Proc. 36th IEEE Conference on Local Computer Networks, Bonn, Germany, Oct. 2011.

Anchor Selection and Topology Preserving Maps in WSNs – A Directional Virtual Coordinate Based Approach

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

Abstract*— Virtual Coordinate Systems (VCS) characterize each node in a network by its hop distances to a subset of nodes called anchors. Performance of VCS based algorithms is highly sensitive to number of anchors and their placement. Extreme Node Search (ENS), a novel and efficient anchor placement scheme, is proposed that demonstrates significantly improved performance over state-of-the-art. ENS starts with two randomly placed anchors and then uses a Directional Virtual Coordinate (DVC) transformation, which restores the lost directionality in traditional VCS, to identify anchor candidates in a completely distributed manner. A vector-based representation is proposed for the DVC domain, which is then used to introduce the concept of angles between virtual directions in the transformed domain. Ability to specify cardinal directions and use angles is a radical change from the traditional VC system approaches. By selecting two anchor pairs with near orthogonal directional coordinates under the DVC transformation, a novel Topology Preserving Map (TPM) generation scheme is developed. This new TPM generation scheme requires significantly less computations than the existing PCA based method. Use of ENS significantly enhances the PCA based TPM generation as well. Simulation results for representative WSNs indicate that the ENS based anchor sets significantly improve performance of all prominent VC based routing schemes. For instance, Directional Virtual Coordinate Routing, when combined with ENS anchor placement strategy, outperforms even the geographic Greedy Perimeter Stateless Routing scheme that relies on exact physical node coordinates.

Keywords- Anchor Placement, Routing, Sensor Networks, Virtual Coordinates, Virtual Directions, Topology Maps

I. INTRODUCTION

Virtual Coordinate Systems (VCS) provide a compelling alternative for structuring/organizing Wireless Sensor Networks (WSNs) without the need for location information. In virtual domain, each node is represented by a M dimensional vector of *virtual coordinates*, consisting of shortest path hop distances to M nodes named *anchors*. They are different from anchors used in physical position based localization schemes in that anchors for VCS are not localized and have the same capabilities as any other node in the network. A major advantage of connectivity information based VCS over physical position based systems is that they completely avoid the cost, complexity, and uncertainties associated with node localization using GPS or distance estimations. GPS equipped nodes are expensive, not energy efficient and not feasible in some applications [2]. The alternative is to use analog measurements, such as Receiver Signal Strength Indicator (RSSI) or Time-of-Arrival (TOA) [2] to estimate distances to other nodes, and thereby obtain node positions. These analog measurements encounter uncertainties and complexities due to multipath fading, scattering, interference and poor line-of-sight, which are difficult to overcome in many indoor and outdoor environments. VCS thus outcompetes geographical location based schemes in large WSNs by facilitating the use of nodes with simpler hardware. VCS also possess the advantage of having connectivity information implicitly embedded in node coordinates generated based on hop distances.

VCS is a WSN friendly self-organization strategy that provides routing schemes with performance, comparable to alternatives that require costly localization [1][3]-[6]. VC based routing (VCR) schemes generally use greedy forwarding, in which a packet is forwarded to the neighbor closest to the destination. The distance between two nodes for this purpose is typically calculated using L^2 norm in VC space. Early VCR schemes such as LCR [4] and CSR [5] differ in the way they deal with local minima or logical voids in this space and the way backtracking is done.

As each Virtual Coordinate (VC) propagates in the network concentrically from the corresponding anchor, VC systems, appear to lose the sense of (cardinal) directionality that is present in geographic coordinate based systems. Two recent developments go a long way toward overcoming this hurdle by recovering or regenerating directional information lost in VCs. First is the methodology for generating Topology Preserving Maps (TPMs) for WSNs using VCS [8]. TPMs and the associated Topology Coordinates provide a powerful alternative for physical maps and location information. TPMs preserve the neighborhood information, boundaries, voids and even the general shape of networks, but they have neither the exact coordinates nor the distances of the physical layout. TPM and VCS have been combined in a routing scheme called Geo-Logical Routing [9], which generally outperforms even the exact physical location based routing scheme Greedy Perimeter Stateless Routing (GPSR) [10]. The second development is the VC to Directional Virtual Coordinate (DVC) transformation we proposed in [6], which embeds virtual directions within a network. The routability is significantly enhanced by using such directional information in conjunction with VCS [6].

The number of anchors and their placement, i.e., the "anchor selection," is the key factor affecting the performance of any VCS based algorithm, including those for routing [4][5] and TPM generation [7]. Under-placement of anchors results in identical coordinates for different nodes. In general, the routability increases and the likelihood of identical coordinates decreases with the increase of number of anchors. But as [5] indicates over placement of anchors may degrade routing performance. Not only that, the higher the number of anchors, the higher the VC generation cost and higher the length of address field in packets, causing higher energy consumption.

Determining the optimal number of anchors for a network however is not trivial. A further complication is due to the fact that the number of anchors and their optimal placement are highly correlated. Adding the $(n+1)^{st}$ anchor to the optimal placement of n anchors does not in general result in the optimal placement of (n+1) anchors. A '*good*' set of anchors provide high routability, while improper placement of the same number of anchors may cause excessive routing failures due to *local minima* [4]-[7] and nodes with identical coordinates [3][5]. When the node currently holding the packet is unable to find a neighbor that is closer than itself to the destination, the packet is said to be at a local minima. How internal anchors cause local minima that impact routing is addressed in [5], which explains the importance of placing the anchors on the boundary. Existing VC based Routing (VCR) schemes use a costly back tracking phase to overcome such local minima, yet packet delivery is not guaranteed. Schemes in [3][4][12] attempt to place anchors as far apart as possible, or on the boundary. Those schemes are expensive to implement, and do not guarantee to achieve the stated goal. Not all boundary nodes are '*good*' anchors either. Discovering boundary nodes without any localization is not straightforward [13]. Boundary detection for placement of anchors also increases the cost of VCS generation as it involves multiple network floodings [3][4][12]. To manage this complexity, many existing schemes determine apriori, the number of anchors to be placed, e.g., a certain percentage of the nodes, and then attempt to place them on boundary or space them far apart [3][4][13]. Thus determination of the number of anchors together with their placement remains a major challenge. No holistic solutions exist, that determines both the number of anchors and their placement simultaneously based on the network topology.

The first contribution of this paper is a simple, efficient, and also a very effective scheme for anchor placement. The proposed Extreme Node Search (ENS) algorithm simultaneously determines both the number of anchors and their locations. The anchor nodes selected in general are *extreme nodes* of the network. Nodes at corners of a network and nodes that are furthest apart are examples of *extreme nodes*. ENS is a completely distributed scheme that allows a node to self-identify whether it is an anchor or not. It exploits the directionality introduced to VCS by the Directional Virtual Coordinate Systems (DVCS) approach [6]. The quality of anchors selected is evaluated based on improvement of routability achieved with Greedy forwarding and existing VCR schemes Directional Virtual Coordinate Routing (DVCR) [6], Convex Subspace Routing (CSR) [5], Logical Coordinate Routing (LCR) [4]. The performance of these schemes with anchors selected by ENS is also compared with Greedy Perimeter Stateless Routing (GPSR)[10], a geographical coordinates based scheme that requires the knowledge of nodes' geographic coordinates. Results presented for several representative networks demonstrate that ENS anchor selection not only improves the routing performance of existing VCS based routing, but also improves the quality of TPMs generated based on [7]. In the process of developing and providing insight to the anchor placement, we also derive additional properties, including the notion of angle between two vectors in DVC space. If two vectors have $\sim 90^0$ angle between them, they provide a good set of Cartesian axes which can also be used to get a good network map. This is the second contribution of this paper. The proposed TPM technique is computationally less complex than that in [7], which is based on singular value decomposition. The notion of specifying cardinal directions and use of angles, introduced in this paper for the first time, is a radical change from the traditional VC system approaches. This, we believe, will significantly enhance the ability to exploit VCs for networking.

Section II discusses work related to anchor placement and VCS based routing. Then Section III derives two basic properties of DVCS for a full grid. ENS based on DVCS is discussed in Section IV followed by performance evaluation in Section V. Section VI discusses existing TPM generation technique for WSNs. Section VII proposes and evaluates TPM based on DVCS. Finally, Section VIII concludes the paper.

II. Anchor Selection And Routing in Virtual Coordinate Systems

Though good anchor placement is critical for VCS performance, only a few schemes are available for discovering good anchor locations. Anchor placement scheme in [12] proposes placing them on the WSN boundary assuming that the boundary nodes are known. But most existing boundary identification schemes require physical localization, which is expensive. Virtual Coordinate assignment protocol (VCap) [3] proposes generating a virtual topology based on a total of three furthest apart for the entire network. In general, multiple nodes flood the network when self-election of anchors is aimed at optimization of prior coordinates. Moreover, this results in a large number of nodes with identical coordinates. Logical Coordinate based Routing (LCR) [4] combines an anchor placement strategy together with GF. Again, the anchor placement scheme attempts to select the furthest apart nodes, but the number is not restricted to three. A backtracking (BT) algorithm is used when a packet reaches local minima. The mode of data packets- GF/BT, is kept in the packet header while their recently visited nodes are kept in a distributed manner at intermediate nodes. Strategies for evaluating anchors and VCS dimensionality reduction are presented in [8].

Single anchor based VCS called Spanning Path Virtual Coordinate System (SPVCS) in [11] uses a depth-first search algorithm starting from the root node for coordinate generation. Best performance is achieved when the anchor or the root node is placed at the center, as it provides a balanced spanning tree. Convex Subspace Routing (CSR) algorithm [5] moves dynamically among different subspaces of VCS to avoid local minima. Directional Virtual Coordinate Routing

(DVCR) [6] is a routing scheme based on Directional VCS (DVCS) in which a node uses DVCS based greedy forwarding when possible. At a local minima, an approximation for hop distance based on two directional coordinates is used for next node selection.

TABLE I NOTATIONS USED IN THE TEXT AND THEOREMS

Notation	Description
N	Number of network nodes
n_i	Node i
M	Number of anchors ($M << N$)
$A_i, i = 1{:}M$	Set of Anchors
$h_{n_i n_j}$	Minimum hop distance between nodes n_i, n_j
$P_{(i)} = [h_{n_i A_1}, \dots, h_{n_i A_M}]$	Node n_i's VC
$P_{D(i)}$	Node n_i's Directional VC
$P_D^{(i)}$	i^{th} coordinate of DVCS
$\vec{u}_{A_i A_j}$	Unit vector of the directional coordinate w.r.t. anchor A_i and A_j
Φ_{ij}	Angle between $P_D^{(i)}$ and $P_D^{(j)}$
$K_h(n_i)$	Set of nodes in n_i's h-hop neighborhood
g_{xy}	Displacement between node n_x and n_y in DVCS

III. PROPERTIES OF DIRECTIONAL VC SPACE

Consider a 2D network with N nodes and M anchors. VCS of the network characterizes the node n_i by $[h_{n_i A_1}, \dots, h_{n_i A_M}]$, where $h_{n_i A_j} (\geq 0)$ is the shortest path hop distance from node n_i to anchor A_j. The notations used is summarized in Table I.

Each coordinate of DVCS is obtained using a pair of anchors in VCS. The pair of VCs of node n_i with respect to A_j and A_k, $[h_{n_i A_j}, h_{n_i A_k}]$ is transformed to the corresponding directional virtual coordinate (DVC) of node n_i as [6]:

$$f\left(h_{n_i A_j}, h_{n_i A_k}\right) = \left({h_{n_i A_j}}^2 - {h_{n_i A_k}}^2\right)/2h_{A_j A_k} \tag{1}$$

This transformation basically introduces directionality to the space, i.e., $f\left(h_{n_i A_j}, h_{n_i A_k}\right)$ spans negative and positive values, with zero at the midpoint between the two anchors. This restores the directionality lost in original VCs, which corresponds to positive values propagating radially away from each anchor. Therefore, a vector notation can now be introduced. Let $\vec{\boldsymbol{u}}_{\boldsymbol{A_j A_k}}$ be the unit vector in *virtual direction* from A_j to A_k in the virtual domain. The vector component of node n_i in DVCS in the direction $\{A_j, A_k\}$, i.e., from anchor A_j to A_k, is [6],

$$\vec{\boldsymbol{f}}\left(\boldsymbol{h_{n_i A_j}}, \boldsymbol{h_{n_i A_k}}\right) = f\left(h_{n_i A_j}, h_{n_i A_k}\right) \vec{\boldsymbol{u}}_{\boldsymbol{A_j A_k}} \tag{2}$$

$f\left(h_{n_i A_j}, h_{n_i A_k}\right)$ is the magnitude in the direction of $\vec{\boldsymbol{u}}_{\boldsymbol{A_j A_k}}$.

The displacement from nodes n_x to n_y (with respect to the coordinate defined by A_j and A_k), denoted by g_{xy}, is therefore

$$g_{xy} = -g_{yx} = f\left(h_{n_y A_j}, h_{n_y A_k}\right) - f\left(h_{n_x A_j}, h_{n_x A_k}\right) \tag{3}$$

Two properties of DVCS are derived next for a full rectangular grid network as shown in Fig. 1 in which each node communicates with four neighbors. This simple case provides insight into the directional coordinates and helps explain the principles behind the anchor placement algorithm. Even though the assumptions are not compatible with general networks, with odd shapes, voids, etc., the results still can be used as a first order approximation to analyze such networks.

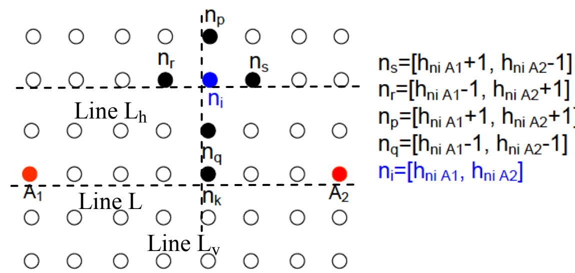


Figure 1. VCs of n_i and its neighbors on a grid with two anchors A_1 and A_2.

Property 1: In a full rectangular grid, when two anchors are placed furthest apart on a connected line L, the displacement between any two adjacent nodes on a line parallel to L (line L_h) is a constant that is greater than unity for each line. Moreover neighbors on a line perpendicular to L (line L_v) have a constant displacement that is less than unity for each line.

Proof: Consider nodes on line L_h passing through nodes n_r, n_i and n_s in Fig 1. The four neighbors of node n_i are n_p, n_q, n_r, and n_s. Latter two neighbors, with DVCs $\left(h_{n_i A_1} + h_{n_i A_2}\right)\left(h_{n_i A_1} - h_{n_i A_2} - 2\right)/2h_{A_1 A_2}$ and $\left(h_{n_i A_1} + h_{n_i A_2}\right)\left(h_{n_i A_1} - h_{n_i A_2} + 2\right)/2h_{A_1 A_2}$ respectively, are on a line parallel to the line through anchors. As the sum of the two VCs, $h_{n_i A_1} + h_{n_i A_2}$, of any node on the line is fixed, the displacements g_{is} and g_{ri} evaluated using (4), are the same:

$$g_{is} = g_{ri} = \left(h_{n_i A_1} + h_{n_i A_2}\right)/h_{A_1 A_2} > 1 \tag{4}$$

Now consider the other pair of neighbors, n_p and n_q, on line L_v, with DVCs $(h_{n_i A_1} - h_{n_i A_2})(h_{n_i A_1} + h_{n_i A_2} + 2)/2h_{A_1 A_2}$ and $(h_{n_i A_1} - h_{n_i A_2})(h_{n_i A_1} + h_{n_i A_2} - 2)/2h_{A_1 A_2}$ respectively. The displacements g_{ip} and g_{qi} (from (2)) are the same:

$$g_{ip} = g_{qi} = \frac{\left(h_{n_i A_1} - h_{n_i A_2}\right)}{h_{A_1 A_2}} = \frac{\left(h_{n_k A_1} - h_{n_k A_2}\right)}{h_{n_k A_1} + h_{n_k A_2}} =< 1 \tag{5}$$

where, n_k is the node at the crossing point for a given vertical line on which n_i lies and the line through the anchors as shown in Fig 1. QED.

Next we define *extreme node* and property 2, which will be used to identify 'good' anchors.

Definition: A node is an extreme node if it is at a local minimum/maximum within its h-hop neighborhoods in its DVC system generated w.r.t two given anchors.

Note that minima occur due to negative values in DVCs.

Property 2: In a full rectangular grid, an internal node cannot be an extreme node.

Proof: Let any two anchors be A_j and A_k. Without loss of generality, consider the transformation at n_i and its 1-hop neighborhood ($K_1(n_i)$) given by (1). If the node n_i is a middle node, then there exist neighbor nodes with VCs ($h_{n_iA_j} \pm a, h_{n_iA_k} \pm a$), where a is 1 or 0. Thus, directional coordinate of any of the neighbors can be expressed as,

$$\left({h_{n_iA_j}}^2 + a \pm 2ah_{n_iA_j} - {h_{n_iA_k}}^2 - a \pm 2ah_{n_iA_k}\right)/2h_{A_jA_k}$$
$$= f\left(h_{n_iA_j}, h_{n_iA_k}\right) \pm \left(2ah_{n_iA_j} + 2ah_{n_iA_k}\right)/2h_{A_jA_k}$$

Thus n_i's coordinate is neither a minimum nor maximum. In contrast, if n_i is an extreme node the only possibility for the neighboring VCs is ($h_{n_iA_j} - a, h_{n_iA_k} - a$). Again a is 0 or 1. Neighbors' coordinates from (1),

$$\left({h_{n_iA_j}}^2 + a - 2ah_{n_iA_j} - {h_{n_iA_k}}^2 - a + 2ah_{n_iA_k}\right)/2h_{A_jA_k}$$
$$= f\left(h_{n_iA_j}, h_{n_iA_k}\right) - \left(2ah_{n_iA_j} - 2ah_{n_iA_k}\right)/2h_{A_jA_k}$$

If $h_{n_iA_j} > h_{n_iA_k}$, where a is 1, n_i is at a maximum while if $h_{n_iA_j} < h_{n_iA_k}$ then n_i is at a minimum. When a is zero, i.e., under identical coordinate situation, both nodes with identical coordinates ($h_{n_iA_j}, h_{n_iA_k}$), are at minima/maxima. QED.

Now consider the circular network with three physical voids in Fig. 2. Let $\{L_h\}$ be the sets of lines that are parallel to line L joining the anchors, i.e., those in $\vec{\boldsymbol{u}}_{\boldsymbol{AB}}$ direction, and $\{L_v\}$ be the set of lines that are perpendicular to L, i.e., those perpendicular to $\vec{\boldsymbol{u}}_{\boldsymbol{AB}}$ direction. We identify $\{L_h\}$ based on the gap between neighbors given by (4) of Property 1. $\{L_v\}$ lines are identified using (5). Figure 2 clearly illustrates that nodes can identify the $\{L_h\}$ and $\{L_v\}$ lines they are on with a high degree of accuracy. Only 65 out of 496 nodes cannot identify their $[L_h, L_v]$ position accurately. This illustrates that the results for full rectangular grid can serve as a good first order approximation. Gray squares are the 65 discordant nodes not compatible with the first order approximation.

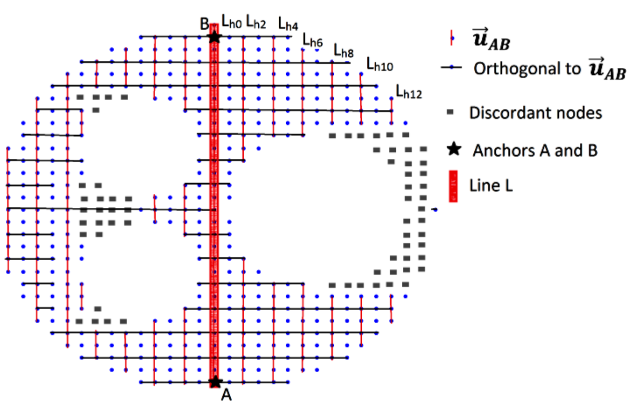


Figure 2. $\{L_h\}$ and $\{L_v\}$ lines of a circular network with three physical voids where anchors are at A and B. Only the even numbered lines in the two directions are shown for clarity.

IV. Extreme Node Search (ENS) - An Anchor

V. Selection Algorithm

Anchor placement has a critical impact on the performance of any VCS based routing algorithm. Determining the optimal number of anchors and their placement are open problems. The fact that the two problems are interdependent makes their solution even more challenging. Optimum number of anchors can be defined as the minimum set of anchors that provide unique VCs for each node and facilitate 100% success in routing using shortest length paths. In this section we propose a simple but effective anchor placement scheme, Extreme Node Search (ENS), that attempts to assign extreme nodes, such as furthest apart and corner nodes of the network as anchors. Results below show that the selected extreme nodes are almost always situated far apart if not furthest apart.

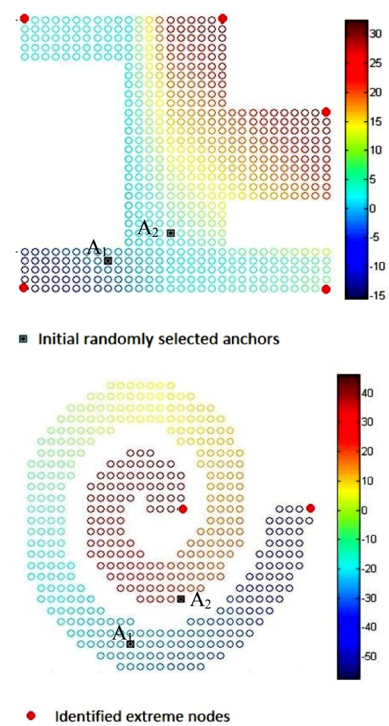


Figure 3. Scatter plot of DVCS coordinate for two sample networks to identify extreme nodes for anchors. Initial anchor pair is (A1,A2), color corresponds to DVCS value, and highlighted nodes are the identified extreme nodes.

Inputs: Neighbors set of $\boldsymbol{n}_i$; $n_j \in K_h(\boldsymbol{n}_i)$,
Output: ENS anchor candidate
Step 1 - Two random nodes initiate floodings to generate a VCS
Step 2 - Each nodes locally generates its DVCS using (1)
Step 3 - Each node checks whether it is a local minimum/maximum in h-hop neighborhood:
If $f(h_{iA_1}, h_{iA_2}) < f(h_{jA_1}, h_{ij})$; $\forall\, n_j \in K_h(\boldsymbol{n}_i)$
n_i is an anchor
End
OR
If $f(h_{iA_1}, h_{iA_2}) > f(h_{jA_1}, h_{ij})$; $\forall\, n_j \in K_h(\boldsymbol{n}_i)$
n_i is an anchor
End
Step 4 – Selected anchor nodes generate the VCS

Figure 4. ENS anchor selection algorithm at a node.

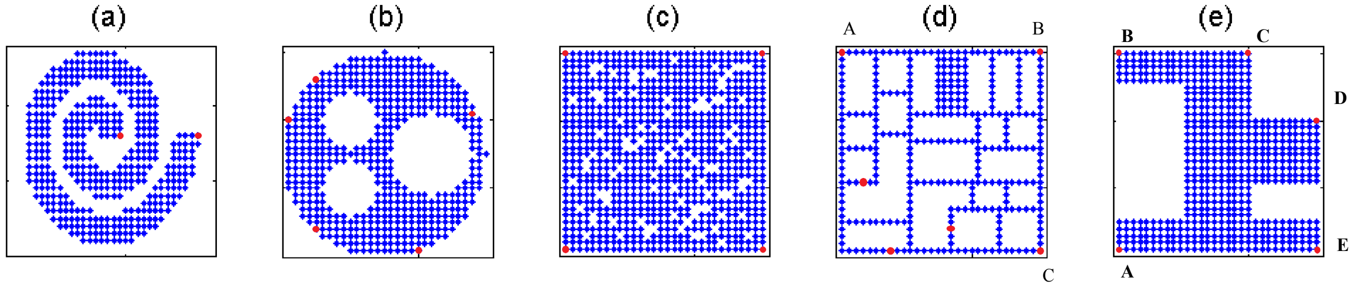


Figure. 5. (a) Spiral shaped network with 421 nodes; (b) a 496-node circular shaped network with three physical voids/holes; (c) A grid based network with 100 randomly missing nodes; (d) a network of 343 nodes mounted on walls of a building; (e) odd shaped network with 550 nodes.

The strategy uses a pair of anchors, which may be randomly selected, to generate a DVC at each node. Fig. 3 indicates the scatter plot of DVC by the initial pair of randomly selected anchors for two example networks, which can be used to identify maxima/minima, i.e., the extreme points of the network.

ENS consists of three steps: First, two randomly selected nodes (A_1 and A_2 in Fig. 3) flood the network thus characterizing each node by two VCs. Then each node evaluates the corresponding DVC using (1). Third, each node evaluates whether it is a local minima/maxima in DVCs, within its h-hop neighborhood $- K_h(\boldsymbol{n}_i)$. h is an important parameter that we can use to restrict the identified anchors. If it is a local maxima or minima in the neighborhood, the node decides to become an anchor of the network. Now a new VCS is generated using these new anchor nodes. ENS is summarized in Fig. 4.

Initially selected anchors can also supplement the selected anchors to be used for routing if necessary. Another fact to notice is that ENS determines the anchor locations as well as the number of anchors for a given network.

VI. Effectiveness of ENS Anchors

In this section, the performance and the effectiveness of the anchor placement is investigated. Five examples shown in Fig. 5, represent a variety of network topologies: (a) spiral shaped network with 421 nodes, (b) a 496-node, circular shaped network with three physical voids/holes [7], (c) a 30x30 grid network with 100 randomly missing nodes, (800 nodes), (d) a network of 343 nodes mounted on walls of a building [8], and (e) an odd shaped network with 550 nodes [8], were used for performance evaluation. Communication range of a node in all five networks is unity. MATLAB® 2010b was used for the computations. We use a four-hop neighborhood, i.e., $h = 4$, for selecting extreme nodes.

The nodes colored in red in Fig. 5 are those identified by the algorithm as the extreme nodes, i.e., local minima/maxima in a 4-hop neighborhood. Thus, they become the anchors for VCS generation. For all the networks except for those in Fig. 5(b) and (d), these very same nodes are identified with a high probability by the scheme irrespective of the initial selection of the two random anchors. The five red nodes in Fig. 5(b) and (d) are for a particular selection of the initial two nodes. This can be attributed to the fact that Fig. 5(b) has a circular boundary, resulting in many possible candidates for extreme nodes. For Fig 5(b) and (d), number of anchors selected varies between 3-8 and 3-10 respectively for different initial anchor selections.

A. Improvement in Greedy Forwarding

Greedy Forwarding (GF) is the underlying mechanism on which existing VC routing schemes [3]-[6] are based. The percentage of packets successfully routed to the destination, is called the Greedy Ratio (GR). A good anchor placement is one that enables high GR, thus, it is a good measure to evaluate the effectiveness of an anchor placement strategy. GR is evaluated for anchors selected by the ENS and is compared with that for several other strategies in Fig. 6. Other anchor placement schemes include: (a) anchors selected anywhere in the network uniformly at random, (b) anchors selected from the boundary nodes (again uniformly at random, similar to that in [3][4]) for the network in Fig. 5(e). The tolerance bars in Fig. 6 indicate the maximum and minimum variation of GR for different random anchor configurations for anchor placement schemes in (a) and (b) above. GR increases from 46.9% to 84.9% as the number of randomly selected anchors from anywhere in

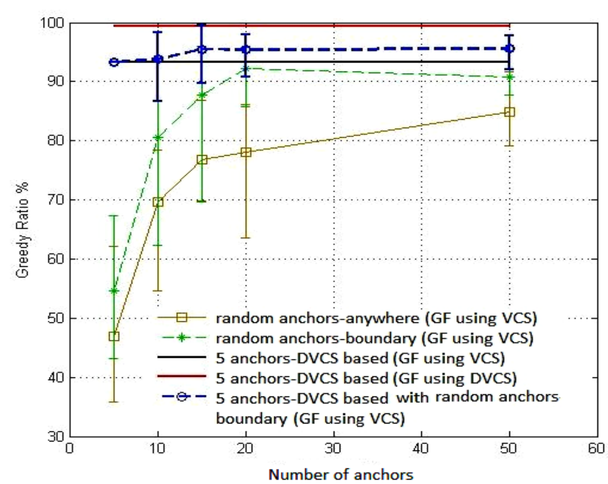


Figure 6. Comparison between, greedy ratio of randomly selected anchors from the boundary/anywhere in the network and that of five anchors selected from ENS for the network in Fig. 5(e).

the network varies from 5 to 50. As the number of boundary anchors increases from 5 to 50, GR increases from 54.5% to 90.7%. GR for the five anchors selected by ENS, i.e., as in Fig. 5(e), is 93.3%, achieving higher routing performance with just five anchors. Note that the horizontal axis of Fig. 5 has no significance for the ENS as it comes up automatically with maximum five anchors. None of the other schemes can achieve a similar performance even with 50 anchors. We have included results only for network in Fig. 5(e) due to limited space. Results for other network configurations also demonstrate a similar improvement.

To evaluate whether the anchor placement provided by ENS can be improved upon, next we add random nodes from the boundary as additional anchors beyond the five already selected. GR, which is 93.3% for the selected five anchors, increased gradually to 95.5% when the total number of anchors is 50 anchors (5 selected anchors from ENS and 45 randomly selected boundary nodes). Thus additional 45 anchors provide just 2% increment in routability. This indicates that anchors from ENS are able to identify the most efficient set of anchors while guaranteeing high routability. Finally GR is evaluated using ENS selected anchors in DVCS based GF. In other words, Greedy decision is based on L^2 distance estimation using entire DVCS. As opposed to VCS, DVCS based GF achieves 99.3% GR under anchors from ENS. Not only that, DVCS based GF outperform VCS based GF under any anchor placement scheme which is an indication of the effectiveness of DVCS.

In evaluating any anchor placement scheme, it is extremely important to consider the impact on communication and computation complexity associated with the number of anchors. For M anchors, the VC generation complexity, i.e., the number of packets needed to generate the VCS is $O(MN)$. The length of address field is $O(M)$. Not only does the ENS selects anchors in the range of 2 – 10 for the given networks in Fig. 5, the routability performance with these few anchors is better than that with 50 anchors

TABLE II R_{AVG}% AND H_{AVG} OF DVCR, CSR, LCR AND GPSR

		Five random anchors			With ENS anchors			GPSR
		CSR	LCR	**DVCR**	CSR	LCR	**DVCR**	
Spiral	R_{AVG}%	49.97	46.9	**92.7**	61.66	67.3	**95.9**	49.1
	H_{AVG}/H_s	0.64	0.64	**1.13**	0.81	0.9	**1.1**	1.65
Grid	R_{AVG}%	61.89	49.7	**93.7**	87.55	52.9	**99.0**	89.5
	H_{AVG}/H_s	0.81	0.78	**1.3**	0.95	0.7	**1.07**	1.05
Circle W holes	R_{AVG}%	43.5	39.7	**90.5**	84.02	75.5	**96.9**	93.8
	H_{AVG}/H_s	0.6	0.69	**1.6**	1.01	0.9	**1.21**	2.46
Building	R_{AVG} %	45.7	40.8	**95.4**	80.14	59.5	**98.7**	97.3
	H_{AVG}/H_s	0.65	0.62	**1.39**	1.06	0.8	**1.18**	1.43
odd NW	R_{AVG} %	70.7	66.0	**93.7**	99.92	93.0	**100**	94.4
	H_{AVG}/H_s	0.88	0.8	**1.3**	1.03	1.00	**1.00**	1.2

(randomly selected, or selected on the boundary), resulting in an order of magnitude reduction in complexity while achieving better performance. selection of anchors on the boundary normally requires apriori knowledge of boundary nodes or use of a boundary node identification scheme that significantly adds to the complexity. Proposed scheme overcomes this hurdle simply relying on the DVCS generated from two random anchors to identify the anchor placement.

B. Improvement in existing routing schemes

Having considered the effectiveness of the proposed anchor placement strategy on GF, next question is whether the existing VC based routing schemes benefit by the ENS based anchors. Thus we evaluate the performance of three existing routing schemes, namely, Convex Subspace Routing (CSR) [5], Logical Coordinate Routing (LCR) [4], and Directional Virtual Coordinate Routing (DVCR) [6]. Comparison is done with ENS based anchor selection Vs. five randomly selected anchors in terms of average routability (R_{AVG}%) and average path length (H_{AVG}) defined as:

$$R_{AVG}\% = \frac{\text{Total \# of packet that reached the destination}}{\text{Total number of packet generated}}\% \quad (6)$$

$$H_{AVG} = \frac{\text{Cumilative number of hops that each packet traversed}}{\text{Total number of packet generated}} \quad (7)$$

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 nodes as the destination. For random anchor placement, R_{AVG}% and H_{AVG} are averaged over 10 different anchor placement configurations. H_s is the shortest hop distance from source to destination, i.e., ideal path length. The routabilities for networks in Fig. 5 are tabulated in Table II for random and ENS based anchor placements. Since CSR requires a minimum of 3 anchors, for the spiral network, two initial randomly selected anchors were also used for routing in addition to two anchors selected by ENS. Moreover in our implementation of LCR, perfect backtracking capability is assumed by saving entire path traversed by each packet at the intermediate nodes. Thus LCR routability reported is more optimistic than that a practical implementation would yield.

Not only do the ENS selected anchors improve the routability of the existing schemes, it also finds shorter paths (See Table II). Routability improvement of CSR is on average 28.4% while that of LCR is 21%. DVCR is improved by 5%. With the anchors from ENS, DVCR outperforms geographical routing scheme - Greedy Perimeter Stateless Routing (GPSR) [10], by 13.4% averaged over all the networks. Unlike GPSR, which requires expensive localization, DVCR relies solely on logical coordinates.

C. Parameter tuning in ENS

The number of anchors chosen depends on the neighborhood size h. For instance, consider the building network in Fig. 5(d) where ENS identifies six anchors when a 4-hop neighborhood (h=4) is used. For an 8-hop neighborhood (h=8), ENS identifies only three anchors, which are marked as A, B, and C in Fig. 5(d). The performance of A, B and C is more or less the same as that with all the ENS anchors. For instance, average Routability of DVCR R_{AVG}% using A, B, and C is 98.8% with 1.23 H_{AVG}/H_s. Main conclusion is 4-hop neighborhood provides a set of anchors with good performance but there exist less number of anchor configuration which provides more or less the same performance under proper choice of *h*.

VII. RELATED WORK-TOPOLOGY PRESERVING MAPS

This section briefly discusses the topology preserving map generation proposed in [7]. Consider a WSN with N nodes, with each node characterized by a vector of VCs, with distance to each of the M anchors ($N >> M$). Let P be the $N \times M$ virtual coordinate matrix of the network. Singular value decomposition of P is

$$P = U.S.V^T \tag{8}$$

where, U, and V are $N \times N$, and $M \times M$ unitary matrices respectively and S is a $N \times M$ matrix where elements S(i,i) are called singular values. Each node thus can be represented by its Principal Components (PC), given by P_{SVD}:

$$P_{SVD} = P.V \tag{9}$$

The second and third columns of P_{SVD} provide a set of 2-dimensional Cartesian coordinates for node positions on a topology preserving map, i.e.,

$$[X_T, Y_T]_{(i)} = [{P_{SVD}}^{(2)}, {P_{SVD}}^{(3)}] \tag{10}$$

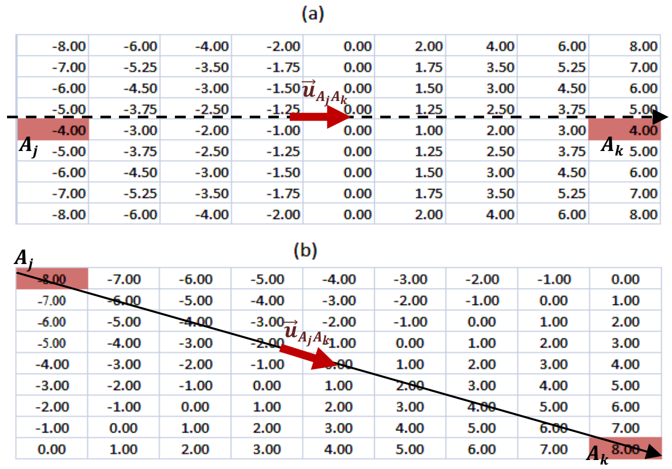
where, ${P_{SVD}}^{(i)}$ is i^{th} column of P_{SVD}, and $[X_T, Y_T]_{(i)}$ is the topological coordinate pair of i^{th} node. Topology Preserving Error (E_{TP}) is defined as the % of nodes out of order/flipped compared to the original network. Results in Section VII show that the anchors from ENS significantly improve the topology preserving maps proposed in [7] in terms of E_{TP}.

VIII. TOPOLOGY PRESERVING MAPS FROM DIRECTIONAL VIRTUAL COORDINATES

Consider the DVCS of a full rectangular grid with two anchors A_j and A_k in Fig. 7(a) and (b). The virtual directions of $\vec{\boldsymbol{u}}_{A_jA_k}$ for different *j* and *k* are indicated by the arrows. As each DVC is generated based on two anchors, it is possible to transform the M virtual coordinates to up to C_2^M DVCs. In practice, however, only a small number of anchor pairs are needed to characterize a network in DVC domain [6].

A. Angle between directional coordinates

As a DVCS contains multiple coordinates, pointing in different virtual directions, it is useful to define the angle between different directions. Let P_D be the $N \times \breve{M}$ matrix containing the entire DVCS, where N is the number of nodes in the network and $\breve{M} \le C_2^M$ is the cardinality in DVC domain. The column $P_D^{(i)}$ corresponds to the i[th] coordinate. The k[th] row $(P_D)_{(k)}$ represents the DVC tuple of node n_k. Consider two coordinates in DVCS, $P_D^{(i)}$ and $P_D^{(j)}$ of the entire network. The angle between i^{th} and j^{th} coordinates is defined as



(a)

-8.00	-6.00	-4.00	-2.00	0.00	2.00	4.00	6.00	8.00
-7.00	-5.25	-3.50	-1.75	0.00	1.75	3.50	5.25	7.00
-6.00	-4.50	-3.00	-1.50	0.00	1.50	3.00	4.50	6.00
-5.00	-3.75	-2.50	-1.25	0.00	1.25	2.50	3.75	5.00
-4.00	-3.00	-2.00	-1.00	0.00	1.00	2.00	3.00	4.00
-5.00	-3.75	-2.50	-1.25	0.00	1.25	2.50	3.75	5.00
-6.00	-4.50	-3.00	-1.50	0.00	1.50	3.00	4.50	6.00
-7.00	-5.25	-3.50	-1.75	0.00	1.75	3.50	5.25	7.00
-8.00	-6.00	-4.00	-2.00	0.00	2.00	4.00	6.00	8.00

(b)

-8.00	-7.00	-6.00	-5.00	-4.00	-3.00	-2.00	-1.00	0.00
-7.00	-6.00	-5.00	-4.00	-3.00	-2.00	-1.00	0.00	1.00
-6.00	-5.00	-4.00	-3.00	-2.00	-1.00	0.00	1.00	2.00
-5.00	-4.00	-3.00	-2.00	-1.00	0.00	1.00	2.00	3.00
-4.00	-3.00	-2.00	-1.00	0.00	1.00	2.00	3.00	4.00
-3.00	-2.00	-1.00	0.00	1.00	2.00	3.00	4.00	5.00
-2.00	-1.00	0.00	1.00	2.00	3.00	4.00	5.00	6.00
-1.00	0.00	1.00	2.00	3.00	4.00	5.00	6.00	7.00
0.00	1.00	2.00	3.00	4.00	5.00	6.00	7.00	8.00

Figure 7. DVCS coordinates with respect to anchor pairs: (a) A1, A2, and b) A3,A4, for a 9x9 rectangular grid. Arrow indicates the direction of the unit vector.

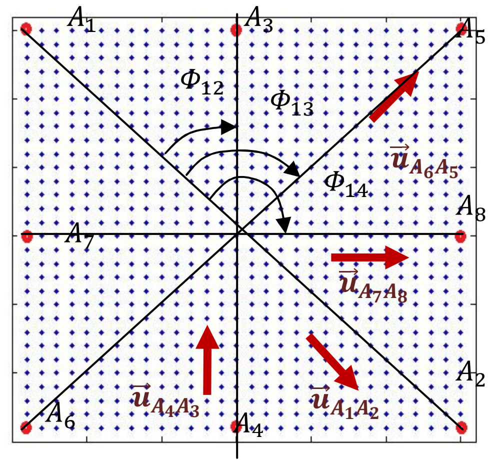


Figure 8. Example to illustrate the angles between different unit vectors $\vec{\boldsymbol{u}}_{A_1A_2}$, $\vec{\boldsymbol{u}}_{A_3A_4}$, $\vec{\boldsymbol{u}}_{A_5A_6}$,and $\vec{\boldsymbol{u}}_{A_7A_8}$

$$\Phi_{ij}=\Phi_{ji}= \cos^{-1}\left(\left(P_D^{(i)}\right)^T P_D^{(j)} / |P_D^{(i)}||P_D^{(j)}|\right) \tag{11}$$

where $|P_D^{(i)}| = \sqrt{\left(P_D^{(i)}\right)^T P_D^{(i)}}$. Consider the example network in Fig. 8 in which 8 anchors are selected in pairs to form four DVCs. The i[th] row of P_D is,

$[f(h_{n_iA_1}, h_{n_iA_2}), f(h_{n_iA_3}, h_{n_iA_4}), f h_{n_iA_5}, h_{n_iA_6}, f(h_{n_iA_7}, h_{n_iA_8})]$, thus columns one through four correspond to four virtual directions $\vec{\boldsymbol{u}}_{A_1A_2}$, $\vec{\boldsymbol{u}}_{A_3A_4}$, $\vec{\boldsymbol{u}}_{A_5A_6}$,and $\vec{\boldsymbol{u}}_{A_7A_8}$. The angles between different pairs of columns of P_D from (11) are Φ_{12}, Φ_{13}, and Φ_{14} are 46.22^0, 90^0 and 133.77^0 respectively. If two of these vectors are orthogonal, they can be used to generate a 2-D topology map of the network. For instance in Φ_{13}, in Fig. 8 corresponding to vectors $\vec{\boldsymbol{u}}_{A_1A_2}$, $\vec{\boldsymbol{u}}_{A_5A_6}$ is 90^0 and so is that between ($\vec{\boldsymbol{u}}_{A_3A_4}$, $\vec{\boldsymbol{u}}_{A_7A_8}$).

Some of the angles between directional coordinates of Fig 9(a) are tabulated in Table III. Anchors are selected using ENS. For example anchors A,C, and D provide three directional coordinates where the directions intersects and form a triangle. Addition of the angles $\emptyset_{AD,DC}(94.9^0)$, $\emptyset_{AD,AC}(16.7^0)$,$\emptyset_{CD,AC}$ (68.4^0) is ~180^0.

TABLE III SOME OF THE ANGLES BETWEEN COORDINATE PAIRS IN FIG. 9(a)

$\emptyset^0$	AC	AB	AE	AD	CB	CE	CD	BE
AB	39.8	-	-	-	-	-	-	-
AE	76.2	101.8	-	-	-	-	-	-
AD	16.7	48.5	64.1	-	-	-	-	-
CB	103.3	63.5	130	109.5	-	-	-	-
CE	152.7	150.0	76.4	138.8	94.2	-	-	-
CD	111.6	121.4	54.0	94.9	106.0	49.7	-	-
BE	124.6	158.7	56.8	114	128.4	34.2	48.4	-
BD	86.0	124	41.5	75.3	161	73.5	54.7	41
ED	21.9	31.9	94.4	30.3	90.4	165	115	139

B. In-Network realization of angle estimation

Angle evaluation in (11) results in the angle between two datasets corresponding to two selected directional coordinates. If a subset of data points, which is a good representation of the entire data set, is selected for two coordinates, still the angular evaluation should hold. Thus anchors VCS, $Q_{M\times M}$ can be used for estimation of angles between two directions. First each node evaluates the DVC matrix Q_D corresponding to Q. i^{th} column of $Q_D \equiv Q_D^{(i)}$ is a subset of data points of i^{th} coordinate of DVCS $P_D^{(i)}$. Therefore, approximate angle between i^{th} and j^{th} coordinates can be evaluated as,

$$\breve{\Phi}_{ij}=\breve{\Phi}_{ji}= \cos^{-1}\left(\left(Q_D^{(i)}\right)^T Q_D^{(j)}/|Q_D^{(i)}||Q_D^{(j)}|\right) \qquad (12)$$

where, $|Q_D^{(i)}| = \sqrt{\left(Q_D^{(i)}\right)^T Q_D^{(i)}}$. The error of this approximation is defined as,

$$E_\phi = \sum_{\forall i,j} |\breve{\phi}_{ij} - \phi_{ij}|/\phi_{ij}$$

E_ϕ for Fig 9(a) is 0.08 and for Fig 5(b) is 0.066, indicating that the in-network realization provides sufficiently accurate results for practical purposes. Worst case transmission complexity of informing the anchors VCS to the entire network is $O(MN)$, which is required by proposed DVCS and SVD based in-network realization of TPM approaches.

C. DVCS based Topology Preserving Maps

With the definition of an angle between two DVCS, next we propose a new method for TPM generation based on the angles between different DVCs. The basic concept is that near orthogonal directions of DVCS provide a good set of axis for obtaining a topology map of the network. For example the TPM of the network in Fig 9(a) is shown in Fig. 9(b). The map in Fig. 9(b) is based on coordinates

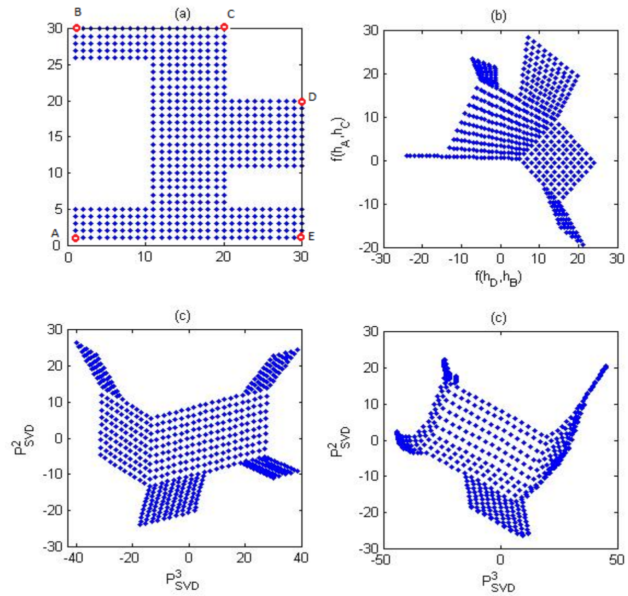


Figure 9. (a) Odd network. Anchors identified by ENS are indicated in red; (b) TPM using near orthogonal DVCs; (c) SVD based TPM using the VCS generated using anchors identified by ENS; (d) SVD based TPM using the VCS generated using 10 randomly placed anchors.

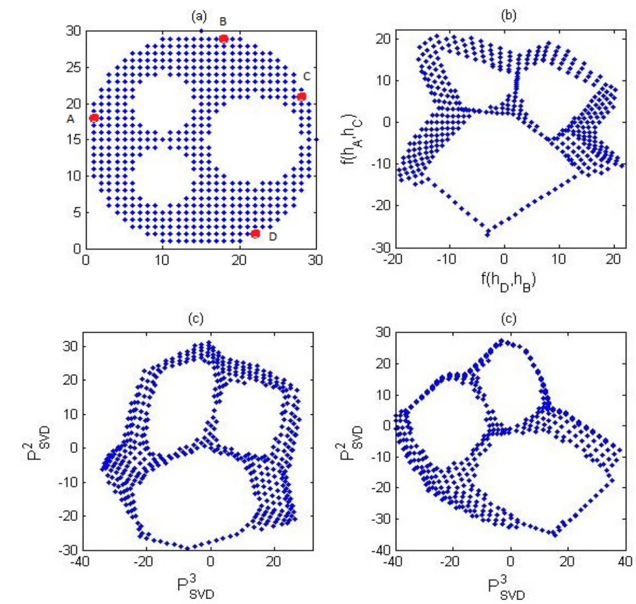


Figure 10. (a) Circular network with voids. Subset of anchors identified by ENS are indicated in red; (b) TPM using near orthogonal DVCs; (c) SVD based TPM using the VCS generated using anchors identified by ENS; (d) SVD based TPM using the VCS generated using 10 randomly placed anchors.

$(f(h_B,h_D), f(h_A,h_C))$ which has $\emptyset_{DB,AC}$ 86^0 (Table II). The resulting topology preserving error E_{TP} [7] is 1.32%. Fig. 9(c) is the TPM using the SVD based scheme as in [7], but with anchors selected by ENS, while Fig. 9(d) is the TPM from SVD on VCS generated using randomly selected 10 anchors, exactly as proposed in [7]. E_{TP} for the former is 0% while the latter is 2.49% averaged over 10 random anchor placement configurations, which clearly indicate the effectiveness of anchor selection of ENS. TPM generated from DVCS has less E_{TP} compared to that with SVD on a VCS with randomly selected anchors.

Similarly TPM for Fig 10(a) generated from near orthogonal DVCs ($\emptyset_{DB,AC}$= 94.12^0), corresponding to anchor pairs (D,B) and (A,C), is shown in Fig 10(b). With an E_{TP} of 0.407%, it achieves better accuracy than the TPM generated from SVD with random anchor placement described in [8] with E_{TP} of 1.5116%, averaged over 10 random anchor placement configurations (See Fig. 10(d)). Again, the ENS based anchor selection improves the TPM generated with SVD based approach, as in Fig. 10(c) reducing E_{TP} to 0.1279%.

Performance of both TPM generation techniques highly depends on the anchor selection and the number of anchors. Hence, the ENS reduces the transmission, memory and computational complexities of both TPM approaches providing impressive E_{TP}.

IX. CONCLUSION

Performance of Virtual Coordinate based algorithms depend heavily on the number of anchors and their placement. We proposed a novel strategy that identifies the extreme nodes in a network as anchors, and thereby minimizes the impact due to local minima problem. Many of these extreme nodes lie on the boundary and are also spaced far apart. Performance evaluation demonstrates that ENS results in significant improvement in performance in terms of Greedy Ratio in Greedy Forwarding with an order of magnitude less anchors compared to randomly placed anchors even on the boundary. It also significantly improves the routability and path lengths of existing VC domain routing schemes, Convex Subspace Routing (CSR), Logical Coordinate Routing (LCR) and Directional Virtual Coordinate Routing (DVCR). Fascinatingly, with the proposed anchor selection, DVCR outperforms geographic routing scheme Greedy Perimeter Stateless Routing (GPSR), in spite of latter's use of node location information. Properties of DVCS domain were investigated by introducing a vector notation. By selecting two directions that are orthogonal to each other, we demonstrated the creation of a topology map of the network. The new DVC based approach for topology map generation involves significantly lower computation complexity compared to existing schemes.

An advantage of nodes knowing TPM and topology coordinates is that even when some nodes including anchors fail in the network, the topological coordinates need not be recalculated. Topological coordinates approximate the physical layout coordinates, albeit with distortions to take into account connectivity. Physical coordinates do not depend on the existence of other nodes, while traditional virtual coordinates have to be recalculated to maintain their validity. Topological coordinates, while initially calculated based on virtual coordinates, maintain a valid topological representation even when a moderate amount of nodes fail. Another consequence of the work presented in this paper is the ability to specify cardinal directions and use angles in VC domain. This is a radical change from the traditional VC system approaches, which we believe will significantly enhance the ability to exploit VCs for networking.

ACKNOWLEDGEMENTS

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

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] Bachrach, J., and Taylor, C. Localization in sensor networks. Ch. 9, Handbook of Sensor Networks. Stojmenovic (Editor), John Wiley 2005.

[3] 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.

[4] 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.

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

[6] D.C. Dhanapala and A.P. Jayasumana, "Directional virtual coordinate systems for wireless sensor networks," Proc. IEEE International Conference on Communications (ICC 2011), June 2011.

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

[8] D.C. Dhanapala and A.P. Jayasumana, "Dimension reduction of virtual coordinate systems in wireless sensor networks," Proc. IEEE Globecom 2010 - Ad-hoc and Sensor Networking Symposium, Dec. 2010.

[9] D. C. Dhanapala and A. P. Jayasumana, "Geo-logical routing in wireless sensor networks," Proc. 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), June 2011.

[10] 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.

[11] K. Liu and N. Abu-Ghazaleh, "Stateless and guaranteed geometric routing on virtual coordinate systems," Proc. 5th IEEE Int. Conf. on Mobile Ad Hoc and Sensor Systems (MASS)2008, Sept. 2008, pp. 340-346.

[12] 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.

[13] O. Saukh, R. Sauter, M. Gauger, and P. Jose´ Marro´ N, "On boundary recognition without location information in wireless sensor networks," ACM Transactions on Sensor Networks, June 2010.