Virtual-Coordinate-Based Delivery-Guaranteed Routing Protocol in Wireless Sensor Networks with Unidirectional Links

Chia-Hung Lin, Bing-Hong Liu, Hong-Yen Yang, Chi-Yen Kao, and Ming-Jer Tsai
Department of Computer Science
National Tsing Hua University
Hsing Chu, Taiwan, ROC
E-mail: mjtsai@cs.nthu.edu.tw

Abstract—A wireless sensor network has unidirectional links because sensors can have different transmission ranges, sensors have unstable transmission ranges, and a hidden terminal problem exists. In this paper, we introduce a virtual coordinate assignment protocol (ABVCap_Uni) to assign virtual coordinates to nodes that have no geographic information in wireless sensor networks with unidirectional links, and we propose a routing protocol based on the ABVCap_Uni virtual coordinates. Our routing protocol guarantees packet delivery without computation and storage of global topology features in a discrete domain. Using simulation, we evaluate the performance of the proposed routing protocol (ABVCap_Uni routing), the greedy landmark-descent routing protocol (GLDR+VLM routing), and the greedy routing protocol based on physical coordinates (Euclidean routing). The simulations demonstrate that our routing protocol ensures moderate routing path length cost overhead.

I. INTRODUCTION

A wireless sensor network consists of several sensors, each of which has the ability to collect, process, and store environmental information as well as to communicate with others via inter-sensor communication. These characteristics allow wireless sensor networks to be used in a wide range of applications, including environmental monitoring, battlefield surveillance, health care, nuclear, biological, and chemical (NBC) attack detection, intruder detection, and so on. In wireless sensor networks, routing is an important issue and has received considerable attention. In this paper, we undertake the development of routing protocols in wireless sensor networks.

Geographic routing uses the physical location as the node address. GPSR [1] forwards the packet to a neighbor having a smaller Euclidean distance to the destination, and uses the FACE algorithm to forward the packet if no neighbor is closer to the destination. GSR [2] uses GPSR to route packets on a planar graph in which there exists a path whose distance, in terms of Euclidean distance or hop distance, is only the constant times the minimum distance between any pair of nodes. GPSR and GSR guarantee packet delivery [3]. In MGGR [4], all nodes are divided into Voronoi cells by landmarks, and each node is addressed by the physical location of the landmark in its residing cell. A packet is routed from the source cell to the destination cell by GPSR on a planar graph extracted from the Combinatorial Delaunay Graph (CDG) that represents the connections between neighboring cells. In geographic routing, nodes are required to have geographic information that is difficult to obtain because GPS devices are expensive and do not work indoors.

In many routing protocols, nodes are not required to obtain the geographic information. GLIDER [5] is a landmark-based routing protocol. Each node must memorize the minimum hop distances to all neighboring landmarks. A packet is routed via the shortest path from the source cell to the destination cell in the CDG identified by the source node. The efficiency of GLIDER depends on whether a good set of landmarks are selected. MAP [6] uses a Medial Axis Graph (MAG) as the guide for routing without selecting landmarks. In GLIDER and MAP, each node is required to compute and memorize global topology features. In GLDR+VLM [7], each node must obtain the minimum hop distances to all landmarks. Subsequently, GLIDER, MAP, and GLDR+VLM demand a considerable amount of message communication and memory overhead. VCap [8], GREENWIS [9], and ABVCap [10] rapidly assign virtual coordinates to nodes, and neither routing protocol requires global topology features to generate a routing decision. ABVCap routing guarantees packet delivery in a discrete domain, but VCap and GREENWIS routing does not. These routing protocols, except for GLDR+VLM, assume that all links are bidirectional.

In actual practice, a wireless sensor network has unidirectional links because sensors can have different transmission ranges [11], [12], [13], [14], sensors have unstable transmission ranges [4], [15], [16], [17], [18], and a hidden terminal problem exists [14], [19], [20], [21]. To the best of our knowledge, GLDR+VLM is the only method capable of being used in wireless sensor networks with unidirectional links. However, GLDR+VLM guarantees packet delivery only in a continuous domain. The remainder of this paper is organized as follows. Section II presents research related to ABVCap and GLDR+VLM routing. In Section III, we propose an Axis-Based Virtual Coordinate assignment protocol (ABVCap_Uni) to assign virtual coordinates to nodes in wireless sensor networks with unidirectional links. In Section IV, the routing pro-
tocol based on ABVCap Uni virtual coordinates is introduced that is shown to be delivery-guaranteed in the discrete domain in Section V. In Section VI, using simulations, we compare the performance of ABVCap Uni routing with Euclidean and GLDR+VLM routing. Finally, this paper concludes with a discussion of future research in Section VII.

II. PRELIMINARIES

Related works ABVCap and GLDR+VLM are presented and selected notations are introduced.

A. ABVCap

Each node is assumed to be static or quasi-static and to have a unique identifier (ID). All links are assumed to be bidirectional. Each node, $u$, is assigned at least one 5-tuple virtual coordinate $(u.lo, u.la, u.rp, u.up, u.dn)$ in a four-phase process, where $u.lo$, $u.la$, $u.rp$, $u.up$, and $u.dn$ denote the longitude, latitude, ripple, up, and down coordinates, respectively.

In phase one, four anchors $X$, $Y$, $Z$, and $Z'$ are selected. $W$ is the sink node, $X$ is the node having the maximum hop distance from $W$, $Y$ is the node having the maximum hop distance from $X$, $Z$ is the node in $S_{x=y\pm 1}$ having the maximum hop distance from $W$, and $Z'$ is the node in $S_{x=y\pm 1}$ having the maximum hop distance from $X$, where $S_{x=y\pm 1}$ is the set of nodes whose hop distances from $X$ and $Y$ each differ by one. In phase two, the shortest path is established from $Y$ to $X$, called the parallel of latitude, and for each node, $u$, in the parallel of latitude, a meridian, consisting of the shortest path from $u$ to $Z$ and the shortest path from $u$ to $Z'$, is established. Each node in the parallel of latitude is assigned the longitude coordinate to the hop distance from $X$, and the latitude and ripple coordinates to 0. Each node in the meridian established by $u$ is assigned the longitude coordinate to the longitude coordinate of $u$, the ripple coordinate to 0, and the latitude coordinate to the hop distance from $u$, if the node is in the meridian between $u$ and $Z$, and the negative value of the hop distance from $u$, if the node is in the meridian between $u$ and $Z'$. In phase three, each node, $u$, not in the meridian, is assigned the longitude and latitude coordinates to the longitude and latitude coordinates of the node in the meridian from which $u$ has the minimum hop distance, and the ripple coordinate to the hop distance from that node. In phase four, each node, $u$, having the longitude coordinate, $i$, is assigned the up (or down) coordinate to the minimum hop distance in the subnetwork induced by the nodes having the longitude coordinate, $i$, from a node that has a neighbor having the longitude coordinate, $i + 1$ (or $i - 1$).

B. GLDR+VLM

Landmarks, including virtual landmarks out of boundary, are selected such that each node can reach at least one landmark within $r$ hops. Each node is required to store the hop distances to all landmarks, and the hop distances from the $k$ nearest landmarks are used as the address of the node. Once a source routes a packet, the node selects, from the landmarks that address the destination, the landmark which maximizes the ratio of hop distances to the node and the destination. Subsequently, the packet is forwarded toward the chosen landmark along the shortest path until reaching a node whose hop distance to the chosen landmark is equal to the hop distance from the chosen landmark to the destination. At this time, the node chooses a landmark by repeating the same strategy. If, for each landmark that addresses the destination, the ratio of hop distances to the node and the destination is not greater than one, the packet is greedily forwarded to the closer neighbor, in terms of $L_1$ and $L_\infty$ norms, to the destination. If no neighbor is closer to the destination, the packet is forwarded by scoped flooding.

C. Notations

In this paper, $G_N$ denotes the Node Graph, $G_V$ denotes the Virtual Node Graph, and $G_E$ denotes the Extended Node Graph, as defined below.

**Definition 1.** $G_N = (V(G_N), E(G_N))$ is a directed graph, where $u \in V(G_N)$ denotes a sensor node in the network, and a directed edge $(u, v) \in E(G_N)$ indicates that $v$ can receive messages from $u$. Nodes $u$ and $v$ have a unidirectional link $(u, v)$ if $(u, v) \in E(G_N)$ and $(v, u) \notin E(G_N)$, and a bidirectional link $(u, v)$ if $(u, v) \in E(G_N)$ and $(v, u) \in E(G_N)$.

**Definition 2.** Given $G_N$, a cluster $CL$ is a maximal set of nodes such that there is a path in which all links are bidirectional between any two nodes in $CL$.

**Definition 3.** Given $G_N$, a directed cycle $C$ in $G_N$ is termed a ring if all nodes in $C$ are not in a cluster.

**Definition 4.** Given $G_N$ and a set, $RS$, of rings in $G_N$, $G_V$ is a directed graph obtained from $G_N$. For each node, $u$, not in a ring in $RS$, a virtual node, $u_1$, exists denoting $u$ in $G_V$. For each node, $u$, in $k$ rings, $R_1$, $R_2$, ..., $R_k$, in $RS$, $k$ virtual nodes, $u_1$, $u_2$, ..., $u_k$, exist denoting $u$ that are in rings, $R_{V_1}$, $R_{V_2}$, ..., $R_{V_k}$, respectively. In $G_V$, a bidirectional link exists between two virtual nodes, $u_i$ and $u_j$, denoting node $u$, and a bidirectional (or unidirectional) link exists between two virtual nodes, $u_i$ and $v_j$, denoting nodes $u$ and $v$, respectively, if $u$ and $v$ have a bidirectional (or unidirectional) link in $G_N$.

**Definition 5.** Given $G_N$, $G_V$, $RS$, and a set, $RS_V$, of rings in $G_V$, $G_E$ is an undirected graph obtained from $G_V$. For each virtual node, $u_1$, not in a ring in $RS_V$, a corresponding extended node, $u_E$, exists in $G_E$. For virtual nodes in ring $R_{V_1}$ in $RS_V$, a corresponding extended node, $a.b.E$, exists in $G_E$ if $a$ in $RS$ is the $b$-th ring initiated by node $a$ in $G_N$. Extended nodes $u_E$ and $v_E$ have a bidirectional link if a bidirectional link exists between some virtual nodes, $u_i$ and $v_j$, whose corresponding extended nodes are $u_E$ and $v_E$, respectively.

**Example 1.** Fig. 1 shows $G_N$. Nodes 16 and 33 are in a cluster, nodes 7 and 30 are in a cluster, and the other nodes are in a cluster. Directed cycle $(29, 33, 16)$ is a ring because nodes 16, 29, and 33 are not in a cluster. Let $R_1 = (29, 33, 16)$,
Fig. 2. Virtual Node Graph, $G_V$, and the assignment of the ring-irrelative coordinates of virtual nodes using ABVCap_Uni.

Fig. 3. Extended Node Graph, $G_E$.

$R_2 = (33, 16, 29)$, $R_3 = (7, 26, 5)$, $R_4 = (26, 5, 7)$, and $R_5 = (30, 7, 26, 5, 28)$ be the first rings initiated by nodes 29, 33, 7, 26, and 30 in $G_N$, respectively. Fig. 2 shows $G_V$, given $RS = \{R_1, R_2, R_3, R_4, R_5\}$. Node 16 is in $R_1$ and $R_2$, so it is denoted by virtual nodes 16$_1$ and 16$_2$. A bidirectional link (16$_1$, 16$_2$) exists because virtual nodes 16$_1$ and 16$_2$ denote the same node in $G_N$. A unidirectional link (16$_1$, 29$_2$) exists because there is a unidirectional link (16, 29). In $G_V$, $R_{V1} = (29, 33, 16, 11, 13, 7, 26, 5, 28, 30)$, $R_{V2} = (33, 16, 29, 7, 26, 5, 28)$. $R_{V3} = (7, 26, 5, 1)$, $R_{V4} = (26, 5, 2, 7, 2)$, and $R_{V5} = (30, 7, 3, 26, 5, 28)$. Fig. 3 shows $G_E$, given $RS_V = \{R_{V1}, R_{V2}, R_{V3}, R_{V4}, R_{V5}\}$. Because $R_1$ (or $R_2$) is the first ring initiated by node 29 (or 33) in $G_N$, the corresponding extended node of virtual nodes 29$_1$, 33$_1$, and 16 (or 33$_2$, 16$_2$, and 29$_2$) in $R_{V1}$ (or $R_{V2}$) is 29.1$_E$ (or 33.1$_E$). Extended nodes 29.1$_E$ and 33.1$_E$ have a bidirectional link because virtual nodes 29$_1$ and 29$_2$ have a bidirectional link.

III. ABVCAP\_UNI

Similarly to ABVCap, each node is assumed to be static or quasi-static and to have a unique ID. We also assume each node $u$ has the information of visible neighbor $v \in u.N_{OUT}$ (see Definition 6) [12]. Each node, $u$, is assigned at least one virtual coordinate, including eight entries: longitude ($u.l$), latitude ($u.la$), ripple ($u.rp$), up ($u.up$), down ($u.dn$), ring-initiator ($u.r initi$), ring-number ($u.r num$), and ring-order ($u.r ord$). We call ($u.r initi$, $u.r num$, $u.r ord$) the ring-relative coordinate of $u$, and ($u.l$, $u.la$, $u.rp$, $u.up$, $u.dn$) the ring-irrelative coordinate of $u$.

Given a strongly connected $G_N$, our concept is to obtain a connected $G_E$, then to assign each virtual node in $G_V$ the same ring-irrelative coordinates as that of the corresponding extended node in $G_E$ assigned by ABVCap, and finally, to assign node $u$ in $G_N$ the ring-irrelative coordinates of all virtual nodes in $G_V$ denoting $u$. Six phases exist in ABVCap\_Uni. The first two phases are used to construct a connected $G_E$, and the others to assign virtual nodes in $G_V$ the ring-irrelative coordinates in a manner analogous to ABVCap for the extended nodes in $G_E$. In phase one, clusters are constructed. In phase two, rings are established. Each virtual node is also assigned a ring-relative coordinate in this phase. In phase three, four anchors are selected. In phase four, axes, including a parallel of latitude and a number of meridians, are established. The longitude, latitude and ripple coordinates are also assigned to virtual nodes on axes in this phase. In phase five, the longitude, latitude and ripple coordinates are assigned to virtual nodes not existing on axes. In phase six, each virtual node is assigned the up and down coordinates. The following notations are used for the description of ABVCap\_Uni.

Definition 6. For two nodes, $u$ and $v$, in a graph, $G$, $v$ is in $u.N_{IO}$, $u.N_{IN}$, and $u.N_{OUT}$ if a bidirectional link $(u, v)$, a unidirectional link $(v, u)$, and a unidirectional link $(u, v)$ exist in $G$, respectively; $v$ is in the 1-hop bidirectional neighborhood of $u$ if $v$ is in $u.N_{IO}$.

Definition 7. Let $u_i$ be a virtual node in $G_V$ denoting node
u in \( G_N \). The virtual ID of \( u_i \) is equal to the ID of \( u \) if \( u \) is not in a ring, and is equal to the number \( a \cdot b \) if \( u \) is in a ring, where \( a = u_i \cdot r - \text{ini} \) and \( b = u_i \cdot r - \text{num} \).

**Definition 8.** The pair of numbers \( a \) and \( b \) is defined to be smaller than the pair of numbers \( c \) and \( d \), denoted by \((a, b) < (c, d)\), if \( a < c \), or \( a = c \) and \( b < d \).

**A. Construction of Clusters**

Each node, \( u \), having the minimum ID value in the 2-hop bidirectional neighborhood in \( G_N \), first generates and broadcasts a \( CL\_SET \) message containing the ID of \( u \). Once a node receives the \( CL\_SET \) message from a neighbor via a bidirectional link, the node broadcasts the message and assigns its Cluster-ID to the ID in the \( CL\_SET \) message that is generated by the node having the minimum ID.

**B. Establishment of Rings**

Each node, \( u \), having a neighbor, \( v \in u \cdot N_{IN} \), in a different cluster generates and forwards by scoped flooding a \( HC\_SET \) message containing the IDs of \( v \) and \( u \), and a hop counter initially set to 1, where the ring-number of \( u \) is not in a ring, and is equal to the number \( x \cdot \text{num} \), \( u \) becomes the predecessor of \( v \). Once node \( u \) receives a \( HC\_SET \) message, the node increases the hop counter in the message by one and broadcasts the updated \( HC\_SET \) message. If a node receives more than one \( HC\_SET \) message containing the same IDs of \( v \) and \( u \) and the ring-number of \( u \), the node broadcasts the message containing the smallest hop counter, which is generated by the virtual node having the maximum ID value in \( G_N \).

**D. Establishment of Axes**

The virtual node having the maximum ID value in a ring that has a virtual node in the 1-hop bidirectional neighborhood in \( G_V \) whose \( x \) (or \( z \)) coordinate is smaller by one, terms the \( x \) (or \( z \)) coordinate of the ring, is first selected. Subsequently, the parallel of latitude from anchor \( X \) to anchor \( Z \) is established. Anchor \( Y \) generates a \( PARA\_SET \) message, which is forwarded until received by anchor \( X \) or, in the case in which \( X \) is in a ring, all virtual nodes in the same ring as \( X \). Each virtual node that receives the \( PARA\_SET \) message is located on the parallel of latitude, and is assigned the longitude coordinate to its \( x \) coordinate and the latitude and ripple coordinates to 0. Virtual node \( u_i \) on the parallel of latitude having the longitude coordinate \( k \) establishes the \( k \)-th meridian by generating a \( MERI_k + \_SET \) (or \( MERI_k - \_SET \)) message if \( u_i \) is not in a ring or if \( u_i \) is the \( z \) (or \( z' \)) representative of the ring. The \( MERI_k + \_SET \) (or \( MERI_k - \_SET \)) message contains the \( z \) (or \( z' \)) coordinate of \( u_i \), which is forwarded until received by anchor \( Z \) (or \( Z' \)), or, in the case in which \( Z \) (or \( Z' \)) is in a ring, all virtual nodes in the same ring as \( Z \) (or \( Z' \)). Each virtual node that receives the \( MERI_k + \_SET \) (or \( MERI_k - \_SET \)) message is located on the \( k \)-th meridian, and the longitude coordinate is assigned to \( k \), the ripple coordinate to 0, and the latitude coordinate to the value (or the negative value) of the \( z \) (or \( z' \)) coordinate of the virtual node that generates the message minus its \( z \) (or \( z' \)) coordinate. Virtual node \( u_i \) deals with the \( PARA\_SET \) (or \( MERI_k + \_SET \), \( MERI_k - \_SET \)) message as follows:
1) if \( u_i \) is in a ring, \( u_i \) forwards the message to the successor;
2) if \( u_i \) is a \((x, z')\) representative of a ring, \( u_i \) forwards the message to a virtual node in the 1-hop bidirectional neighborhood in \( G_V \) whose \((x, z')\) coordinate is smaller by one; or,
3) if \( u_i \) is not in a ring, \( u_i \) forwards the message to a virtual node whose \((x, z')\) coordinate is smaller by one.

**E. Assignment of Longitude, Latitude, and Ripple Coordinates**

Every virtual node, \( u_i \), existing on axes first generates and broadcasts a \( \text{COOR}_\text{SET} \) message containing the longitude and latitude coordinates of \( u_i \) and a hop counter initially set to 0. Once a virtual node receives a \( \text{COOR}_\text{SET} \) message that contains the smallest hop counter, it first assigns the longitude and latitude coordinates to the longitude and latitude coordinates of the virtual node that generates the message, and then updates the message and assigns the ripple coordinate in a manner analogous to that for the \( \hat{W}_\text{SET} \) message and the \( w \) coordinate.

**F. Assignment of Up and Down Coordinates**

Every virtual node, \( u_i \), that has a virtual node in the 1-hop bidirectional neighborhood in \( G_V \) whose longitude coordinate is larger (or smaller) by one, assigns the up (or down) coordinate to 0 and generates and broadcasts an \( \text{UP}_\text{SET} \) (or \( \text{DOWN}_\text{SET} \)) message containing a hop counter initially set to 0. Once a virtual node receives an \( \text{UP}_\text{SET} \) (or \( \text{DOWN}_\text{SET} \)) message containing the smallest hop counter from the neighbor having the same longitude coordinate, it updates the message and assigns the up (or down) coordinate in a manner analogous to that for the \( \hat{W}_\text{SET} \) message and the \( w \) coordinate. In addition, if \( u_i,lo = \hat{Y}.lo \) (or \( u_i,lo = 0 \)), the virtual node, \( u_i \), assigns the up (or down) coordinate to 0, where \( \hat{Y}.lo \) is the longitude coordinate of \( \hat{Y} \). A virtual node, \( u_i \), knows whether \( u_i,lo = \hat{Y}.lo \) by the same method used in ABVCap.

**Example 2.** An example of the assignment of the virtual coordinates to nodes using ABVCap_\text{Uni} is presented in Fig. 1. In phase one, node 17 assigns its Cluster-ID to 1 because the node receives the \( \text{CL}_\text{SET} \) message generated by node 1. In phase two, because node 16 is in \( 29.N_{IN} \) and in different cluster from that of node 29, node 29 generates a \( \text{HC}_\text{SET} \) message that contains the IDs of nodes 16 and 29, the ring-number of node 29 which is equal to 1 because this is the first ring node 29 initiates, and a hop counter equal to 1. After node 16 receives the \( \text{HC}_\text{SET} \) message from node 33, node 16 sets the ring-relative coordinate to \((29, 1, 2)\) and the predecessor to node 33. Subsequently, node 33 sets the ring-relative coordinate to \((29, 1, 1)\) and the predecessor to node 29. Finally, node 29 sets the ring-relative coordinate to \((29, 1, 0)\). This establishes the ring \((29, 33, 16)\), denoted by \( R_1 \). Similarly, rings \( R_2 = (33, 16, 29) \), \( R_3 = (7, 26, 5) \), \( R_4 = (26, 5, 7) \), and \( R_5 = (30, 7, 26, 5, 28) \) are established. Node 16 is in two rings, \( R_1 \) and \( R_2 \), and is denoted by two virtual nodes, \( 16_1 \) and \( 16_2 \), as shown in Fig. 2, where \( R_{V1} = (29_1, 33_1, 16_1) \), \( R_{V2} = (33_2, 16_2, 29_2) \), and \( R_{V3} = (7_1, 26_1, 5_1) \), \( R_{V4} = (26_2, 5_2, 7_2) \), and \( R_{V5} = (30_1, 7_3, 26_3, 5_3, 28_1) \).

In phase three, virtual node 21 (the virtual node of the sink node) generates a \( \hat{W}_\text{SET} \) message with a hop counter equal to 0. After virtual node 35_1 receives the \( \hat{W}_\text{SET} \) message via the bidirectional link \((21, 35_1)\), virtual node 35_1 updates the hop counter to 1 and sets the \( w \) coordinate to 1. After virtual node 5_1 receives the \( \hat{W}_\text{SET} \) message via the bidirectional link \((35_1, 5_1)\), virtual node 5_1 updates the hop counter to 2 and sets the \( w \) coordinate to 2. Virtual node \( 7_1 \) sets the \( w \) coordinate to 2 after it receives the \( \hat{W}_\text{SET} \) message from virtual node 5_1 because virtual nodes 5_1 and \( 7_1 \) are in a ring. Virtual node 36_1 is selected as anchor \( \hat{X} \) because it has the maximum pair of the \( w \) coordinate and the virtual ID value. Virtual nodes \( 33_2 \), \( 24_1 \), and \( 23_1 \) are selected as anchors \( \hat{Y}, \hat{Z} \), and \( \hat{Z}' \), respectively, in a similar manner. In phase four, virtual node \( 33_2 \) establishes the parallel of latitude by generating a \( \text{PARA}_\text{SET} \) message that is forwarded to virtual node 16_2, a successor of virtual node 33_2 in ring \( R_{V2} \). Subsequently, the \( \text{PARA}_\text{SET} \) message is forwarded to virtual node 29_2, a successor of virtual node 16_2 in ring \( R_{V2} \). Virtual node 29_2 forwards the \( \text{PARA}_\text{SET} \) message to virtual node 27_1 because virtual node 29_2 is the \( x \) representative of ring \( R_{V2} \) and the \( x \) coordinate of virtual node 27_1 is smaller than that of virtual node 29_2 by one. The \( \text{PARA}_\text{SET} \) message is forwarded until virtual node 36_1 is reached. Virtual node 29_2, the \( z'(z') \) representative of ring \( R_{V2} \), establishes the 5th meridian by generating a \( \text{MERI}_{5+}_\text{SET} \) (or \( \text{MERI}_{5-}_\text{SET} \)) message. Subsequently, the \( \text{MERI}_{5+}_\text{SET} \) (or \( \text{MERI}_{5-}_\text{SET} \)) is forwarded to virtual node 19_1 (or \( 6_1 \)) whose \( z'(z') \) coordinate is smaller than that of virtual node 29_2. Once virtual node 19_1 (or \( 6_1 \)) receives the \( \text{MERI}_{5+}_\text{SET} \) (or \( \text{MERI}_{5-}_\text{SET} \)) message, it sets the longitude, latitude, and ripple coordinate to 5, \(-1\), and 1, respectively, where virtual node 19_1 (or \( 6_1 \)) is assigned the latitude coordinate 1 (or \(-1\)) because its \( z \) (or \( z' \)) coordinate is smaller than that of virtual node 29_2 by one. In phase five, virtual node 6_1 generates and broadcasts a \( \text{COOR}_\text{SET} \) message that contains a hop counter set to 0. Once virtual node 10_1 receives the \( \text{COOR}_\text{SET} \) message generated by virtual node 6_1, it sets the longitude, latitude, and ripple coordinate to 5, \(-1\), and 1, respectively. Finally, in phase six, virtual node 11 generates and broadcasts an \( \text{UP}_\text{SET} \) message because the longitude coordinate of virtual node 18_1, in the 1-hop bidirectional neighborhood of virtual node 11 in \( G_V \), is larger than that of virtual node 11 by one. Once virtual node 25_1 receives the \( \text{UP}_\text{SET} \) message generated by virtual node 11, it assigns the up coordinate to 1. Node 29 is assigned the ring-irrelative coordinates \((5, 0, 0, 0, 0)\) and \((5, 0, 0, 0, 0)\), which are the ring-irrelative coordinates of virtual nodes 29_1 and 29_2, respectively, as shown in Fig. 1.

**IV. ABVCap_\text{Uni Routing}**

Each node, \( u \), constructs a local routing table that contains routing information for nodes having the same longitude and latitude coordinates by obtaining the local routing table with neighbors in \( u.N_{IO} \cup u.N_{OUT} \) that also have the same longitude and latitude coordinates. We assume that a node, \( u \),
receives all multiple virtual coordinates of each neighbor in $u.N_{IO} \cup u.N_{OUT}$ and that the virtual coordinate of the destination received by the source is unique [22]. If $u.lo \neq d.lo$, longitude routing is used to forward the packet; otherwise, if $u.la \neq d.la$, latitude routing is used. Proactive routing is used if $u.lo = d.lo$ and $u.la = d.la$. Two phases are required for one node, $u$, to route the packet in longitude and latitude routing. In phase one, $u$ chooses a virtual coordinate for itself and for each neighbor $v$ in $u.N_{IO} \cup u.N_{OUT}$. In phase two, $u$ routes the packet based on the virtual coordinates chosen. The following notation is necessary for the description of ABVCapunami Routing.

Definition 9. Given the ring-irrelative coordinate of the destination $d$, $(d.lo, d.la, d.rep, d.up, d.dn)$, for the ring-irrelative coordinate of $u$, $(u.lo, u.la, u.rep, u.up, u.dn)$, the notation $u.rep$ denotes $u.up$ if $u.lo < d.lo$, denotes $u.dn$ if $u.lo > d.lo$, and denotes $|u.lo - d.la|$ if $u.lo = d.lo$.

A. Choice of Virtual Coordinates

Once node $u$ routes a packet, the one that has the smallest pair of numbers $(|u.lo - d.lo|, u.rep)$ is the virtual coordinate chosen in longitude routing, and the one that has the smallest pair of numbers $(|u.la - d.la|, u.rep)$ is the virtual coordinate chosen in latitude routing. Repeating the same strategy, $u$ chooses a virtual coordinate for each neighbor, $v$, in $u.N_{IO} \cup u.N_{OUT}$ that has multiple virtual coordinates.

B. Selection of Forwarding Node

In longitude routing, node $u$ forwards the packet to node $v \in u.N_{IO} \cup u.N_{OUT}$ whose chosen virtual coordinate has the smallest pair of numbers $(|v.lo - d.lo|, v.rep)$ if $(|v.lo - d.lo|, v.rep) < (|u.lo - d.lo|, u.rep)$ or if $v$ is the successor of $u$ in a ring. In latitude routing, $u$ forwards the packet to node $v$ whose chosen virtual coordinate has the smallest pair of numbers $(|v.la - d.la|, v.rep)$ if $v.lo = d.lo$ and $(|v.la - d.la|, v.rep) < (|u.la - d.la|, u.rep)$ or if $v.lo = d.lo$ and $v$ is the successor of $u$ in a ring.

Example 3. Consider the packet routed by ABVCapunami routing from node 22 to node 25 in Fig. 1. Node 22 proceeds with longitude routing. For node 22, $(22.lo - 25.lo, 22.rep)$ is equal to $(2,0)$, $(3,0)$, $(4,0)$, and $(5,0)$ if the virtual coordinate is $(2,-2,0,0,0)$, $(3,-2,0,0,0)$, $(4,-2,0,0,0)$, and $(5,-3,0,0,0)$, respectively; otherwise, $(2,-2,0,0,0)$ is the virtual coordinate chosen. Also, node 22 chooses $(0,-5,0,0,0)$, $(0,-4,0,0,0)$, and $(4,-1,0,0,0)$ as the virtual coordinates of nodes 23, 12, and 11, respectively. The packet is forwarded to node 12 because $(|12.lo - 25.lo|, 12.rep) = (0,0) \leq (|v.lo - 25.lo|, v.rep)$ for $v = 2, 3, 11, 17, 23$. After node 12 receives the packet, the node chooses $(0,-4,0,0,0)$ as its virtual coordinate and proceeds with longitude routing. Node 12 also chooses a virtual coordinate for each of nodes 22, 23, and 35. The packet is forwarded to node 26 because $(|26.la - 25.la|, 26.rep) = (1,1) \leq (|v.la - 25.la|, v.rep)$ for $v = 2, 22, 23, 35$. Similarly, the packet is forwarded to node 5, and then to node 20 using latitude routing. Finally, using proactive routing, the packet is forwarded to node 25.

V. ANALYSIS OF ABVCAPunami AND ITS ROUTING PROTOCOL

We first show that ABVCapunami assigns at least one 8-tuple virtual coordinate to each node. Secondly, we show that ABVCapunami routing guarantees packet delivery.

A. ABVCapunami Virtual Coordinate Assignment

Lemma 1. If $G_N$ is strongly connected, $G_E$ is connected.

Proof: Let $u_1$ and $v_2$ be two virtual nodes in $G_v$ denoting nodes $u$ and $v$ in $G_N$, and let $u_E$ and $v_E$ be the corresponding extended nodes of $u_1$ and $v_2$ in $G_E$, respectively. Suppose that $(u,v) \in E(G_N)$. It suffices to show that $u_E = v_E$ or that a path exists between $u_E$ and $v_E$ in $G_E$. Two cases are discussed: (c1) a ring exists that contains $u$ and $v$, and (c2) no ring contains both $u$ and $v$. For c1, if $u_1$ and $v_2$ are in a ring, $u_E = v_E$; otherwise, because a bidirectional link exists between any two virtual nodes denoting $u$ (or $v$), if $u_E \neq v_E$, a path exists between $u_E$ and $v_E$ in $G_E$. For c2, link $(u,v)$ can be (c2.1) bidirectional or (c2.2) unidirectional. For c2.1, a bidirectional link exists between $u_1$ and $v_2$, so $(u_E,v_E) \in E(G_E)$. For c2.2, $u$ and $v$ are in a cluster. A path exists in which all links are bidirectional between $u$ and $v$ in $G_N$; therefore, a path exists between $u_E$ and $v_E$ in $G_E$ by c1 and c2.1.

Lemma 2. Let $hc$ be a hop counter forwarded and incremented by virtual nodes in $G_v$ using ABVCapunami. Let $hc_{u_1}$ and $hc_{v_2}$ be the values of $hc$ updated by virtual nodes $u_1$ and $v_2$, respectively, and let $u_E$ and $v_E$ be the corresponding extended nodes of $u_1$ and $v_2$ in $G_E$, respectively. If $hc_{u_1}$ is updated by $v_2$ due to the receipt of $hc$ from $u_1$, $hc_{u_1} = hc_{v_2}$ if $u_E = v_E$, and $hc_{v_2} = hc_{u_1} = 1$ if $(u_E,v_E) \in E(G_E)$.

Proof: Note that $u_E = v_E$ if, and only if, $u_1$ and $v_2$ are in a ring, and $(u_E,v_E) \in E(G_E)$ if, and only if, there is a bidirectional link between some virtual nodes, $x_m$ and $y_n$, not in a ring whose corresponding extended nodes are $u_E$ and $v_E$, respectively. Because $v_2$ increases $hc$ by one if the message is forwarded via a bidirectional link not in a ring, $hc_{u_1} = hc_{v_2}$ if $u_E = v_E$ and $hc_{v_2} = hc_{u_1} = 1$ if $(u_E,v_E) \in E(G_E)$.

Lemma 3. Let $u_i$ be a virtual node in $G_v$ and $u_E$ be the corresponding node of $u_i$ in $G_E$. Then, the ring-irrelative coordinates of $u_i$ assigned by ABVCapunami are the same as that of $u_E$ assigned by ABVCap.

Proof: By Lemma 2, it is easy to verify that anchors $\hat{X}$, $\hat{Y}$, $\hat{Z}$, and $\hat{Z}'$ selected in $G_v$ by ABVCapunami correspond to anchors $X$, $Y$, $Z$ and $Z'$ selected in $G_E$ by ABVCap, respectively, and that the virtual nodes in the parallel of latitude (or i-th meridian) correspond to the extended nodes in the parallel of latitude (or i-th meridian). Also, virtual nodes are assigned the longitude, latitude, ripple, up, and down coordinates identical to the corresponding extended nodes assigned by ABVCap.
Theorem 1. If $G_N$ is strongly connected, ABVCap\_Uni assigns each node in $G_N$ at least one 8-tuple virtual coordinate.

Proof: It suffices to show that each virtual node is assigned at least one 8-tuple virtual coordinate by ABVCap\_Uni. First note that each virtual node is assigned exactly one ring-relative coordinate by ABVCap\_Uni. Next, if $G_N$ is strongly connected, $G_E$ is connected by Lemma 1. Using ABVCap, each extended node in $G_E$ is assigned at least one ring-relative coordinate by Theorem 1 of [10]. This implies that by using ABVCap\_Uni, each virtual node is assigned at least one ring-relative coordinate by Lemma 3.

B. Guaranteed Delivery of ABVCap\_Uni

Lemma 4. The nodes in $G_N$ having the same longitude and latitude coordinates induce a strongly connected subnetwork.

Proof: It suffices to show the virtual nodes in $G_V$ having the same longitude and latitude coordinates induce a strongly connected subnetwork. Let $u_i$ and $v_j$ be two virtual nodes in $G_V$ having the same longitude and latitude coordinates, and let $u_E$ and $v_E$ be the corresponding extended nodes of $u_i$ and $v_j$ in $G_E$, respectively. Let NET($u_i$) (or NET($v_j$)) denote the strongly connected subnetwork induced by the virtual nodes in $G_V$ having the same longitude and latitude coordinates, including $u_i$ (or $v_j$), respectively. By Lemma 3, ABVCap assigns $u_E$ and $v_E$ the same longitude and latitude coordinates. By Lemma 1 of [10], the extended nodes in $G_E$ having the same longitude and latitude coordinates induce a connected subnetwork. Thus, we must show that $v_j$ is in NET($u_i$) (c1) if $u_E = v_E$ or (c2) if $(u_E, v_E) \in E(G_E)$. For c1, $u_i$ and $v_j$ are in a ring; therefore, $v_j$ is in NET($u_i$). For c2, a bidirectional link exists between some virtual nodes, $x_m$ and $y_n$, whose corresponding extended nodes are $u_E$ and $v_E$, respectively. By c1, if $u_i \neq x_m$ (or $v_j \neq y_n$), $x_m$ (or $y_n$) is in NET($u_i$) (or NET($v_j$)). This implies that $v_j$ is in NET($u_i$).

Theorem 2. ABVCap\_Uni routing can always route the packet from source, $s$, to destination, $d$.

Proof: We first show that longitude routing can always forward the packet to a node, $v$, with $v.lo = d.lo$. Only the case in which $s.lo < d.lo$ is considered here. The case in which $s.lo > d.lo$ can be proved by a similar argument. Let $u^i$ be the $i$-th forwarding node. We claim for all $i$: 1) $[u^i.lo - d.lo, u^i + 1.rep] \preceq [u^i.lo - d.lo, u^i.rep]$ and 2) $[u^i + m.lo - d.lo, u^i + m.rep] \prec [u^i.lo - d.lo, u^i.rep]$ for some integer $m$. For claim 1, because $u^i$ selects $u^{i+1}$, that has the smallest pair of numbers $[v.lo - d.lo, v.rep]$, from all $v \in u^i.N_{IO} \cup u^i.N_{OUT}$, it suffices to show that there exists a $v \in u^i.N_{IO} \cup u^i.N_{OUT}$ such that $[v.lo - d.lo, v.rep] < [u^i.lo - d.lo, u^i.rep]$ or $v$ is the successor of $u^i$. If $u^i$ is in a ring, $u^i$ has a successor. Otherwise, if $u^i.up = 0$, $u^i$ has a neighbor $v \in u^i.N_{IO}$ that has a longitude coordinate $v.lo = u^i.lo + 1$, and if $u^i.up \neq 0$, $u^i$ has a neighbor $v \in u^i.N_{IO}$ that has a longitude coordinate $v.lo = u^i.lo$ and an up coordinate $v.up = u^i.up - 1$, in which cases $[v.lo - d.lo, v.rep] < [u^i.lo - d.lo, u^i.rep]$. For claim 2, in each ring, at least one node, $u$, has a neighbor $v \in u.N_{IO}$ such that $[v.lo - d.lo, v.rep] < [u.lo - d.lo, u.rep]$. Because $[u^{i+1}.lo - d.lo, u^{i+1}.rep] = [u^i.lo - d.lo, u^i.rep]$ only if $u^{i+1}$ is the successor of $u^i$ in a ring, integer $m$ is bounded by the length of the ring. Claims 1 and 2 imply that the packet is forwarded eventually to a node $v$ with $v.lo = d.lo$.

Next, we claim for all $i$: 3) $u^{i+1}.lo = u^i.lo$, $(u^{i+1}.lo - d.lo, u^{i+1}.rep) \preceq (u^i.lo - d.lo, u^i.rep)$ and 4) $(u^{i+m}.lo - d.lo, u^{i+m}.rep) \prec (u^i.lo - d.lo, u^i.rep)$ for some integer $m$, where $u^i$ denotes the $i$-th forwarding node. We omit the proofs of claims 3 and 4 due to their similarities with that of claims 1 and 2. Therefore, latitude routing can always forward the packet to a node, $v$, with $v.lo = d.lo$ and $v.lo = d.lo$.

Because nodes having the same longitude and latitude coordinates induce a strongly connected subnetwork by Lemma 4, proactive routing routes the packet to the destination.

VI. PERFORMANCE EVALUATION

In our simulation, network behaviors such as packet loss, packet delay, etc., were not taken into consideration. A network was generated by randomly deploying $n$ nodes in a $1000m \times 1000m$ square region. A link exists between two nodes with a distance not larger than $100m$. Each link is associated with a random number between 0 and 1. The link is bidirectional if the associated random number is not larger than $pb$; otherwise, it is unidirectional. The networks that are not strongly connected are discarded. We also generated networks with obstacles, where the obstacles were circles of radius $100m$ and could be overlapped. In the simulation, the percentage of bidirectional links, $pb$, was 0.6, 0.8, or 1; the number of obstacles, $ob$, was 0, 15, or 30; and, the number of nodes, $n$, was 300, 400, 500, 600, or 700. To evaluate the performance of routing protocols, we investigated the average delivery rate and the average path length of ABVCap\_Uni routing, GLDR+VLM routing, and Euclidean routing. Empirical data were obtained by averaging the data of 500 source-destination pairs from 200 networks. Parameters $r$ and $k$ in GLDR+VLM routing were set to 10, the same as in reference [7]. The maximum number of allowed local detours per packet, $c$, was set to 2 in Euclidean routing [8]. To evaluate the communication and memory overhead of ABVCap\_Uni, we examined the average number of virtual coordinates, the average hop distance from a visible neighbor, and the average number of broadcasts per node. Empirical data were obtained by averaging the data of 200 networks.

A. Delivery Rate

Fig. 4 shows the simulation results for the average delivery rate. ABVCap\_Uni routing successfully sets a path for every source-destination pair. With GLDR+VLM and Euclidean routing, the greater the number of nodes (the higher-density network), the higher the delivery rate. This observation results because fewer dead-end nodes exist in a higher-density network. The higher the percentage of bidirectional links, the higher the delivery rate because more paths exist between
every source-destination pair. The number of obstacles appears to have a more significant impact on the delivery rate with Euclidean routing than with GLDR+VLM routing. With GLDR+VLM routing, the more obstacles exist, the higher is the delivery rate, because more landmarks are selected. With Euclidean routing, however, the more obstacles exist, the lower is the delivery rate, because more dead-end nodes exist in the network. The delivery rate is about 69 – 87% with GLDR+VLM routing, and 68 – 99% with Euclidean routing.

B. Routing Path Length

Fig. 5 illustrates the simulation results for the average length of routing paths successfully set for source-destination pairs. The number of obstacles appears to have a more significant impact on the routing path length, as compared to the number of nodes and the percentage of bidirectional links. With ABVCap_Uni and GLDR+VLM, as we expect, the fewer obstacles exist, the shorter is the routing path. With Euclidean routing, the more obstacles exist, the shorter is the routing path. The observation results from that as the more obstacles exist in networks, Euclidean routing has a lower probability of setting a path for the source-destination pair having a long distance and thus has the smaller average distance of source-destination pairs. ABVCap_Uni routing path is longer than GLDR+VLM routing path by about 0 – 12% because GLDR+VLM routing has the smaller average distance of source-destination pairs due to the lower probability of setting a path for the source-destination pair having a long distance.

C. Number of Virtual Coordinates

Fig. 6a shows the simulation results for the average number of virtual coordinates assigned to a node using ABVCap_Uni. The greater the number of nodes, the fewer the number of obstacles, or the higher the percentage of bidirectional links, the fewer virtual coordinates are assigned to a node, because fewer axes overlap. The number of virtual coordinates per node is about 2 – 8.

D. Hop Distance from a Visible Neighbor

Fig. 6b shows the simulation results for the average hop distance for node $u$ from a visible neighbor $v \in u.N_{OUT}$. The more nodes exist or the higher is the percentage of bidirectional links, the smaller is the hop distance from a visible neighbor. This observation is reasonable. The more obstacles exist, the smaller is the hop distance from a visible neighbor because of the higher network density. The number of nodes, the percentage of bidirectional links, and the number of obstacles each have minor effects on the hop distance from a visible neighbor, which is about 2 in all cases.

E. Number of Broadcasts

Fig. 6c illustrates the simulation results for the average number of broadcasts required for each node using ABV-
Cap Uni. In the simulation, messages are broadcast using the technique on trading time [8], and the COOR_SET, UP_SET, and DOWN_SET messages are broadcast at approximately the same time and each travels at approximately the same speed [6]. A node, $u$, sends a message to a neighbor, $v \in u.N_{IN}$, using scoped flooding with a limited hop distance, initially set to 2, that is increased by one until $v$ receives the message. The greater the number of nodes, the fewer the number of obstacles, or the higher the percentage of bidirectional links, fewer broadcasts result because nodes are assigned fewer virtual coordinates. The number of broadcasts per node is about $10 - 27$.

VII. CONCLUSION

We propose a virtual coordinate assignment protocol, ABVCap Uni, to assign virtual coordinates to nodes that have no geographic information, and a routing protocol based on the ABVCap Uni virtual coordinates in wireless sensor networks with unidirectional links. The virtual coordinate includes eight entries: longitude, latitude, ripple, up, down, ring-initiator, ring-number, and ring-order. The longitude and latitude coordinates denote the location of the node, and the other coordinates are used to assist packet delivery. In ABVCap Uni routing, nodes are not required to compute and memorize global topology features. ABVCap Uni guarantees packet delivery in the discrete domain, in which case, the routed packet only needs to carry the longitude and latitude coordinates of the destination.

Simulations show that the delivery rate is about $69 - 87\%$ and $68 - 99\%$ with GLDR+VLM and Euclidean routing, respectively. In terms of the path length, ABVCap Uni routing is larger than Euclidean and GLDR+VLM routing by about $26 - 77\%$ and $0 - 12\%$, respectively. Simulations also show that the number of virtual coordinates assigned to each node is about 2 – 8 and that the number of broadcasts required for each node in ABVCap Uni is about $10 - 27$.

Future research includes study of the manner in which to guarantee packet delivery in a component of the network that becomes disconnected due to switch-off or mobility of sensors. Additional research includes guaranteeing strong connectivity of the network while minimizing power levels of nodes so that ABVCap Uni can always route the packet to the destination with minimum power consumption.

REFERENCES