IEEE International Conference on Communications ICC 2011, Kyoto, Japan, June 2011

Directional Virtual Coordinate Systems for Wireless Sensor Networks*

Dulanjalie C. Dhanapala and Anura P. Jayasumana, *Member, IEEE*
Department of Electrical and Computer Engineering,
Colorado State University, Fort Collins, CO 80523, USA
{dulanjalie.dhanapala, Anura.Jayasumana}@Colostate.edu

***Abstract*— A Directional Virtual Coordinate System (DVCS) is proposed based on a novel transformation that restores the lost directionality information in a Virtual Coordinate System (VCS). VCS is an attractive option to characterize the node locations in Wireless Sensor Networks (WSNs), instead of using geographical coordinates, which is expensive or difficult to obtain. A VCS characterizes each node in a network with the minimum hop distances to a set of anchor nodes as its coordinates. The proposed transformation supplements the virtual coordinates, thus preserving all the inheriting properties such as embedded information of geodesic distances in the coordinates. The virtual directionality introduced, alleviates the local minima issue present in original VCS. Properties of this virtual directional domain are discussed. With these directional properties, it is possible, for the first time, to consider deterministic algorithms in the virtual domain, as illustrated with a constrained tree network example. A novel routing scheme called Directional Virtual Coordinate Routing (DVCR), which illustrates the effectiveness of the Directional Virtual Coordinate domain is proposed. DVCR significantly outperforms existing VCS routing schemes Convex Subspace Routing (CSR) and Logical Coordinate Routing (LCR), while achieving a performance similar to the geographical routing scheme - Greedy Perimeter Stateless Routing (GPSR), but without the need for node location information.**

***Index Terms*— Routing, Sensor Networks, Virtual Coordinates, Virtual Directions**

I. Introduction

VIRTUAL Coordinate Routing (VCR) and Geographical Routing (GR) are two main classes of address-based routing schemes for WSNs. Geographical routing [1][6] relies on physical location information of nodes, and directional information that can be derived from individual node locations. Obtaining location information however requires mechanisms like GPS, which are costly or infeasible in some applications, or localization algorithms, which are complex and error prone as a result of their reliance on measurements such as RSSI or time delay. GR also suffers from poor routability in the presence of concave physical voids. Connectivity based approaches provide an alternative solution to overcome weaknesses associated with location determination and geographical voids. VCR [2]-[4] uses a Virtual Coordinate System (VCS) that characterizes each node by a coordinate vector of size M, consisting of the shortest hop distance to each of a set of M anchors, which may be generated using network wide flooding [3]. The number of anchors becomes the networks' dimensionality.

Routing in virtual domain has two phases. Most of the VCR schemes [2][3][10] use Greedy Forwarding (GF) combined with a back-tracking algorithm. In GF, a packet is simply forwarded to a neighbor that is closer to the destination than the packet holding node. Virtual Coordinates (VCs) of nodes are used for distance evaluation between nodes as well as for node identification (ID). Distance is estimated using either L^1 or L^2 norm based on VCs; such values are often unreliable estimates of the distance as the contributions due to different anchors are not orthogonal. When a closer neighbor cannot be found, i.e., the packet is at a local minima, back-tracking is employed to climb out of it.

Performance of VCR and anchor placement is highly correlated. Anchors may be selected randomly [4] or by selecting nodes with specific properties, e.g., by selecting all the perimeter nodes [9]. When a message reaches a local minima, an expanding ring search is performed in [9] until a closer node is found or TTL (Time-To-Live) expires. In Virtual Coordinate assignment protocol (VCap), the coordinates are defined based on three anchors [2]. At local minima, VCap causes a packet to follow a rule called local detour. In Logical Coordinate based Routing (LCR) [3] backtracking is used when GF fails at a local minima. Aligned virtual coordinate system (AVCS) [8] re-evaluates VCs by averaging its own coordinates with neighboring coordinates to overcome local minima. In Axis-based Virtual Coordinate Assignment Protocol (ABVCap) [10], 5-tuple VC is assigned to each node corresponding to longitude, latitude, ripple, up, and down. All these VCR protocols rely mainly on Greedy forwarding, followed by a backtracking scheme to overcome local minima. Convex Subspace Routing [4], in contrast, selects dynamically changing subsets of anchors to provide convex distance surfaces for routing.

VCS has its inherent advantages and disadvantages. VCS is a connectivity based higher dimensional transformation of WSN, resulting in some attractive properties such as considerably high routability without any geographical information, effectiveness of connectivity information embedded in VCs, and insensitivity to physical voids and to localization errors. Physical domain to virtual domain transformation is many to one as VCs are insensitive to

*This research is supported in part by NSF Grant CNS-0720889

directions, which is one main cause of identical coordinates and local minima. If an adequate number of anchors are not appropriately deployed, it may also cause the network to suffer from identical coordinates and local minima [4] resulting in logical/virtual voids. Identification of the optimal number of anchors and proper anchor placement remains a major challenge [4].

Inadequacies associated with VCS are due to loss of directionality information and the lack of information about physical network topology. This paper proposes, for the first time, novel transformation with which the VCS can regain its lost directionality, thus acquiring some sense of physical location, to supplement the connectivity information embedded in original VCS. No additional transmission cost is involved, as each node can evaluate the directional values with VCs available locally. Acquiring directionality provides new information, hitherto not available in VCS, facilitating a new approach for designing a broad spectrum of WSN algorithms. Technique to identify 'good' anchors alleviating the issues in VCS discussed in [4], novel routing schemes and generating topology preserving maps with lower cost compared to the proposed scheme in [5] are among potential applications of DVCS. As an example, we illustrate a deterministic algorithm for routing in a *constrained tree* topology, based on new transformed coordinates in directional virtual space. To our knowledge, no deterministic algorithms have been developed before using the VC domain.

In sum the contribution of this paper can be listed as below

1. Novel concept of transforming directionless VCS to directional VCS
2. Properties in the new directional VCS and deterministic routing in a constrained tree network
3. Directional Virtual Coordinate Routing (DVCR) - Routing in directional VCS

The proposed routing scheme in directional virtual space-Directional Virtual Coordinate Routing (DVCR) is compared with CSR [4] and LCR [3]. Moreover it is compared with a geographical routing scheme called Greedy Perimeter Stateless Routing (GPSR) [6] which makes greedy forwarding decisions until it fails, for example due to a geographical void, and attempts to recover by routing around the perimeter of the void. DVCR outperforms, CSR and LCR with a noticeable value achieving more or less the similar performance as GPSR.

Section II explains the new transformation of VCS to directional VCS. In Section III, a deterministic routing based on directional coordinates in a simple tree is discussed and in Section IV transformed domain network partition property was illustrated. A novel routing protocol is proposed in Section V. Performance evaluation is in Section VI. Finally, Section VII concludes the contribution of this paper.

II. Directionless Virtual Space to Directional Virtual Space Transformation

As a virtual coordinate corresponds to the distance to a particular anchor, the physical domain to virtual domain transformation is many to one. The coordinate propagates concentrically around the anchor, thus losing the directional information. Consequences of this mapping include identical coordinates and local minima encountered in routing [4]. A novel transformation of VCs to regain the directionality lost is proposed next. The notations used are summarized in Table 1.

TABLE I
NOTATIONS USED IN THE TEXT AND THEOREMS

Notation	Description
N	Number of network nodes
$N_i, N_d, N_s \in N$	Node i, Destination, Source
M	Number of anchors
$A_i, i = 1{:}M$	Anchor set (a subset of N)
$h_{N_iN_j}$	Minimum hop distance between node N_i and N_j
$[h_{N_iA_1}, \dots, h_{N_iA_M}]$	Node N_i's VC
$[n_{i1} \dots n_{ij} \dots n_{iL}]$; $j = 1{:}L, L \le C_2^M$	Node N_i's transformed VC
$D_{N_iN_j}$	Distance between N_i and N_j in transformed domain
K	Neighbors set
$N_{i,Prev}$	Node that forward the packet to current node
$N_{i,Next}$	Node that current node will forward the packet

First consider a 1-D network where one can easily visualize the concept behind the transformation. Table 2 contains the VCS for the 1-D network shown in Fig. 1 with respect to two anchors A_1 and A_2, which are $h_{A_2A_1}$ hops apart (8 hops in Fig. 1). Note that $h_{N_iA_1}$ propagates symmetrically from the corresponding anchor, thus loosing directionality. Even though $(h_{N_iA_1} - h_{N_iA_2})$ provides the sense of directionality for the region between anchors, as can be seen in Table 2, it remains constant outside the region bounded by anchors, thus failing to provide directional information. Conversely, $(h_{N_iA_1} + h_{N_iA_2})$ has a constant value in between the anchors, but linearly varies elsewhere. By combining those, a node N_i is characterized using $f(h_{N_iA_1}, h_{N_iA_2})$ that defined as,

$$f(h_{N_iA_1}, h_{N_iA_2}) = \frac{1}{2h_{A_2A_1}}(h_{N_iA_1} - h_{iA_2})(h_{N_iA_1} + h_{N_iA_2}) \quad (1)$$

$f(h_{N_iA_1}, h_{N_iA_2})$, as shown in Table 2, maps the nodes' VC $\equiv (h_{N_iA_1}, h_{N_iA_2})$ linearly to the real axis with positive and negative values with center at the midpoint of A_1 and A_2, providing directional information in the virtual domain. The term $\frac{1}{2\,h_{A_2A_1}}$ normalizes the distance to provide a unit difference of the ordinate between two adjacent nodes.

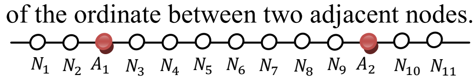

Fig. 1. 1D network with two anchors A_1 and A_2.

TABLE II
EXAMPLE VC TRANSFORMATION STEPS IN 1D NETWORK SHOWN IN FIG. 1

NODE ID	N_1	N_2	A_1	N_3	N_4	N_5	N_6	N_7	N_8	N_9	A_2	N_{10}	N_{11}
h_{iA_1}	2	1	0	1	2	3	4	5	6	7	8	9	10
h_{iA_2}	10	9	8	7	6	5	4	3	2	1	0	1	2
$h_{iA_1} - h_{iA_2}$	-8	-8	-8	-6	-4	-2	0	2	4	6	8	8	8
$h_{iA_1} + h_{iA_2}$	12	10	8	8	8	8	8	8	8	8	8	10	12
$f(h_{iA_1}, h_{iA_2})$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6

Prior VC based routing schemes such as LCR [3] encountered local maxima at anchors even for this simple linear array [4], but with this transformation, the local minima problem is overcome completely.

Furthermore, one can now view each node as a point in a vector space. Define, $\vec{\boldsymbol{u}}_{\boldsymbol{A_1A_2}}$ as the unit vector in A_1A_2 direction, which is named as a *virtual direction.* Ordinate in Fig. 1 can be written in the form:

$$\vec{\boldsymbol{f}}(\boldsymbol{h_{N_iA_1}}, \boldsymbol{h_{N_iA_2}}) = f(h_{N_iA_1}, h_{N_iA_2})\vec{\boldsymbol{u}}_{\boldsymbol{A_1A_2}} \quad (2)$$

Now consider a 2D sensor network. Select two coordinates of node N_i: $h_{N_iA_j}$ and $h_{N_iA_k}$, with respect to anchors, A_j and A_k. Then the magnitude of the virtual distance vector component in the A_jA_k direction is given by

$$f\left(h_{N_iA_j}, h_{N_iA_k}\right) = \frac{1}{2h_{A_jA_k}}\left({h_{N_iA_j}}^2 - {h_{N_iA_k}}^2\right) \quad (3)$$

Since there are M ordinates available, C_2^M different virtual directions (though not orthogonal to each other) may be specified, and can be evaluated locally at each node. Some of the properties in this directional Virtual Space are discussed next. Each transformed domain ordinate can be written in the form:

$$\vec{\boldsymbol{f}}\left(\boldsymbol{h_{N_iA_j}}, \boldsymbol{h_{N_iA_k}}\right) = f\left(h_{N_iA_j}, h_{N_iA_k}\right)\vec{\boldsymbol{u}}_{\boldsymbol{A_jA_k}} \quad (4)$$

where, $\vec{\boldsymbol{f}}\left(\boldsymbol{h_{N_iA_j}}, \boldsymbol{h_{N_iA_k}}\right)$ is the vector representation of the transformed ordinate of A_iA_j and $\vec{\boldsymbol{u}}_{\boldsymbol{A_jA_k}}$ is the virtual direction obtained by A_jA_k. We can also define the virtual distance between two nodes N_p and N_q in this direction to be,

$$F_{A_jA_k}(N_p, N_q) = f\left(h_{N_pA_j}, h_{N_pA_k}\right) - f\left(h_{N_qA_j}, h_{N_qA_k}\right) \quad (5)$$

Fig. 2 shows the 2D extension of the transformation to a grid network. Transformed coordinates are given by $f(A_1, A_2)$, $f(A_3, A_4)$], providing directionality, and dividing the grid in to four quadrants.

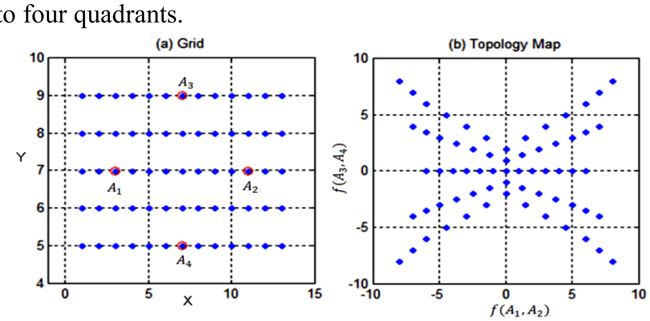

Fig. 2. (a) Physical map of the grid (b) Directional domain map, $f(A_1, A_2)$ Vs. $f(A_3, A_4)$, of 2D grid.

Property 1: Consider a 2-D network with two anchors $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$, which are $\boldsymbol{h_{A_1A_2}}$ hops apart, the transformation $\boldsymbol{f}(\boldsymbol{h_{N_iA_1}}, \boldsymbol{h_{N_iA_2}}) = \frac{1}{2\boldsymbol{h_{A_1A_2}}}({\boldsymbol{h_{N_iA_1}}}^2 - {\boldsymbol{h_{N_iA_2}}}^2)$ partitions the network in to two sections with positive and negative ordinates. Moreover, the nodes that are equidistance in terms of hops to $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ have $\boldsymbol{f}(\boldsymbol{h_{N_iA_1}}, \boldsymbol{h_{N_iA_2}}) = \boldsymbol{0}$.

Proof: Consider the transformation in Eq. (3). The nodes with $h_{N_iA_1} > h_{N_iA_2}$, have positive transformed ordinate while nodes with $h_{N_iA_1} < h_{N_iA_2}$ have negative transformed ordinates. Nodes with $h_{N_iA_1} = h_{N_iA_2}$ have zero ordinate in the transformed domain. QED

By selecting more anchor pairs - randomly or systematically, the network can be partitioned based on the sign of each ordinate as in four quadrant Cartesian coordinate system. Partitioning networks based on the sign is demonstrated in Section V. The relationship of (3), allows the derivation of distance and coordinate relationships, that in turn allow for systematic, and even deterministic methods for routing using VCs. To our knowledge, this is the first instance of use of VCs this way. We illustrate the use of the directional information using a simple example next. A routing algorithm based on DVCS is presented and evaluated in Section V.

III. ROUTING IN DIRECTIONAL VIRTUAL DOMAIN

In this section we demonstrate how it is possible to exploit properties of DVCS to develop strategies to identify routing paths, which was not feasible with directionless VCS. For the example presented in this section, that of a simple tree, a DVCS based deterministic routing protocol can be used to guarantee 100% routability. This direction based method can be considered as a foundation for developing relationships to discover routing paths in more complex topologies.

Consider a *Constrained Tree* (CT) network with branches extending to both the sides off one main trunk (backbone). Assume that the maximum degree of a node is three, i.e., no two branches occur at same point, and there are no branches off branches (See Fig. 3). CT network topologies fit well in environments such as mine-shafts and pipeline distribution systems. The traditional VCR schemes such as LCR and CSR cannot guarantee 100% routability in these networks. Consider the network shown in Fig. 3, where a packet is to be sent from node $N_s \equiv (7{,}13)$ to $N_d \equiv (16{,}10)$. In this case packet will be forwarded to N_{s+1} in Greedy Forwarding based on VCs, whereas the correct neighbor to forward the packet is N_{s-1}.

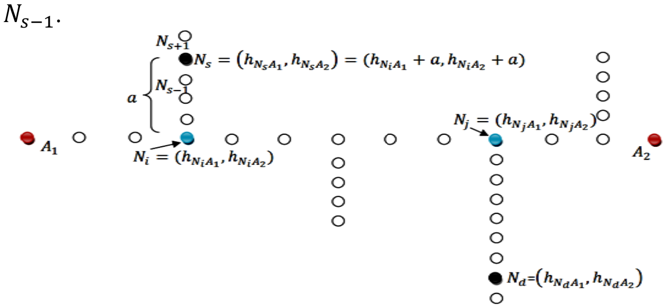

Fig 3. Constrained tree network with two anchors A_1 and A_2.

Property 2: In a CT, the gap between any two adjacent nodes in a branch is constant, and uniquely dependent on the junction node. Specifically, for a branch off node $\boldsymbol{N_i} = (\boldsymbol{h_{N_iA_1}}, \boldsymbol{h_{N_iA_2}})$, the gap is given by

$$\boldsymbol{g} = (\boldsymbol{h_{N_iA_1}} - \boldsymbol{h_{N_iA_2}})/(\boldsymbol{h_{N_iA_1}} + \boldsymbol{h_{N_iA_2}}) \quad (6)$$

Proof: Junction node coordinates, are unique. Consider the junction node $N_i = (h_{N_iA_1}, h_{N_iA_2})$. A positive integer x which makes $(h_{N_iA_1} + x) - (h_{N_iA_2} - x) = h_{N_iA_1} - h_{N_iA_2}$ does not exist. Hence gap in each branch given by (6) is unique. QED

Property 3: In a CT, only the members of the backbone $\boldsymbol{N_i} = (\boldsymbol{h_{N_iA_1}}, \boldsymbol{h_{N_iA_2}})$ satisfies

$$\boldsymbol{h_{N_iA_1}} + \boldsymbol{h_{N_iA_2}} = \boldsymbol{h_{A_1A_2}} \quad (7)$$

Proof: This can be proved by the characteristics of VC. QED

Property 4: In a CT, junction node $N_i = (h_{N_iA_1}, h_{N_iA_2})$ can identify the members on its branch and how many hops that each member of the branch is away from itself.

Proof: As in property 2, gap in a branch is unique and it is known by the branching node. Assume a node $N_d = (h_{N_dA_1}, h_{N_dA_2})$ is a member of the branch from junction N_i. Then virtual ordinates of N_d satisfies $h_{N_dA_1} = h_{N_iA_1} + ga$ and $h_{N_dA_2} = h_{N_iA_2} + ga$. g can be found as in (6). Thus *'a'*, the number of hops to the backbone from N_d can be calculated. Consider another junction node $N_j = (h_{N_iA_1} + x, h_{N_iA_2} - x)$ with gap in its branch g'. There do not exist a positive integer x and a', which satisfy $h_{N_dA_1} = h_{N_iA_1} + x + g'a'$ and $h_{N_dA_2} = h_{N_iA_2} - x + g'a'$. Hence N_d exists only in the branch of N_i. QED

Theorem 1: In a tree with branches on two sides off one main trunk, with no branches off branches, with node degree $\leq$ 3, 100% routability can be achieved with two anchors placed one at each extreme of the trunk.

Proof: Routing is performed in two steps to achieve 100% routability. Initially packet will be routed to the junction node where the current node holding branch connects to the backbone. Then packet is routed to the destination from the junction node. But current node should find out the coordinates of the junction node. Consider Fig 3. Let the current node be N_s and destination be N_d. Distance between anchors, $h_{A_1A_2}$, in (7) is known and (6) is simply the difference between current node and the neighbor. Therefore the VC of the branching node $(h_{N_iA_1}, h_{N_iA_2})$ can be found. Thus *'a'*, the number of hops to the backbone,can be found. If a and $(h_{N_iA_1}$, $h_{N_iA_2})$ are known, a packet can reach the backbone, i.e. node $(h_{N_iA_1}, h_{N_iA_2})$, and then it should be routed to the destination. Any junction node can identify whether N_d is in its branch or not. If N_d is a member of the branch, junction node will forward the packet to its neighbor on the branch. If N_d is not a member of its branch, junction node will forward the closest neighbor, excluding the neighbor on the branch, to N_d. QED

A_1 and A_2 need not be at corners but we should make sure all the branches are in between A_1 and A_2. Moreover, if number of nodes in-between A_1 and A_2 is odd, and if there is a branch at the middle point, all the nodes in that branch will have zero ordinate (See proof of Property 1). That can be avoided by assuring hop distance between A_1 and A_2 to be even when anchor A_2 is selected. Furthermore, in a tree with branches on two sides (provided branches are off one main trunk), when there is a branch off a branch, it can be treated as a sub-constrained tree network hence need to add 'exactly one' additional anchor. Routing should be done in each sub-constrained tree based on corresponding anchors. This will allow us to get the number of anchors needed for any tree - i.e., any graph without cycles.

A simple adjustment can be proposed if there are two branches at the same node. After generating VCS with respect to anchors A_1 and A_2, members of the backbone and branching nodes which have two branches can identify themselves. After that junction nodes with two branches can add one more bit to the coordinate of the nodes in one of the branch to indicate whether it's the upper or lower branch. This newly added bit can be used to prevent identical coordinates in the upper and lower branches. Theorem 1 holds for the network after the small adjustment. Moreover, as observed, the same approach can be applied in a tree network with degree 3.

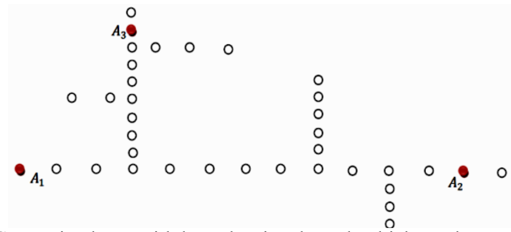

Fig 4. Constrained tree with branches in a branch which can be modeled as a sub constrained tree.

IV. Simulation Results: Partitions in Directional Virtual Space

Effectiveness of the directional domain is evaluated in five representative examples of variety of networks, as shown in Fig. 5. (a) spiral shaped network with 421 nodes, (b) a grid based network with 100 randomly missing nodes (800 nodes) (c) a 496-node, circular shaped network with three physical voids/holes, (d) a network of 343 nodes mounted on walls of a building (e) odd shaped network with 550 nodes. Communication range of a node in all five networks is unity. MATLAB® 2009b was used for the computations.

Network partitioning based on sign of new ordinates

As stated in property 1, the sign of each ordinate in transformed domain is used to identify different sectors of the network. In the networks shown in Fig. 5, randomly selected three anchors, A_1, A_2,and A_3 were placed. Then using the transformation given in (3), the new coordinates, $[f(h_{iA_1}, h_{iA_2}), f(h_{iA_1}, h_{iA_3}), f(h_{iA_2}, h_{iA_3})]$ were generated by each node locally. Based on the sign of the each ordinate in the directional coordinate, i.e. positive/ negative, , different sections were colored as shown in Fig. 5. Since 3 anchors' ordinates are used for transformation, cardinality in transformed domain is 3 (C_2^M). Hence the maximum possible sign combinations in the network is 2^3. As in Fig. 5, not all the sign combinations exist but existing combinations clearly partition the network. Applications of directional virtual coordinate space are diverse:

Routing: Transformed domain properties lead to the design of various routing schemes. Since DVCS preserves the connectivity information of its parent VCS whilst gaining the directionality, a novel routing schemes combining the advantages of VCs and directionality can be designed. Also existing Geographical coordinate based routing schemes can be adapted to DVCS.

Strategically anchor placements schemes: Based on single virtual direction, corner nodes of the network can be identified providing an ideal technique to recognize 'good' anchor candidates.

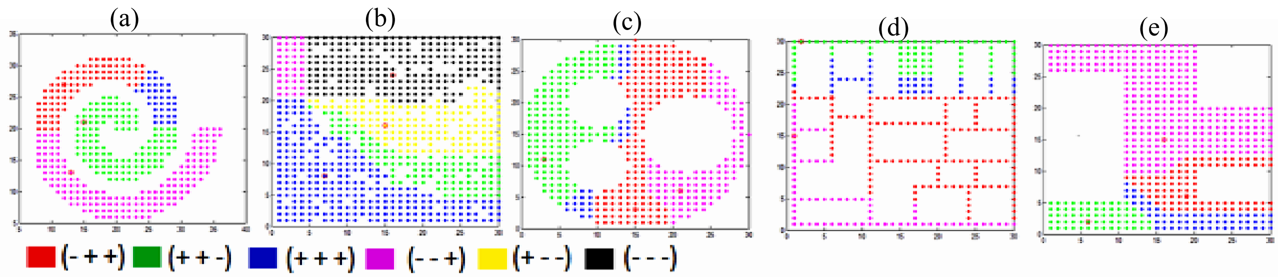

Fig. 5. Partitions of (a) Spiral shaped network with 421 nodes, (b) A grid based network with 100 randomly missing nodes (c) A 496-node circular shaped network with three physical voids/holes, (d) A network of 343 nodes mounted on walls of a building (e) Odd shaped network with 550 nodes, based on the sign of the ordinates in transformed domain created by three randomly selected anchors A_1, A_2, A_3. Transformed domain has three ordinates generated by (A_1, A_2), (A_1, A_3), and (A_2, A_3) pairs.

V. Directional Virtual Coordinate Routing - DVCR

In this section we present a novel routing scheme based on transformed coordinates. In a network with M randomly selected anchors, a node can evaluate its transformed coordinates of cardinality $P = C_2^M$ locally. Let the transformed current node coordinate be $N_i \equiv [n_{i1} \dots n_{ij} \dots n_{iP}]$ and that of the destination be $N_d \equiv [n_{d1} \dots n_{dj} \dots n_{dP}]$. L^2 distance between N_i and N_d is using transformed coordinates is,

$$D_{N_i N_d} = \sqrt{\textstyle\sum_{\forall j}(n_{ij} - n_{id})^2}\,; j = 1{:}J \leq C_2^M \tag{8}$$

The packet is forwarded to a neighbor using Greedy Forwarding (GF), To overcome the local minima, the minima node performs an approximate hop distance estimation from itself and also from neighbors as explained next, based on (9),(10) and (11). Assumption is that L^1 in transformed domain (see (5)) is a good representation of the hop distance. Hence there exists a neighbor which has lower hop distance (estimated) to destination. For this estimation, two directional ordinates are sufficient.

Define the ordinate difference set $\Delta_{A_pA_q}$ between current node N_i and all the neighbors $N_k \in K$ as,

$$\Delta_{A_pA_q} = \left|\left(F_{A_pA_q}(N_i, N_k)\right)\right| \;;\; N_k \in K \tag{9}$$

Since there are $|K|$ number of neighbors, size of $\Delta_{A_pA_q}$ is the same as that of $|K|$. Consider $\Delta_{A_1A_2}$ with respect to $f(A_1, A_2)$. Let α_{12} and β_{12}, be $\max(\Delta_{A_1A_2})$ and $\min(\Delta_{A_1A_2})$ respectively. Then the approximate hop distance between N_i and N_d is represented with respect to $f(A_1, A_2)$,

$$\alpha_{12} n + \beta_{12} m = \left|\left(F_{A_1A_2}(N_i, N_d)\right)\right| \tag{10}$$

Similarly α_{34} and β_{34} is obtained following the same method with respect to another ordinate $f(A_3, A_4)$. Another representation of the approximate hop distance between N_i and N_d with respect to $f(A_3, A_4)$,

$$\alpha_{34} n + \beta_{34} m = \left|\left(F_{A_3A_4}(N_i, N_d)\right)\right| \tag{11}$$

By solving (10) and (11) the approximate hop count from current node to destination which is $n + m$ can be estimated. This can be repeated for neighbors set K to get the hop distances from neighbors to destination. Packet will be greedily forwarded to the neighbor selected by this hop count approximation. Algorithm of the routing protocol can be summarized as in Fig. 6.

while ($N_d \neq N_i \,\|\, TTL \geq 0$) % *TTL*- Time-To-Live
    $R_k = D_{jd}\,; N_j \in K; N_j \neq N_{i,Prev}, N_{i,Next}$ % Calculate the transformed
    domain distance from Neighbors set K to destination excluding
    $N_{i,Next}$ and $N_{i,Prev}$
    $d_{jd} = D_{jd}$ %Current distance to desination

  if $Min(R_k) == 0$
    if $N_d == N_i$
      ROUTED
    else % if identical coordinates
      Find $n + m$ for N_i and $N_j \in K; N_j \neq N_{i,Prev}, N_{i,Next}$ and pick the closest neighbor. If no closer neighbor exists routing fails
    end
  elseif $Min(R_k) \leq d_{id}$
    $N_{i,Prev} = N_i$
    $N_i = \underset{j}{argmin}\, R_k(j)$
    $N_{i,Next} = \underset{j}{argmin}\, R_k(j)$
  elseif $Min(R_k) > d_{id}$ %Local minima
      Find $n + m$ for N_i and $N_j \in K; N_j \neq N_{i,Prev}, N_{i,Next}$ and pick the closest neighbor. If no closer neighbor exists routing fails
  end
end

Fig. 6. Pseudo code of DVCR algorithm.

VI. Performance of Directional Virtual Coordinate Routing

The performance of proposed Directional Virtual Coordinate Routing (DVCR) scheme is evaluated next, for the five networks introduced in Fig. 5. Performance of DVCR is compared with two virtual coordinate-based routing schemes - Logical Coordinate Routing (LCR) [3]and Convex Subspace Routing (CSR) [4], and a geographic routing scheme - Greedy Perimeter Stateless Routing (GPSR) [6]. Five randomly selected nodes serves as anchors. In LCR implementation, we assumed that the entire path traversed is available at each node so that backtracking can be perfectly performed avoiding any loops; i.e., the implemented case is the best case of LCR, and is not achievable in practice due to the cost involved in transmitting the required information. Time-To-Live (TTL) of the packet is set to 100 hops.

Average routability, average path length that packets traversed, and average energy consumption per successfully delivered packet are used as the performance metrics. Average routability evaluation considers all source-destination pairs; i.e., each node generated a set of (N-1) messages, with one message for each of the remaining node as destination.

Average routability = $R_{AVG}\%$

$$\frac{\text{Total \# of packet that reached the destination}}{\text{Total number of packet generated}}\% \tag{12}$$

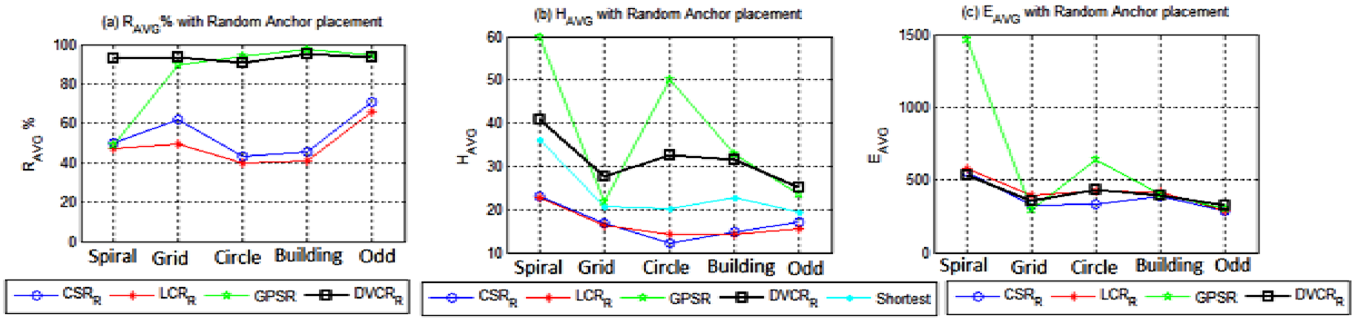

Fig 7. (a) R_{AVG}% (b) H_{AVG}(c) E_{AVG} of CSR, LCR, GPSR and DVCR with 5 randomly placed anchors in Spiral, 30 by 30 node Grid with 800 nodes, Circle with 3 holes, building and odd networks.

$$\text{Average path length} = H_{AVG} = \frac{\text{Cumilative number of hops that each packet traversed}}{\text{Total number of packet generated}} \quad (13)$$

Note that the average path length calculation includes the path lengths for unrouted messages as well.

In order to have a fair estimation of the energy consumption, average energy consumption per successfully routed packet was defined as,

$$\text{Average energy consumption per successful packet delivery} = E_{AVG} = \frac{E_{\propto} \times H_{AVG} \times Packet\ size}{R_{AVG}\%} \quad (14)$$

where $E_{\propto}$ is average energy per byte. For all the routing schemes a fixed packet length of 12 bytes was assumed where 4 bytes each for destination ID, current node ID and VC/Physical coordinates. For the random anchor placement performance was averaged over five random anchor configurations.

Performance comparison in terms of R_{AVG}%, H_{AVG} and E_{AVG} are as shown in Fig. 7. With random anchor placement, the proposed scheme DVCR out performs CSR and LCR with H_{AVG} to shortest distance path length (H_s) ratio close to unity.

Also it out performs GPSR in spiral and grid with missing nodes and achieves almost the same performance in rest of the networks. Even though DVCR achieves a higher $R_{AVG}\%$, E_{AVG} (see Fig 7 (c)) is less than that of GPSR while very close to E_{AVG} of CSR and LCR. It is important to note that GPSR relies on accurate location information, achievable via expensive hardware such as GPS, or localization schemes subject to significant complexity and estimation errors. The importance of directionality information was illustrated by the performance of DVCR and anchor selection mechanism. More importantly required number of anchors is 5, which is significantly lower number compared to the anchors used in other literature, to achieve $R_{AVG}\%$ over 95%.

VII. CONCLUSIONS

A simple and novel transformation is proposed for virtual coordinates that for the first time allows VCS to recover directionality lost during the coordinate generation, thereby significantly increasing the effectiveness of virtual coordinate systems in routing. The issues such as identical coordinates and local minima are caused by mainly due to the loss of directionality in virtual coordinate system generation are mitigated in the directional domain. Regained directions are called virtual directions. Network partitioning, routing in special cases of tree networks are some of the properties discussed.

Directional space contains the inherent connectivity information while sense of directions of the node arrangement, which provides a good environment for routing. The proposed routing scheme - Directional Virtual Coordinate Routing (DVCR) outperforms Convex Subspace Routing (CSR) and Logical Coordinate Routing (LCR) with 38.9% and 44.6% average increment in average routability over five network types respectively, in 1.35 average path length to shortest path length ratio with randomly selected 5 anchors, which is less than 1.5% of nodes.

Effective anchor placement strategy and topology map generation by selecting nearly orthogonal virtual directions are under investigation.

REFERENCES

[1] J.N. Al-Karaki, and A.E. Kamal, "Routing techniques in wireless sensor networks: a survey," IEEE Wireless Communications, Vol. 11, pp.6-28, Dec. 2004.

[2] A. Caruso, S. Chessa, S. De, and A. Urpi, "GPS free coordinate assignment and routing in wireless sensor networks," Proc. 24th IEEE Joint Conf. of Computer and Communications Societies, Vol. 1, pp. 150-160, Mar. 2005.

[3] Q. Cao and T. Abdelzaher, "Scalable logical coordinates framework for routing in wireless sensor networks," ACM Transactions on Sensor Networks, Vol. 2, pp. 557-593, Nov. 2006.

[4] D. C. Dhanapala and A. P. Jayasumana, "CSR: Convex subspace routing protocol for WSNs," Proc. 34rd IEEE Conf. on Local Computer Networks, Oct. 2009.

[5] D. C. Dhanapala and A. P. Jayasumana, "Topology preserving maps from virtual coordinates for wireless sensor networks," Proc. 35rd IEEE Conf. on Local Computer Networks, Oct. 2010.(Accepted)

[6] B. Karp and H. T. Kung., "Greedy perimeter stateless routing (GPSR) for wireless networks," Proc. 6th annual ACM/IEEE Int. Conf. on mobile computing and networking (mobicom), 2000, pp. 243–254.

[7] C-H Lin, B-H Liu, H-Y Yang, C-Y Kao, and M-J Tsai, "Virtual-coordinate-based delivery-guaranteed routing protocol in wireless sensor networks with unidirectional links," Proc. IEEE INFOCOM 2008, pp: 351-355 , 13-18 April 2008.

[8] K. Liu and N. Abu-Ghazaleh, "Aligned virtual coordinates for greedy routing in WSNs," Proc. IEEE Int. Conf. on Mobile Ad-hoc and Sensor Systems, pp.377–386, Oct. 2006.

[9] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica , "Geographic routing without location information," Proc. 9th Int. conf. on Mobile computing and networking, pp.96 - 108, 2003.

[10] M. J. Tsai, H. Y. Yang, and W. Q. Huang, "Axis based virtual coordinate assignment protocol and delivery guaranteed routing protocol in wireless sensor networks," Proc. INFOCOM 2007, pp: 2234-2242, 6-12 May 2007.