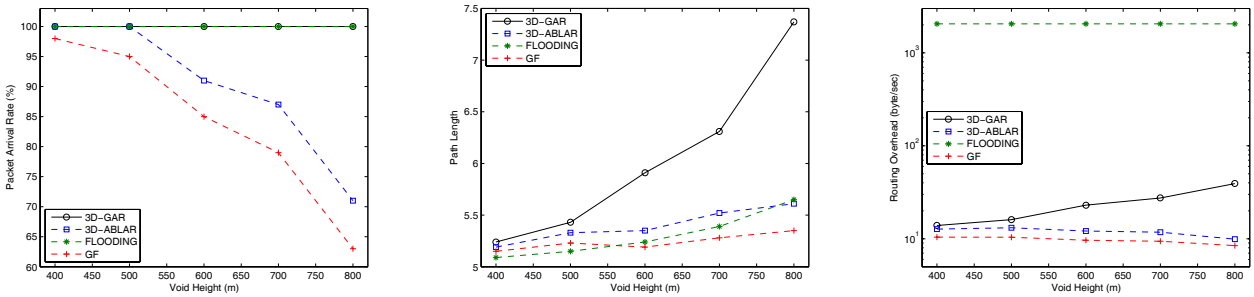
Proof: With the existence of the void problem occurring at any void node N_V, the 3D-RUT scheme is utilized by initiating an SP ($\boldsymbol{s}_V$) with the rolling ball $RB_{N_V}(\boldsymbol{s}_V, R/2)$ hinged at N_V. The 3D-RUT scheme within the 3D-GAR protocol will conduct boundary traversal via the associated boundary node set $\mathbf{B}$ under the condition that $d(P_{N_i}, P_{N_D}) \geq d(P_{N_V}, P_{N_D})$ for all $N_i \in \mathbf{B}$. If the boundary within the underlying network is completely traveled based on Theorem 1, it indicates that the SNs inside the boundary (e.g., N_V) are not capable of communicating with those located outside of the boundary (e.g., N_D). The result shows that there does not exist a route from the void node (N_V) to the destination node (N_D), i.e., the existence of network partition. On the other hand, if there exists a node N_Y such that $d(P_{N_Y}, P_{N_D}) < d(P_{N_V}, P_{N_D})$ (e.g., in Fig. 1), the GF algorithm will be adopted within the 3D-GAR protocol to conduct data delivery toward the destination node N_D. Therefore, the 3D-GAR protocol solves the void problem with guaranteed packet delivery, which completes the proof. □

IV. Protocol Implementation

After describing the design concept of the proposed 3D-GAR scheme, the implementation issues of the proposed protocol consisting of both the GF and the 3D-RUT algorithms are explained in this section. The GF scheme is considered a sequential table-lookup algorithm that only requires the implementation of the one-hop neighbor table. Therefore, both the time and space complexities are $O(m)$, where m represents the number of neighbors specified in the one-hop neighbor table. If the void problem occurs, the 3D-RUT scheme is utilized to forward packets to the nodes in the boundary node set $\mathbf{B}$ as defined in Definition 4. Since a node N_i's neighbors in the boundary node set can construct rolling balls with N_i, the original mechanism of forwarding packets to the nodes in $\mathbf{B}$ can therefore be transformed into a simple forwarding rule. In other words, node N_i which currently conducts the 3D-RUT scheme simply forwards packets to those neighbors that can form a rolling ball with N_i, where these neighbors can be obtained by the following method. For each pair of nodes (N_j, N_k) in the one-hop neighbor table of N_i, if a node-free-inside circumscribed ball hinged at N_i with a radius of $R/2$ can be established with both nodes N_j and N_k on the surface of the ball, the definition of the rolling ball (i.e., Definition 2) will be satisfied since the two conditions are satisfied as follows: (a) node N_i is located on the surface of the ball; and (b) there does not exist any neighbor node situated inside the ball. As a result, both N_j and N_k are considered as the next hopping nodes of N_i for packet forwarding. It is noted that the time complexity of this process is $O(m^3)$ since three nested loops to go through the one-hop neighbor table are required to conduct this procedure; while the space complexity is still $O(m)$ since only the construction of the neighbor table is required. Finally, by considering both the GF and the 3D-RUT schemes, the time and space complexities of the 3D-GAR protocol can be acquired as $O(m^3)$ and $O(m)$ respectively, where m is the number of neighbors specified in the one-hop neighbor table.

V. Performance Evaluation

The performance of the proposed 3D-GAR algorithm is evaluated and compared with other three protocols via simulations, including the 3D-ABLAR, the network flooding, and the reference GF algorithms. The simulation settings are explained as follows. A number of 1000 SNs are randomly deployed in the Euclidean 3D box ranging from (0, 0, 0) to (1000, 800, 1200) in the unit of meters. The transmission range of a node is 250 m. A pair of source and destination nodes are respectively located at (0, 400, 600) and (1000, 400, 600). The source node is with the data transmitting rate of 16 Kbps and data packet size of 512 bytes. There also exists a void block with length 400 m, width 800 m, and variable heights. This void block is randomly placed in the network in order to simulate the occurrence of void problems. In other words, there are SNs around the peripheral of the void block; while none of the nodes is situated inside the void block. Three performance metrics are utilized in the simulations for performance comparison as follows: (a) packet arrival rate: the ratio of the number of received data packets to the number of total data packets sent by the source; (b) path length: the average path length of successful routing in the unit of hop count; and (c) routing overhead: the average number of transmitted bytes per second for a network node.



(a) Packet arrival rate (%) (b) Path length for successful transmission (c) Routing overhead (byte/sec)

Fig. 3. Performance evaluation for the proposed 3D-GAR protocol.

Fig. 3(a) shows the packet arrival rate performance versus the void height. Due to the property of guaranteed delivery, both the 3D-GAR and network flooding algorithms will result in the delivery rate of 100%. The 3D-ABLAR and the GF protocols incur less delivery rate with regard to the increase of void height since the void problem occurs frequently when the void height becomes large. The 3D-ABLAR protocol has higher delivery rate than the GF algorithm owing to the reason that the GF scheme drops packet directly as the void problem occurs. Fig. 3(b) illustrates the performance of average path length for successful routing versus the void height. The network flooding algorithm results in the lowest value under small void heights since it can always find the shortest path between the source and the destination. However, as the void height becomes large, both the 3D-ABLAR and GF schemes will incur relatively smaller value of path length. The major reason is due to the 100% of packet delivery rate from the network flooding algorithm, which requires additional packet rerouting as the void problem becomes severe. The same reason can be applied to the curve obtained from the 3D-GAR protocol, which possesses a slightly longer routing path than the other protocols, e.g., around 1.8 additional hops under the void height of 800 m. Nevertheless, even with the slightly larger path length, the 3D-GAR protocol can result in guaranteed delivery rate as shown in Fig. 3(a), which outperforms both the 3D-ABLAR and GF schemes with lowered packet delivery rate.

Fig. 3(c) shows the performance of routing overhead versus the void height. It can be observed that both the 3D-ABLAR and GF schemes result in comparatively low overhead since most of the packets are dropped due to the void problem. In order to guarantee delivery and to find the shortest path, the network flooding algorithm generates a large number of packets in comparison with other protocols, which contributes to a significant amount of routing overhead as shown in Fig. 3(c). Comparing with the network flooding algorithm, the 3D-GAR protocol can achieve guaranteed packet delivery with a comparably smaller number of packets since the 3D-GAR scheme limits the packet rerouting only to nodes that are in the boundary node set. The merits of the proposed 3D-GAR protocol can therefore be observed, which achieves guaranteed packet delivery with reasonable routing overhead.

VI. Conclusion

In this letter, a three-dimensional greedy anti-void routing (3D-GAR) protocol is proposed to completely resolve the void problem incurred by the conventional greedy forwarding algorithm under the 3D environment. The 3D rolling-ball UBG boundary traversal (3D-RUT) scheme is adopted within the 3D-GAR protocol to solve the boundary finding problem, which results in the guarantee of packet delivery. In the end, the correctness proofs, protocol implementation, and performance evaluation of the proposed algorithms are properly provided.

References

[1] M. C. Domingo and R. Prior, “Energy analysis of routing protocols for underwater wireless sensor networks," *Computer Commun.*, vol. 31, no. 6, pp. 1227-1238, Apr. 2008.

[2] G. G. Finn, “Routing and addressing problems in large metropolitan-scale internetworks," *Inf. Sci. Inst. (ISI), Tech. Rep. ISI/RR-87-180*, Mar. 1987.

[3] B. Karp and H. T. Kung, “GPSR: greedy perimeter stateless routing for wireless networks," in *Proc. ACM/IEEE Int. Conf. Mobile Computing Networking*, Aug. 2000, pp. 243-254.

[4] A. E. Abdallah, T. Fevens, and J. Opatrny, “Hybrid position-based 3D routing algorithms with partial flooding," in *Proc. Canadian Conf. Electrical Computer Engineering*, May 2006, pp. 227-230.

[5] G. Kao, T. Fevens, and J. Opatrny, “3-D localized position-based routing with nearly certain delivery in mobile ad hoc networks," in *Proc. Int. Symp. Wireless Pervasive Computing*, Feb. 2007, pp. 344-349.