Article

Localization in Wireless Sensor Networks and Anchor Placement

Thomas Kunz * and Benjamin Tatham

Systems and Computer Engineering, Carleton University, 1125 Colonel By Drive, Ottawa, ON K1S 5B6, Canada; E-Mail: tatham@ieee.org

* Author to whom correspondence should be addressed; E-Mail: tkunz@sce.carleton.ca; Tel.: +1-613-520-3573.

Abstract: Applications of wireless sensor network (WSN) often expect knowledge of the precise location of the nodes. Many different localization protocols have been proposed that allow nodes to derive their location rather than equipping them with dedicated localization hardware such as GPS receivers, which increases node costs. We provide a brief survey of the major approaches to software-based node localization in WSN. One class of localization protocols with good localization performance patches together relative-coordinate, local maps into a global-coordinate map. These protocols require some nodes that know their absolute coordinates, called anchor nodes. While many factors influence the node position errors, in this class of protocols, using Procrustes Analysis, the placement of the anchor nodes can significantly impact the error. Through simulation, using the Curvilinear Component Analysis (CCA-MAP) protocol as a representative protocol in this category, we show the impact of anchor node placement and propose a set of guidelines to ensure the best possible outcome, while using the smallest number of anchor nodes possible.

Keywords: WSN; localization protocols; protocol performance; anchor node placement; CCA-MAP

1. Introduction

Scientists, engineers, and researchers use wireless sensor networks (WSN) for a wide array of applications. Many of these applications rely on knowledge of the precise position of each node.

While some may only require relative coordinates within the network, most biological, geophysical, and other scientific applications require coordinates on a global coordinate system. Perhaps the obvious solution is for each node in the network to be equipped with GPS or other location positioning services. However, constraints on cost, power consumption, as well as visibility of satellites dictate the need for an alternative solution.

Many protocols have been proposed to calculate the relative positions of nodes in a network, see for example [1,2]. The next section provides a classification and brief survey of localization protocols proposed in the literature. Procrustes analysis is a common method to convert from relative to global coordinates, requiring some of the nodes to have a local source of global coordinates. This can be achieved by operators recording the global coordinates during network deployment, by embedding a GPS receiver in a subset of the nodes, or some other means. We call these enhanced nodes anchors. After the initial review of localization protocols, focusing on a specific approach to node localization, we explore the effect of anchor node placement within the network on the overall localization errors, on a network-wide basis. This provides network planners with a set of general rules to minimize the number of anchor nodes required while avoiding poor node localization.

During previous work designing localization protocols, researchers typically placed anchors at random within the network [3,4]. They then simulate the network multiple times with different anchors in order to statistically exclude anchor node placement from their results [1,2]. However, our initial investigations and simulations demonstrate however that the placement of anchor nodes in the network does have an often dramatic effect on the localization error. The two plots shown in Figure 1 graphically establishes that anchor node position does make a difference. Each plot shows the same network with a different choice of three anchors. A line is drawn between the actual and calculated position of each node to visualize the localization error. The circles show the radio range of each anchor, and a triangle is drawn between the three anchors used in this example.

Figure 1. Sample localization results. (**a**) reasonable; (**b**) poor.

Max error: 1.4675 Mean error: 0.4825

(**a**)

Max error: 17.0875 Mean error: 6.5325

(**b**)
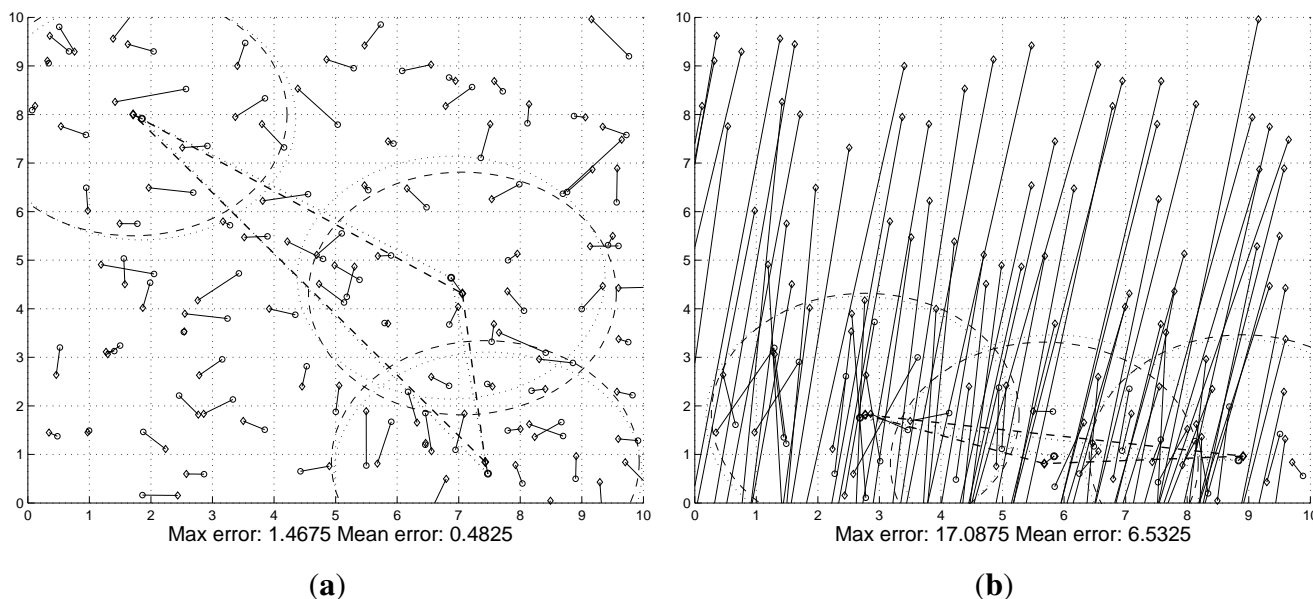
The two plots shown are taken from a set of 100 randomly chosen anchor sets. While the first choice has reasonable errors, the second choice has an error more than ten times the mean error of the first. The two anchor set triangles do not immediately show an obvious reason that could explain this drastic change in error.

Since there is a chance that the localization errors can be large due to anchor node placement, scientists and engineers need to know what degree of accuracy the localization algorithm provides in many cases. First of all, if a network designer has an existing sensor network or is planning to deploy one, they need to know where to place the anchor nodes to ensure localization accuracy. While the freedom of where to place the anchor nodes may be constrained due to physical factors, the network designer still must be able to choose anchor placements wisely. Secondly, even if the network is already deployed, the guidelines can provide confidence that the localization results are good enough and the research resulting from the sensed data itself is valid with respect to location of the sensed data.

Some papers have touched upon anchor node placement [5–8], which we review in Section 3. These few studies are quite limited and typically focus on particular use cases or localization algorithms. We have yet to come across a comprehensive study of the anchor node placement, and this paper provides such a study of possible anchor node placements and their effects on overall network localization accuracy. Specifically, this paper establishes two guidelines based on metrics that a user of a wireless sensor network can calculate at any time during network planning or deployment, based on information readily available to them. These guidelines provide a way to exclude extremely poor localization results and ensure that the localization results fall within a range of errors that are statistically insignificant. We do not address a variety of other possible factors that may may also influence localization error, such as overall properties (size, density, *etc.*) of the network.

At a high level, the guidelines are a result of the fact that the probability of extremely high localization error results from anchor nodes being roughly in a geographically straight line. As the anchor nodes are spread out from a straight line, the probability of high errors decreases, leaving network designers a relatively simple chore when choosing anchor nodes locations.

While this paper is a comprehensive study of anchor node placements when using Procrustes analysis, we focus on a specific set of network design assumptions. We assume the network planners want to minimize the number of anchors since they consume more power and cost more. We therefore use the minimum number of anchors of three to provide two-dimensional coordinates. Finally, we assume that the network density is sufficient to provide a fully connected network, meaning that there is a single connected graph of all the nodes in the network.

The rest of this paper is organized as follows: Section 2 surveys different approaches to software-based node localization. Section 3 presents the limited related work on anchor node placement. Section 4 introduces CCA-MAP, the localization protocol that we use to experimentally explore the impact of anchor node placements (and which was used in deriving the motivating example in Figure 1). Section 5 outlines various proposed anchor node placement metrics and summarizes how they perform. The cause of extreme edge cases is explored in more detail in Section 6. Section 7 applies the derived guidelines to networks with different shapes, and Section 8 summarizes the results and suggests future research.

2. A Survey of Localization Algorithms

Many localization algorithms have been proposed for WSNs. In this section, we first provide a classification of localization techniques in static WSNs. Then, papers presenting algorithms proposed for mobile WSNs are reviewed and finally papers focusing on testbed implementation for WSNs are discussed.

2.1. Centralized vs. Distributed

Localization algorithms can be categorized as centralized [5,9] or distributed [10,11] algorithms based on their computational organization. In centralized algorithms, nodes send data to a central location where computation is performed and the location of each node is determined and sent back to the nodes. The drawbacks of centralized algorithms are their high communication costs and intrinsic delay. In most cases, these costs increase as the number of nodes in the network increases, thus making centralized algorithms inefficient for large networks. As a result, distributed algorithms that distribute the computational load across the network to decrease delay and to minimize the amount of inter-sensor communication have been introduced [9]. In distributed algorithms, each node determines its location by communication with its neighboring nodes. Generally, distributed algorithms are more robust and energy efficient since each node determines its own location locally with the help of its neighbors, without the need to send and receive location information to and from a central server. Distributed algorithms however can be more complex to implement and at times may not be possible due to the limited computational capabilities of sensor nodes.

2.2. Range-Free vs. Range-Based

For determining the location of a sensor node, two types of techniques exist: range-free [3,12,13] and range-based [14–17]. Range-free techniques use connectivity information between neighboring nodes to estimate the nodes' position while range-based techniques require ranging information that can be used to estimate the distance between two neighboring nodes. On the one hand, range-free techniques do not require any additional hardware and use proximity information to estimate the location of the nodes in a WSN, and thus have limited precision. On the other hand, range-based techniques use range measurements such as time of arrival (ToA), angle of arrival (AoA), received signal strength indicator (RSSI), and time difference of arrival (TDoA) to measure the distances between the nodes in order to estimate the location of the sensors.

2.3. Anchor-Based vs. Anchor-Free

Another classification of localization algorithms for WSNs is based on whether or not external reference nodes (*i.e.*, anchors) are needed. These nodes usually either have a GPS receiver installed on them or know their position by manual configuration. They are used by other nodes as reference nodes in order to provide coordinates in the absolute reference system being used.

Anchor-based algorithms [1,2,10,18] use anchor nodes to rotate, translate and sometimes scale a relative coordinate system so that it coincides with an absolute coordinate system. In such algorithms, a fraction of the nodes must be anchor nodes or at least a minimum number of anchor nodes are required for adequate results. At least three noncollinear anchor nodes for 2-dimensional spaces and four noncoplanar anchor nodes for 3-dimensional spaces are required. The final coordinate assignments of the sensor nodes are valid with respect to a global coordinate system or any other coordinate system being used. A drawback to anchor-based algorithms is that another positioning system is required to determine the anchor node positions. Therefore, if the other positioning system is unavailable, for instance, for GPS-based anchors located in areas where there is no clear view of the sky, the algorithm may not function properly. Another drawback to anchor-based algorithms is that anchor nodes are expensive as they usually require a GPS receiver to be mounted on them. Therefore, algorithms that require many anchor nodes are not very cost-effective. Location information can also be hard-coded into anchor nodes, however, in this case, careful deployment of anchor nodes is required, which may be very expensive or even impossible in inaccessible terrains.

In contrast, anchor-free localization algorithms [19,20] do not require anchor nodes. These algorithms provide only relative node locations, *i.e.*, node locations that reflect the position of the sensor nodes relative to each other. For some applications, such relative coordinates are sufficient. For example, in geographic routing protocols, the next forwarding node is usually chosen based on a distance metric that requires the next hop to be physically closer to the destination, a criteria that can be evaluated based on relative coordinates only.

2.4. Individual vs. Collaborative Localization

Localization protocols also differ in their basic approach to calculating node position. In one class of protocols, nodes individually determine their location, using information collected from other nodes, typically involving trilateration, triangulation, or multilateration [21]. In a straightforward way, direct reach of at least three anchor nodes is needed for a node to compute its location coordinates [3,13,21]. In computing the position using any of the above methods, algorithms often employ iterations [4,21], to start from the anchor nodes in the network and to propagate to all other free nodes calculating their positions. One of the problems of this approach is its low success ratio when the network connectivity level is not very high or when not enough well-separated anchor nodes exist in the network. To localize all the nodes, these algorithms quite often require that 20%–40% of the total nodes in the network be anchor nodes [13,22], unless anchor nodes can increase their signal range [3].

To solve the problem of demanding large numbers of anchor nodes, some approaches apply limited flooding to allow reach of anchor nodes in multiple hops, and to use approximation of shortest distances over communication paths as the Euclidean distance [13]. However, such a hop-based distance approximation works rather poorly in anisotropic networks, introducing large position errors. In many scenarios, they do not seem to significantly reduce the number of anchor nodes either. The high network connectivity levels required for success of such algorithms also give rise to practical concerns, as dense neighborhoods often severely impede radio network throughputs. In addition to the issues of large

number of anchor nodes and high connectivity level, the accumulated location errors also need to be dealt with in this type of approach [4] to maintain the accuracy of position estimates.

Cooperative localization schemes take a quite different approach, formulating the localization problem as joint estimation problems. Instead of using only the constraints between the sensor nodes and anchor nodes as in trilateration, triangulation, or multilateration, cooperative solutions jointly utilize as much as possible the connectivity and distance constraints among all the nodes and apply optimization techniques to derive location coordinates. Thus the cooperative method often needs fewer anchors and applies more constraints in generating a better position projection. In [1,9,10], the node positions are calculated using the connectivity constrains applying a technique called Multi-Dimensional Scaling (MDS). In [5], inter-node distance measurements are modeled as convex constraints. Then linear programming methods were adopted to estimate the locations of free nodes. In our own prior work [2,23], we applied Curvilinear Component Analysis, an efficient non-linear mapping technique, to jointly determine node locations, based on connectivity constraints. These cooperative localization methods are often quite powerful as they require only a minimum number of anchor nodes and produce highly accurate results, whether the network is isotropic or not.

2.5. Mobile WSN Localization Algorithms

Recently, the subject of mobility in WSNs has gained much interest due to the increasing number of applications that require mobile sensor nodes. Animal tracking, logistics applications, and elderly healthcare home monitoring are but a few of such applications. In addition, studies conducted on introducing mobility in WSNs have demonstrated an overall improvement not only by increasing the overall network lifetime, but also by improving the data capacity of the network as well as addressing delay and latency problems [24,25]. As a result, many researchers have started to investigate the concept of mobility in WSNs. A number of algorithms that can estimate the location of mobile nodes within WSNs have been developed [20,25–32]. Some algorithms, such as our own prior work on iCCA-MAP reported in [26,27] assume that the anchor nodes are static and one a few sensor nodes are mobile. Others, such as the Mobile Anchor Point (MAP) algorithm [33], use mobile anchor nodes to localize static WSN nodes. A last group of papers allow both the anchors and the mobile nodes to be mobile, such as the Extended Mobile Anchor Point (EMAP) algorithm [30] and the work by Hu and Evans [29] or Stevens-Navarro *et al.* [32].

2.6. WSN Localization Algorithms Testbed Implementations

Most proposed protocols are evaluated via simulations to demonstrate the effectiveness of their localization algorithms. However, some researchers also implement their work in a testbed to experimentally validate their localization protocol. The next paragraphs highlight some works involving real testbed measurements.

Chenji *et al.* [34] proposed a fuzzy logic based approach for mobile node localization in challenging environments. Their paper is mainly based on simulation results but they also present a proof-of-concept implementation on hardware. Similarly, the authors in [35] deploy and study an algorithm called ROCRSSI+ based on simulation and implementation results.

Another study, presented in [36], investigates the performance of different localization algorithms (namely ML, Min-Max, Multilateration and ROCRSSI) based on radio signal strength measurements in an indoor environment. Likewise, a second paper [37] analyzed an outdoor RF-based localization algorithm based on a mobile beacon.

In [38], the authors focus on applications in harsh and dynamic factory automation environments. In fact, they developed a testbed in order to characterize the radio signal propagation model in order to correlate the RSSI and range empirically.

Finally, in our own prior work, we studied the implementation of CCA-MAP [2,23] in a testbed of TinyOS-based motes, as reported in [39]. The results show that the localization error in the testbed confirms the strong performance of CCA-MAP.

3. Related Work on Anchor Node Placement

While much attention has been paid to localization accuracy and computational effort, the importance of intelligent anchor node placement is often recognized, but dismissed as future study.

Often, authors will come across anchor placement by accident and discuss it based on their own empirical evidence. Shang *et al*. [1] and Li *et al*. [2] both choose anchors at random within the network. Shang *et al*. do mention that a co-linear set of anchors chosen in one example "represents a rather unlucky selection", without supporting evidence of why this is unlucky.

Earlier work by Doherty *et al*.[5] requires anchor nodes to be placed at the edges, and ideally at the corners of the network. In this case, however, the algorithm is a simple constraint problem. One constraint requires that all the unknown nodes be placed within the convex hull of the anchors, and therefore, better results are obtained when anchors are at the corners.

While few, there have been a some explicit studies of anchor node placement. Hara *et al*. [40] propose a method of choosing anchor node locations to achieve a specific accuracy target. The proposal, however, only applies to rectangular network areas and anchor nodes must be placed at the center of a sub-rectangle of the original rectangle when divided into equal sized rectangles. Further, it assumes simple RSSI-based localization.

Ash *et al*. [6] provide analytical proof that placing anchor nodes uniformly around the perimeter of a network provides the best results, in the absence of any other information about the sensor node positions. However, again this assumes a rectangular network, and more importantly a simple localization algorithm like [5] or other multilateration techniques. When using all inter-node distances at once, as in MDS-MAP [1] and CCA-MAP [2], this analysis breaks down.

Karl and Willig [7] dedicate an, albeit short, sub-chapter to the *Impact of anchor placement* in their book. Referring [5] and [41], again they defer to perimeter anchor placement as the optimal choice. Unfortunately, the technique proposed involves adaptive deployment, whereby a mobile node with absolute positioning available, like GPS, wanders through the network and attempts to determine the optimal anchor placements as it travels. For the purposes of a priori planning, this technique is not feasible.

Cheng *et al*. [42] present a novel technique to handle the effects of adverse anchor placement, specifically in clumps. The algorithm, *HyBloc*, is a hybrid of MDS and proximity-distance map

(PDM) [43]. HyBloc combines the two algorithms to draw on the best aspects of each. Namely, PDM is shown to have good performance in anisotropic networks, while MDS performs well even with few anchors. Anisotropic networks have irregular shapes or holes in the network connectivity. HyBloc, therefore, works by using MDS to add artificial, secondary anchor nodes in specific isotropic areas of the network, and then uses PDM to complete the overall localization for the entire network.

Another study focuses on the effect of indoor conditions and anchor placement as it relates to RSSI and other radio propagation measurements [44]. The experiments were conducted in a small, enclosed space, and anchor nodes were placed either on the ceiling or the floor of the room. The study concludes that anchor nodes on the ground are better for monitoring moving people in the room, the extension of which is that anchor nodes need to be in the same plane as the nodes they are being used to locate.

More recently, reference [8] studied anchor node placement for passive localization. Here, nodes passively listen to global events such as thunder, lightning, seismic waves, *etc*. They found that, for effective passive localization, the optimal placement of the anchor nodes is at the center of the network in such a way that no three anchor nodes share linearity. The more the non-linearity, the better the localization. These results are closest to our own, in that they do not restrict the placement of anchor nodes to the perimeter of the network. However, in our results, we show that anchor nodes can also provide good localization results when placed at the perimeter. Also, we found that there there is a limit to the benefit that can be achieved by careful anchor node placement, with specific lower bounds on the degree of collinearity of anchor nodes sufficient to achieve optimal localization results.

Overall, the previous studies on anchor placement are limited. Specifically, they focus on particular use cases and assumptions that are different from the ones here. The overarching theme of the studies, though, is to place the anchors at the edges of the network. Despite the different assumptions, this idea makes sense from a purely geometric point of view, and is therefore used as the initial hypotheses in this paper.

4. The CCA-MAP Algorithm

In this section, the CCA-MAP [2,23] algorithm is briefly introduced as it is the basis for the experimental studies we report later on. CCA-MAP was chosen as localization algorithm since it is among the best performing algorithms, with the following characteristics [23]:

- High localization accuracy, even when using only the minimal number of required anchor nodes.
- Where available, supports the use of ranging information, resulting in even better localization performance (assuming that the ranging measurements are reasonably accurate).
- Flexibility whether to run it as a distributed or a centralized algorithm.
- Has a variant, iCCA-MAP [26,27], that can localize nodes in a mobile WSN.

CCA-MAP is a cooperative node localization algorithm, which applies an efficient non-linear data mapping technique called Curvilinear Component Analysis (CCA) [45], to localize nodes in a WSN. Cooperative localization schemes formulate the localization problem as a joint estimation problem and apply optimization techniques to derive location coordinates considering all constraints on inter-node distances, rather than considering only constraints between the sensor nodes and anchor nodes.

The CCA-MAP scheme builds local maps at each node in the network and then patches them together to form a global map. CCA is employed in computing the node coordinates in the local map. Each node computes its local map using only the local information. If accurate ranging capability is available in the network, local distance between each pair of neighboring nodes is measured and known. Otherwise, only connectivity information is used, assigning a value of 1 to the edge between each neighboring pair of nodes. Then, a distance matrix for all the nodes in the R hop neighborhood of node x can be constructed using the shortest distance matrix as approximation. CCA's reduction technique can generate quite accurate results with a reasonably accurate distance matrix of a small size. In addition, a smaller matrix results in faster computation time. Therefore, R can be set to 1 if the network is dense; otherwise it is set to 2. In the ranging-based scenarios, a one-hop neighborhood distance matrix of at least 12×12 produced good results. In the range-free cases, the distance matrix is considerably more inaccurate using the hop count approximation. Thus a larger distance matrix assists in mapping to the node position coordinates using CCA. We set $R = 1$ when the one-hop neighborhood has a very large size (above 30), otherwise $R = 2$ is chosen. The steps of CCA-MAP are thus as follows:

1. For each node x, neighbors within R hops are included in building its local map. Compute the shortest distance matrix of the local map and take it as the approximate distance matrix LD.
2. Each node applies the CCA algorithm using the local distance matrix LD as both the input data set and the distance matrix of the input data set. This generates the relative coordinates for each node in the local map of node x of its R hop neighborhood.
3. Merge local maps.
4. Given sufficient anchor nodes (≥ 3 for 2-dimensional space and ≥ 4 for 3-dimensional), transform the merged map to an absolute map based on the absolute positions of anchors.

In step 3, a randomly selected node's local map is used as the starting point. Each time, the neighbor node whose local map shares the most nodes with the current map is selected to merge its local map into the current map. Two maps are merged/patched using the coordinates of their common nodes. To merge a new local map B into the current map A, a linear transformation (translation, reflection, orthogonal rotation, and scaling) is determined to ensure that the coordinates of the common nodes in map B after transformation best conform with those in current map A. This operation, called the Procrustes method, is the same whether we are merging maps or translating the relative coordinates into global coordinates, based on the known anchor locations.

Briefly, the Procrustes method [46] works as follows: Using the known coordinates of the anchor nodes, a linear transformation of translation, reflection, orthogonal rotation, and scaling is determined for the anchor nodes from the calculated to known coordinates. The transformation is chosen by minimizing the sum of squared errors for the resulting coordinates. Specifically, given actual anchor coordinates, Y, Procrustes gives a transformation as shown in Equation (1) that minimizes the difference between the actual coordinates and the calculated coordinates, Z. In the equation, b is a scalar component, T is the rotation/reflection component, and c is the translation component.

$$Z = b * Y * T + c \tag{1}$$

The transformation is then applied to all nodes in the network by simply replacing Y in Equation (1) with the local coordinates for nodes to be localized in the given absolute coordinate system (*i.e.*, the one used to determine the anchor node locations).

Computing of local maps can be distributed to each local node, or can be carried out at more powerful gateway nodes of each cluster, should the sensor network have a hierarchical structure to relieve the resource-constrained sensor nodes from any of the computing and communication demands imposed by localization. The local maps can be merged in parallel in different parts of the network by selected nodes. The computations can also be done sequentially in a centralized fashion on a central computer. There is no need for anchor nodes in patching the maps. When at least three anchor nodes are found in the patched map of a subnetwork, an absolute map of the subnetwork can be computed using the coordinates of the anchor nodes to obtain the absolute coordinate values of all the nodes in the map of the subnetwork.

In the remainder of this paper, we focus on the last step in CCA-MAP: once a global map with relative coordinates has been determined, the Procrustes method turns these relative coordinates into absolute coordinates, based on the anchor nodes. We are interested in determining rules for placing anchor nodes such that the result of this step results in the best possible localization performance.

5. Best Anchor Node Placements

Before searching for the best anchor node placement we must first define what *best* means, in terms of localization error. Localization error is the distance between the actual and calculated position of each node, measured as a factor of radio radius (or range). This coincidentally also implies that we are exploring this problem in the context of global localization—how well do the calculated global coordinates match the actual global coordinates of a node. While every network has its own application requirements, and thus there are many options for what statistics to examine for accessing the quality of locations, this paper looks at the mean localization error across all nodes in the network. Alternatives could include looking at the best 50% or 80% of nodes, based on how many nodes could be removed from the network for the purpose of the resulting sensed data.

5.1. Anchor Node Placement Metrics

As discussed above, we limit ourselves to three anchor nodes. These three anchors span a triangle in the network. In order to determine the best anchor node placement, we examine three categories of metrics: the error of the anchor node localization itself, network area coverage, and characteristics of the triangle formed by the anchor nodes. The data in the following graphs is taken from a random selection of 1,000 anchor sets of three nodes each. The node placement is random within a 20 by 20 unit square area, and all nodes have a radio range of 2.5 units, providing a completely connected network. All networks, unless stated differently, have 400 nodes, and the radio model is the disk model (*i.e.*, every node within radio range can receive messages sent by a transmitter). To localize nodes, CCA-MAP uses the range-free localization approach. In all graphs, distances are expressed relative to our chosen geographic unit.

5.1.1. Anchor Node Error

The first category is to choose nodes as anchors that we expect to have high location accuracy. Obviously, this information is only available after running the localization algorithm. Because we are trying to provide an a priori design technique for network planners, the information chosen to reach this goal must be something the network planner can determine before any localization has been performed. For this purpose we choose the number of one-hop neighbors. On average, increased network density results in higher localization accuracy [1,2]. Since most localization algorithms depend heavily on network connectivity, the hypothesis is that nodes with more neighbors will have lower localization errors. The idea is that the lower error in the anchors themselves should then translate into a more accurate transformation of the entire network.

To test this idea, we used the sum of the unique one-hop neighbors for all anchor nodes and the mean error of the anchor nodes themselves as metrics. Unfortunately, neither of these metrics provide better localization accuracy, and therefore we did not explore this hypothesis further.

5.1.2. Network Area Coverage

As suggested in many of the related studies in Section 3, an optimal anchor placement is to have the anchors as far apart as possible, around the edges of the network. The rationale behind this hypothesis is that if a wider area of the network is covered by the anchor nodes, then the resulting calculated transformation will take into account more network variations. To measure this, we take the sum of the distance between each pair of anchor nodes.

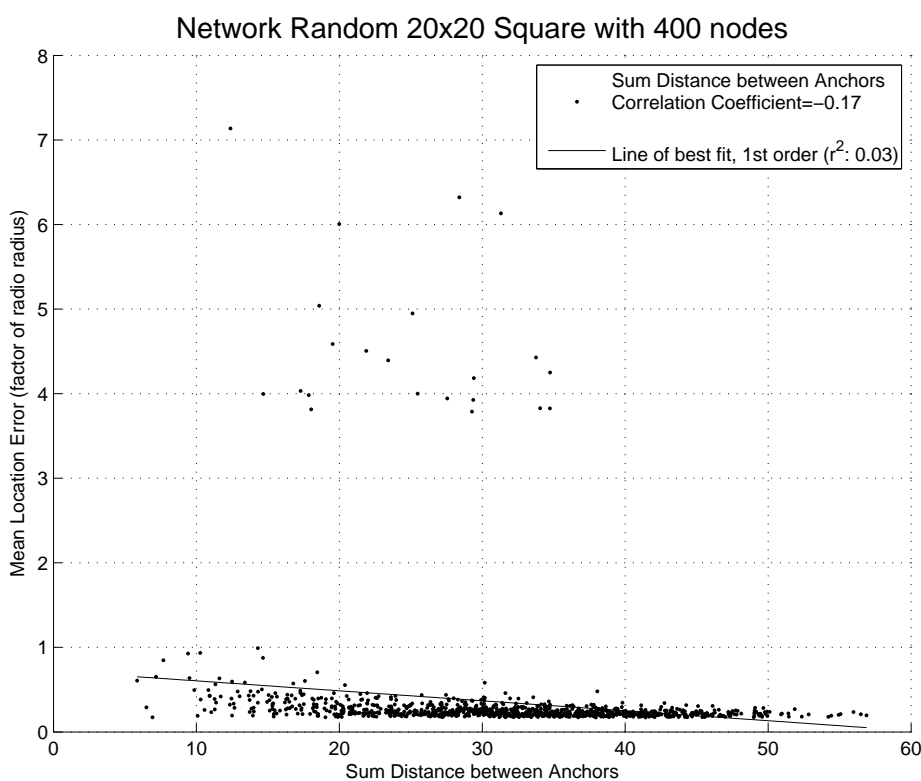**Figure 2.** Sum of distance between anchors *vs.* localization error.

Figure 2 shows the sum of distances for each anchor set versus mean localization error for that anchor set. The plot shows that again there is low correlation between the distance between anchors and the location calculation. Also, it is clear that a small subset of the anchor placements lead to far greater error than the norm. We explore the cause of these outliers in Section 6. The outliers are spread relatively evenly regardless of distance between anchor nodes, and therefore this is not a good indicator of an outlier. However, when the outliers are excluded, a moderate level of correlation is seen: as the distance increases between the anchor nodes, the probability of getting a high localization error decreases.

5.1.3. Anchor Node Triangle

Based on the speculative statements by Shang in [1], we propose and examine an additional metric attempting to measure collinearity of the anchor nodes: the height of the anchor node triangle. More specifically, we calculate the height in each direction of the triangle, and use the shortest value.

Figure 3 plots the minimum height of the triangle formed by the three anchor nodes versus mean localization error. The first observation is that the outlier cases are more tightly correlated to triangle height than any of the other metrics. The outlier points all have comparatively small minimum triangle heights. This is explored in more detail in Section 6.2.

Figure 3. Minimum height of anchor triangle *vs.* localization error.
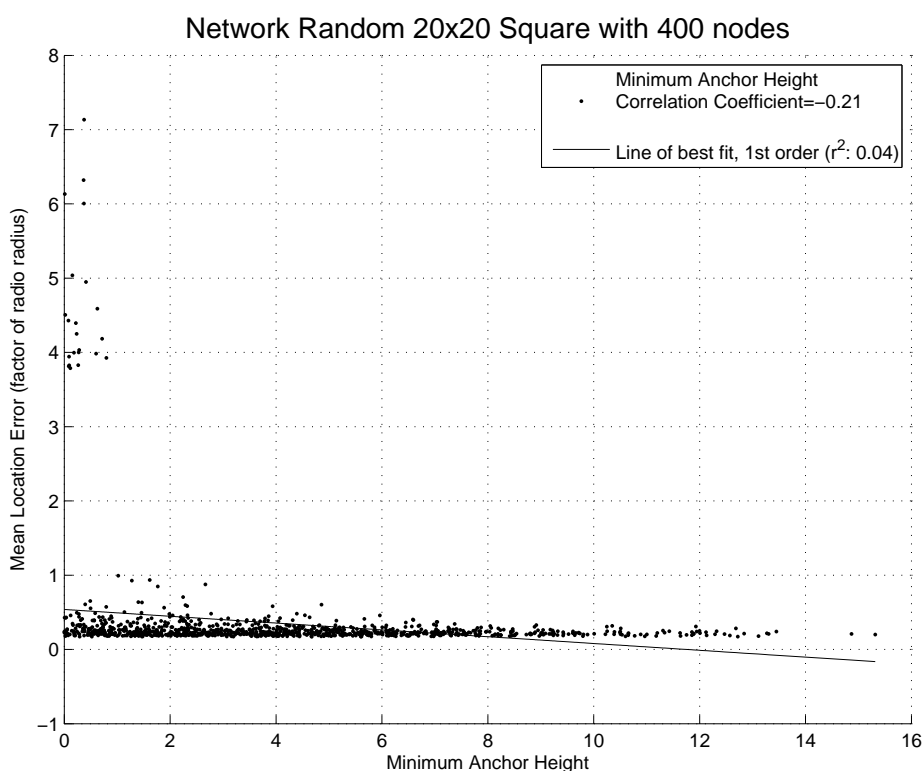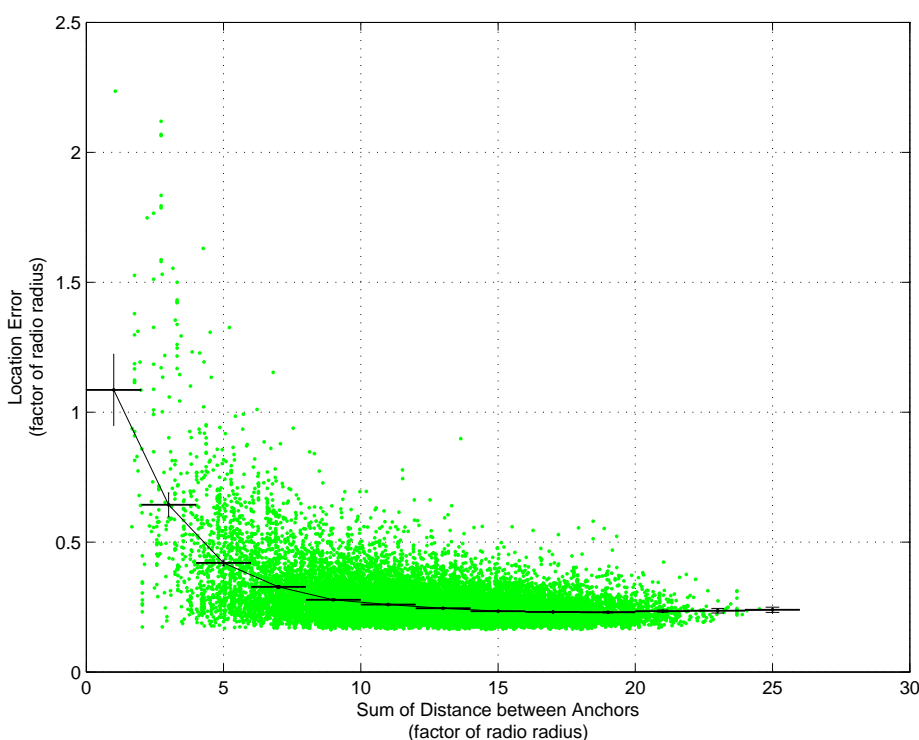

5.2. *Best Anchor Node Placement*

Analyzing the metrics discussed above, the best indicator of a good anchor placement is the sum of the distance between anchor nodes, once the outliers are excluded. This metric is best because it has the best fit (lowest r^2) to the mean localization error. Practically speaking, this means that network designers

can choose between proposed anchor placements and use the measured distance between them to provide guidance about which placement will result in the least amount of error.

Figure 4 uses data from 20 networks, each occupying an area of 20×20 units with a radio range of 2.5 units. All nodes are randomly placed in each network. 5,000 anchor sets of three nodes each are chosen from the randomly placed nodes. The outliers are then removed from the data and the sum of distance between the anchors is plotted against the mean localization error. The data is grouped into intervals 2 units wide to generate a histogram of the data. A horizontal bar through the mean of each interval displays the width of that histogram interval. Further, a 95% confidence interval of the mean value is shown as a vertical bar. This plot shows that, as the sum of distances between the anchors approaches ten times the radio range, the differences in localization error are statistically insignificant. Therefore, network designers can choose anchor node locations at their convenience, as long as they meet this criteria.

Figure 4. Histogram: Sum of distance between anchors *vs.* localization error for 20 random square networks, excluding outliers.

6. Avoiding the Worst Anchor Node Placements

As seen in the previous section, some anchor node placements are significantly worse than the average case. In this section, we explore in more detail the cause of these outliers and, more importantly, whether this condition is detectable a priori and hence preventable in the real world.

6.1. Effects of Procrustes Analysis

Figure 1(b) shows a network map of an example outlier anchor node placement for a random network. A line is drawn between the real location and the calculated location for each individual node. The length of this line is the localization error. For purposes of comparison, Figure 1(a) is a randomly chosen non-outlier case, where the node difference lines are short, representing low localization error. However, in the outlier case, the node difference lines criss-cross the network along a clear, single angle. This visual representation suggests the reflection component of the final linear transformation is to blame for the extremely poor, outlier results.

Unfortunately, simply disabling the reflection component of the Procrustes transformation algorithm does not solve the problem. The output of the Procrustes algorithm is a linear transformation, which includes a rotation or reflection matrix. We examined the characteristics of the generated transformation functions to look for patterns. For a given network, the transformation for most of the anchor sets has almost the same angle of either rotation or reflection, while the outliers have the opposite property with a wide variance in the angle. However, in each network, it is not consistently either rotation or reflection that leads to poor localization results. Therefore, relying on whether the transformation contains a reflection or rotation component is not a sufficient indicator for a network designer knowing that an outlier case has been detected and that the localization results are essentially useless. Also, neither the scaling or translation component of the linear transformation indicate the extremely poor localization performance.

6.2. Outlier Indicators

As discussed above, unfortunately there is no direct indicator that a particular anchor set generates a transformation with an incorrect angle. Therefore, network designers must avoid all anchor sets that could potentially generate such an angle. As demonstrated in Section 5.1.3, the height of the triangle formed by the anchor nodes is a good indicator of the possibility of a poor transformation angle. This is further explored in Figure 5. This time, the x-axis is on a log scale to better show the outlier cases. Also, the height has been normalized to a fraction of the transmission radius. Further, 95% confidence intervals are shown for the mean localization error of each 0.1r increment. A solid horizontal line is shown throw each mean value, indicating the width of the increment, since it can be difficult to visualize on a log scale. The horizontal dashed line indicates the cutoff for what are considered outliers, based on the large gap in data points. The vertical dashed line indicates the first interval in which there are no outliers. As the minimum height increases, the mean localization error decreases, as does the confidence interval of that mean.

Based on the minimum height statistics, we can now give network designers a metric of how collinear is too collinear for a set of anchor nodes. The raw statistics indicate that if the triangle formed by the anchor nodes has a minimum height greater than 0.6 times the radio range, then there is very low probability of getting an extreme outlier case. However, also in the data shown, there are some near-outlier cases beyond this point. While these cases are not of the extreme nature discussed, it may be worthwhile to provide a margin of error here to avoid these cases as well. Therefore, we assert that

the triangle formed by the anchor nodes should have a minimum height equal to the radio range of the network.

Figure 5. Histogram: Minimum anchor height *vs.* localization error, grouped in intervals of 0.1r.
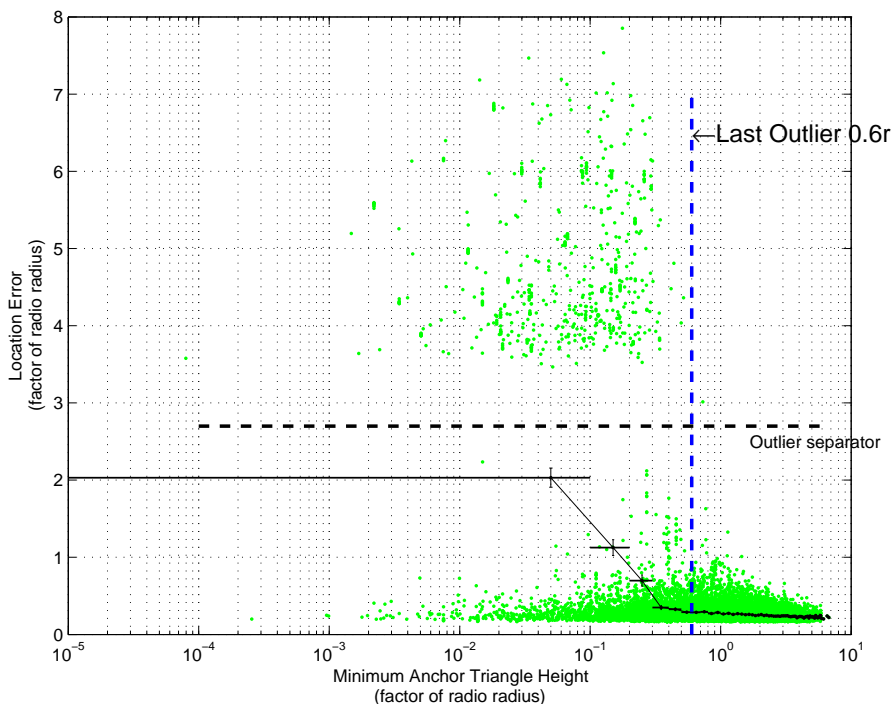
7. Applicability to Varying Network Topologies

So far, we have examined anchor node placement for an isotropic square network, with randomly placed nodes. However, in the real world, networks are not always so simple. There are often regions in the network where it is not possible to put nodes, due to physical barriers like a lake, buildings or access to property. In this section, we look at how the results thus far apply to C-shaped and long, narrow pipeline network topologies.

7.1. C-Shape Network Topology

A C-shape network consists of a relatively square region with an empty area on one side, as shown in Figure 6. In terms of anchor placement requirements, we do not expect any difference between the recommendations presented for square networks.

To show that there is no difference between C-shape and square network topologies, 10 random C-shape networks with 5000 anchor sets each were simulated in the same manner as the square networks. Figure 7(a) shows the localization error relative to the sum of the distances between the anchors, as in Section 5.2. As expected, the mean localization error flattens out as the sum of distances between anchor nodes reaches about 10 times the radio range. There is a slight increase in the floor of the mean localization error as compared with the square network, but this has to do with the performance of the CCA-MAP algorithm in the presence of the empty region of nodes in the network. The increase in

mean localization error at the end of the plot, and specifically the increase in the confidence interval, is caused by the small sample size in the random selection of anchor set with a very high distance between nodes.

Figure 6. Example of C-shape network topology.
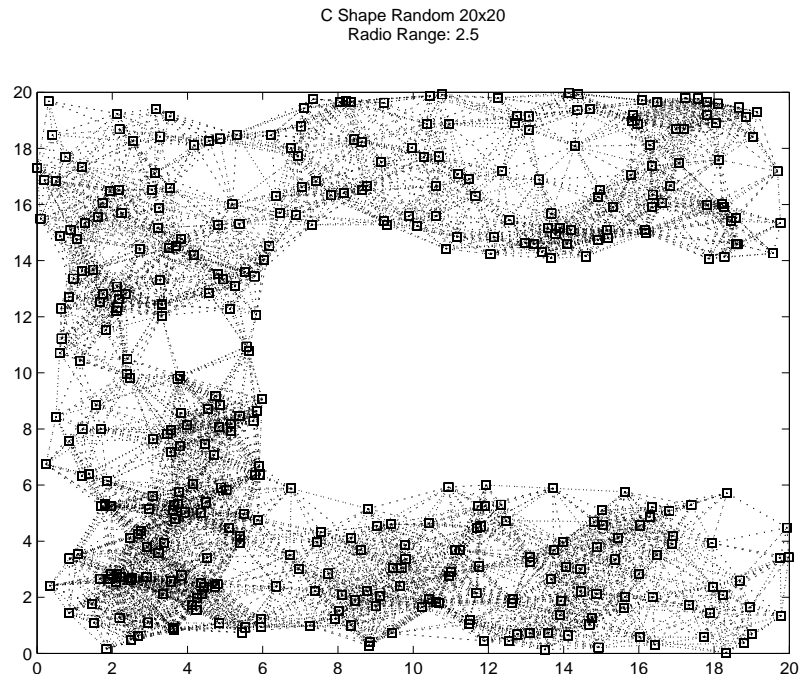
Figure 7. Anchor metrics for 10 random C-shape 20-by-20 unit networks with 400 nodes, 5,000 anchor sets each, and a radio range of 2.5 units. (**a**) Sum of distance between anchors *vs.* mean localization error, excluding outliers; (**b**) Minimum height of anchor triangle *vs.* mean localization error.
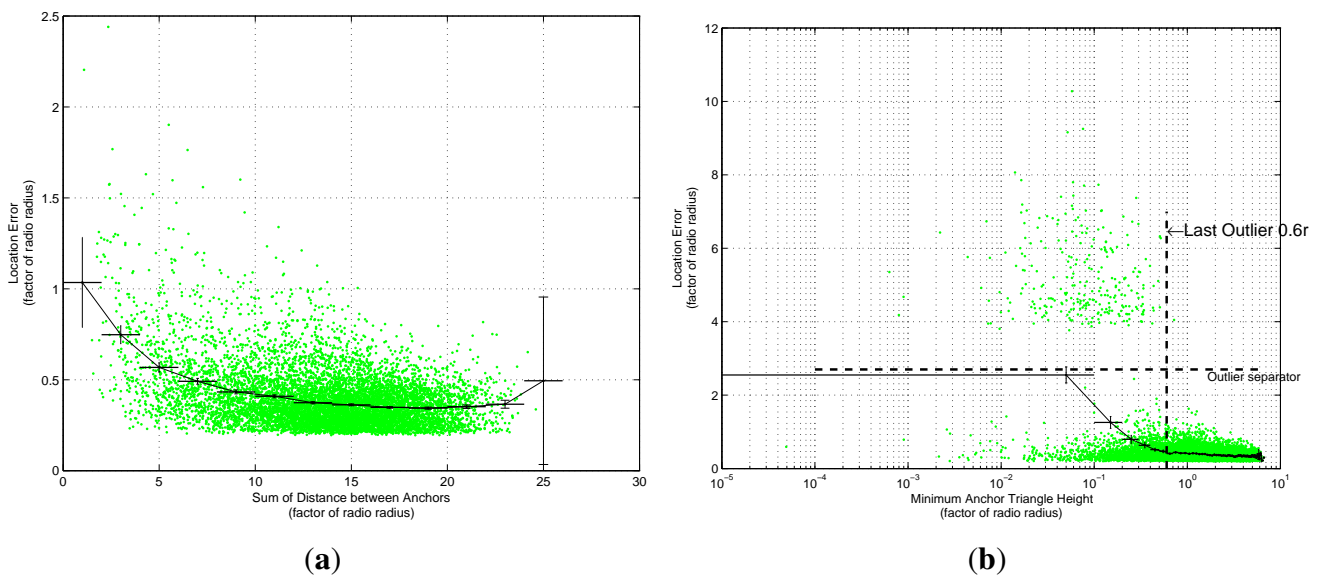
Similarly, the same criteria for eliminating the outlier localization results also holds for the C-shape topology. Figure 7(b) shows that as long as the minimum height of the triangle formed by the anchors nodes is greater than the radio range, the outlier case can be avoided.

7.2. Pipeline Network Topology

In some applications, the network region has very little depth to it, such as when monitoring a gas pipeline or along a highway or railroad line. The extreme case is where there is a single node placed along a straight axis. This is the worst possible scenario, as it is most likely to cause the outlier condition for localization. In that case, it may be worthwhile to explore the possibility of using other localization techniques, such as the use of GPS at each node, or recording the location as the nodes are placed.

Figure 8. Sum of distance between anchors *vs.* mean localization error for random pipeline 25-by-1,2,3 and 4 unit networks with 200 nodes, 1,000 anchor sets each, and a radio range of 2.0 units.
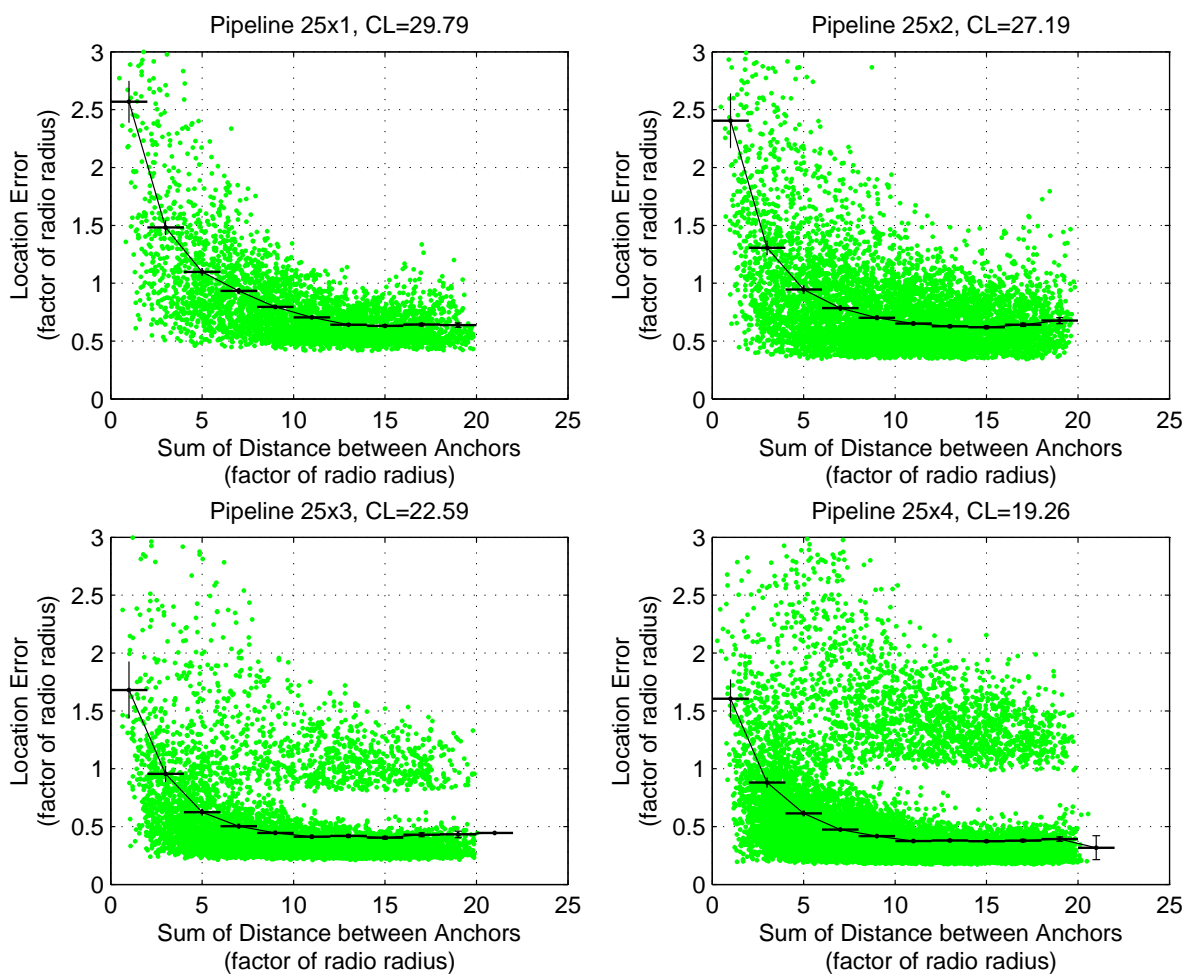
However, when there is a bit of depth to the network, there is still the possibility of achieving good localization results through proper anchor node placement. To demonstrate the importance of having an anchor node triangle height of at least the radio range, Figures 8 and 9 show mean localization error for four different networks, with increasing network depth, but the same length. As before, we attempt

to distinguish the outlier case with a plot of the sum of distance between anchors. Figure 8 shows that as the network depth increases, the floor of possible localization errors drops. In particular, when the network depth is only 1, the floor is about 0.5r, whereas when the depth is 4 units, or twice the radio range, the floor drops to roughly 0.25r. Also, as the sum of the distance increases and the network depth is greater than the radio range, the outlier cases become apparent. This is somewhat counterintuitive: in flatter networks, each node has, on average, more neighbors (captured by CL or Connectivity Level in the figure headings). For localization algorithms such as CCA-MAP or MDS-MAP, that should, all else being equal, result in better localization performance.

Figure 9. Minimum height of anchor triangle *vs.* mean localization error for random pipeline 25-by-1,2,3 and 4 unit networks with 200 nodes, 1,000 anchor sets each, and a radio range of 2.0 units.
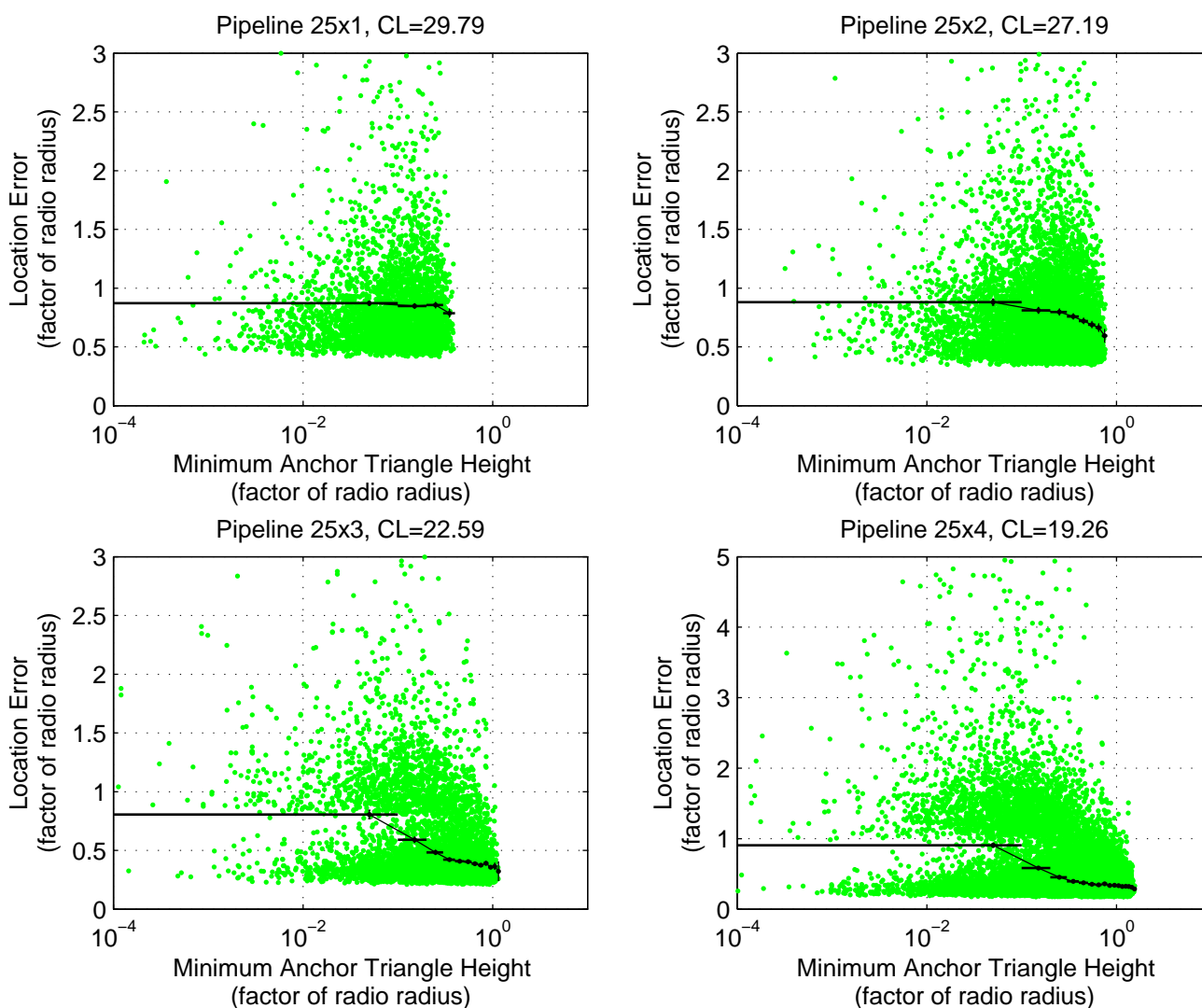
The plots of minimum triangle height in Figure 9 show that for networks with a depth less than the radio range, the separation of good and outlier cases is impossible. Specifically, when the network depth is one, half the radio range, the mean localization error is basically flat across all triangle heights possible. As it becomes possible to have anchor sets forming a triangle with height more than the radio

range, localization performance drops to below 0.5r when the triangle height is also greater than the radio range. However, the minimum triangle height no longer appears to be a good indicator of outlier cases.

In the end, even though the results are better when the depth of the pipeline is more than the radio range, it is impossible to reliably avoid the outlier condition. Therefore, for a pipeline topology, we recommend to use a different class of localization algorithms that does not rely on transforming local coordinates into global coordinates with anchor nodes. In particular, localization algorithms that exploit the specific constraints of a pipeline topology may perform well here. Note that CCA-MAP, like most other proposed localization algorithm, is neither dependent on nor exploiting specific topological properties. If no other algorithms are possible or desirable, we recommend making the network as deep as possible and to spread the anchors as far apart along the length of the pipeline as possible.

8. Conclusions

This study provides two key guidelines for network designers and users of wireless sensor networks when choosing anchor node positions or assessing the quality of localization results. Namely, make sure that the sum of the distance between anchor nodes is at least ten times the radio range and that the minimum height of the triangle formed by the anchor nodes is at least equal to the radio range. Further, the larger these two metrics are, the lower the mean localization error of the network will be, on average, and the lower the probability of using an anchor set that will cause extremely poor localization performance. Informally, this means do not put the anchor nodes in a straight line or close to each other. We have further shown that these criteria apply to network topologies where the overall network area is two-dimensional, but fails when the network topology approaches one-dimension, meaning it is extremely narrow compared with the radio range, for example when monitoring pipelines or roads.

While the simulations in this study use the CCA-MAP algorithm, the results apply to any protocol in which a local coordinate system is transformed into a global coordinate system using a set of anchor nodes with Procrustes analysis. Further, the scope of the study extends beyond wireless sensor network protocols and can be applied to any transformation problem where a small subset of points is used as the basis to transform a set of points.

As reported in [47,48], the Procrustes method suffers from numerical instabilities and errors. In fact, some of the outliers in our results may be explained by this, so further studies on the numerical stability of the method and its impact on the localization performance would be warranted.

While we have provided some recommendations for anchor placement, there are a few key areas where future study could enhance the results. First, an examination of the other factors affecting localization performance would be beneficial, specifically, an analysis of the impact of the network connectivity levels. A raw examination of other as of yet undiscovered factors could also uncover ways to further improve localization accuracy. Second, expanding these results to three dimensions may benefit some network designers. For example, a sensor network may be deployed through a high-rise building or on a bridge. While we expect the results to be similar, it is worthwhile to confirm this. Outside the scope of sensor networks and the physical world, these results could also apply to any dimension of data. This would effectively be an exhaustive study of Procrustes analysis where a set of data in one coordinate system is transformed into another coordinate system based on a subset of known points.

A final area of future research could be to explore other methodologies besides Procrustes analysis to provide a transformation between local and global coordinates. This may be especially beneficial if applied to pipeline topologies.

References

1. Shang, Y.; Ruml, W.; Zhang, Y.; Fromherz, M. Localization from connectivity in sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2004**, *15*, 961–974.

2. Li, L.; Kunz, T. Cooperative node localization using non-linear data projection. *ACM Trans. Sens. Netw.* **2009**, *5*, 1–26.

3. He, T.; Huang, C.; Blum, B.M.; Stankovic, J.A.; Abdelzaher, T. Range-Free Localization Schemes for Large Scale Sensor Networks. In *Proceedings of the International Conference on Mobile Computing and Networking (ACM MOBICOM)*, San Diego, CA, USA, September 2003; pp. 81–95.

4. Liu, J.; Zhang, Y.; Zhao, E. Robust Distributed Node Localization with Error Management. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Com puting (ACM MobiHoc)*, Florence, Italy, May 2006.

5. Doherty, L.; Pister, K.; El Ghaoui, L. Convex Position Estimation in Wireless Sensor Networks. In *Proceedings of the INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, Helsinki, Finland, June 2001; Volume 3, pp. 1655–1663.

6. Ash, J.; Moses, R. On Optimal Anchor Node Placement in Sensor Localization by Optimization of Subspace Principal Angles. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008*, Las Vegas, NV, USA, 31 March–4 April 2008; pp. 2289–2292.

7. Karl, H.; Willig, A. *Protocols and Architectures for Wireless Sensor Networks*; John Wiley & Sons, Ltd: Hoboken, NJ, USA, 2005.

8. Akl, R.; Pasupathy, K.; Haidar, M. Anchor Nodes Placement for Effective Passive Localization. In *Proceedings of the 2011 International Conference on Selected Topics in Mobile and Wireless Networking (iCOST)*, Shanghai, China, October 2011; pp. 127–132.

9. Shang, Y.; Ruml, W.; Zhang, Y.; Fromherz, M. Localization from Mere Connectivity. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MobiHoc)*, Annapolis, MD, USA, June 2003; pp. 201–212.

10. Shang, Y.; Ruml, W. Improved MDS-Based Localization. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, Hong Kong, China, March 2004; pp. 2640–2651.

11. Ihler, A.; Fisher, J.I.; Moses, R.; Willsky, A. Nonparametric belief propagation for self-localization of sensor networks. *IEEE J. Sel. Areas Commun.* **2005**, *23*, 809–819.

12. Bulusu, N.; Heidemann, J.; Estrin, D. GPS-less low cost outdoor localization for very small devices. *IEEE Pers. Commun. Mag.* **2000**, *7*, 28–34.

13. Niculescu, D.; Nath, B. DV based positioning in ad hoc networks. *Kluwer J. Telecommun. Syst.* **2003**, *22*, 267–280.

14. Bahl, P.; Padmanabhan, V. RADAR: An In-Building RF-Based User Location and Tracking System. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, Tel Aviv, Israel, March 2000; pp. 775–784.

15. Priyantha, N.B.; Chakraborty, A.; Balakrishnan, H. The Cricket Location-Support System. In *Proceedings of the 6th Annual ACM International Conference on Mobile Computing and Networking (MOBICOM)*, Boston, MA, USA, August 2000; pp. 32–43.

16. Chen, W.; Mei, T.; Meng, Q.; Liang, H.; Liu, Y.; Zhou, Y.; Sun, L. A Localization Algorithm Based on Discrete Imprecision Range Measurement in Wireless Sensor Networks. In *Proceedings of the IEEE International Conference on Information Acquisition*, Shandong, China, August 2006; pp. 644–648.

17. Alippi, C.; Vanini, G. Wireless Sensor Networks and Radio Localization: A Metrological Analysis of MICA2 Received Signal Strength Indicator. In *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks*, Tampa, FL, USA, November 2004; pp. 579–580.

18. Boukerche, A.; Oliveira, H.; Nakamura, E.; Loureiro, A. A Novel Lightweight Algorithm for Time-Space Localization in Wireless Sensor Networks. In *Proceedings of the 10th ACM Symposium on Modeling, Analysis, and Simulation of Wireless and Mobile Systems*, Chania, Greece, October 2007; pp. 336–343.

19. Priyantha, N.B.; Balakrishnan, H.; Demaine, E.; Teller, S. *Anchor-Free Distributed Localization in Sensor Networks*; Technical Report MIT-LCS-TR-892; MIT Lab for Computer Science: Cambridge, MA, USA April 2003.

20. Yu, G.; Yu, F. A Localization Algorithm for Mobile Wireless Sensor Networks. In *Proceedings of the IEEE International Conference on Integration Technology*, Shenzen, China, March 2007; pp. 623–627.

21. Savvides, A.; Han, C.; Strivastava, M. Dynamic Finegrained Localization in Ad-Hoc Networks of Sensors. In *Proceedings of the Proceedings of the 7th Annual International Conference on Mobile Computing and Networking MobiCom'01*, Rome, Italy, July 2001; pp. 166–169.

22. Langendoen, K.; Reijers, N. Distributed localization in wireless sensor networks: A quantitative comparison. *Comput. Netw.* **2003**, *43*, 499–518.

23. Li, L.; Kunz, T. Localization Applying an Efficient Neural Network Mapping. In *Proceedings of the First International Conference on Autonomic Computing and Communication Systems*, Rome, Italy, October 2007; Volume 1.

24. Liu, B.; Brass, P.; Dousse, O.; Nain, P.; Towsley, D. Mobility Improves Coverage of Sensor Networks. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MobiHoc)*, Urbana-Champaign, IL, USA, May 2005; pp. 300–308.

25. Wang, W.; Srinivasan, V.; Chua, K.C. Using mobile relays to prolong the lifetime of wireless sensor networks. In *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking*, Cologne, Germany, 28 August–2 September 2005; pp. 270–283.

26. Alikhani, S.; St-Hilaire, M.; Kunz, T. iCCA-MAP: A New Mobile Node Localization Algorithm. In *Proceedings of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, Marrakech, Morocco, October 2009; pp. 382–387.

27. Alikhani, S.; Kunz, T.; St-Hilaire, M. iCCA-MAP *versus* MCL and Dual MCL: Comparison of Mobile Node Localization Algorithms. *Lect. Notes Comput. Sci.* **2010**, *6288*, 163–176.

28. Chang, T.C.; Wang, K.; Hsieh, Y.L. Enhanced Color-Theory-Based Dynamic Localization in Mobile Wireless Sensor Networks. In *Proceedings of the IEEE Wireless Communications and Networking Conference*, Hong Kong, China, March 2007; pp. 3064–3069.

29. Hu, L.; Evans, D. Localization for Mobile Sensor Networks. In *Proceedings of the International Conference on Mobile Computing and Networking*, Philadelphia, PA, USA, 26 September–1 October 2004; pp. 45–57.

30. Ou, C.H. Range-Free Node Localization for Mobile Wireless Sensor Networks. In *Proceedings of the 3rd International Symposium on Wireless Pervasive Computing*, Santorini, Greece, May 2008; pp. 535–539.

31. Shee, S.H.; Wang, K.; Hsieh, Y.L. Color-Theory-Based Dynamic Localization in Mobile Wireless Sensor Networks. In *Proceedings of the Workshop on Wireless, Ad Hoc, Sensor Networks*, Columbus, OH, USA, June 2005; pp. 73–78.

32. Stevens-Navarro, E.; Vivekanandan, V.; Wong, V. Dual and Mixture Monte Carlo Localization Algorithms for Mobile Wireless Sensor Networks. In *Proceedings of the IEEE Wireless Communications and Networking Conference*, Hong Kong, China, March 2007; pp. 4024–4028.

33. Ssu, K.F.; Ou, C.H.; Jiau, H.C. Localization with mobile anchor points in wireless sensor networks. *IEEE Trans. Veh. Technol.* **2005**, *54*, 1187–1197.

34. Chenji, H.; Stoleru, R. Mobile sensor network localization in harsh environments. *Lect. Notes Comput. Sci.* **2010**, *6131*, 244–257.

35. Crepaldi, R.; Casari, P.; Zanella, A.; Zorzi, M. Testbed Implementation and Refinement of a Range-Based Localization Algorithm for Wireless Sensor Networks. In *Proceedings of the 3rd International Conference on Mobile Technology, Applications & Systems*, Bangkok, Thailand, October 2006; pp. 1–5.

36. Zanca, G.; Zorzi, F.; Zanella, A.; Zorzi, M. Experimental Comparison of RSSI-Based Localization Algorithms for Indoor Wireless Sensor Networks. In *Proceedings of the The REALWSN'08 Workshop on Real-World Wireless Sensor Networks*, Glasgow, Scotland, April 2008; pp. 1–5.

37. Sichitiu, M.; Ramadurai, V. Localization of Wireless Sensor Networks with a Mobile Beacon. In *Proceedings of the IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, Fort Lauderdale, FL, USA, October 2004; pp. 174–183.

38. Bal, M.; Xue, H.; Shen, W.; Ghenniwa2, H. A Testbed for Localization and Tracking in Wireless Sensor Networks. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, San Antonio, CA, USA, October 2009; pp. 3591–3586.

39. Al-Odhari, A.; Alikhani, S.; Dirie, A.; Beki, H.O.; Sirko, L.; St-Hilaire, M.; Kunz, T. Implementation of the CCA-MAP Localization Algorithm on a Wireless Sensor Network Testbed. In *Proceedings of the 7th International Wireless Communications and Mobile Computing Conference (IWCMC 2011)*, Istanbul, Turkey, July 2011; pp. 992–997.

40. Hara, S.; Fukumura, T. Determination of the Placement of Anchor Nodes Satisfying a Required Localization Accuracy. In *Proceedings of IEEE International Symposium on Wireless Communication Systems (ISWCS '08)*, Reykjavik, Iceland, October 2008; pp. 128–132.

41. Savarese, C.; Rabaey, J.M.; Langendoen, K. Robust Positioning Algorithms for Distributed Ad-Hoc Wireless Sensor Networks. In *Proceedings of the ATEC '02: Proceedings of the General Track of the Annual Conference on USENIX Annual Technical Conference*, Monterey, CA, USA, June 2002; USENIX Association: Berkeley, CA, USA, 2002; pp. 317–327.

42. Cheng, K.Y.; Lui, K.S.; Tam, V. HyBloc: Localization in sensor networks with adverse anchor placement. *Sensors* **2009**, *9*, 253–280.

43. Lim, H.; Hou, J. Localization for Anisotropic Sensor Networks. In *Proceedings of the INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, Miami, FL, USA, March 2005; Volume 1, pp. 138–149.

44. Zemek, R.; Takashima, M.; Hara, S.; Yanagihara, K.; Fukui, K.; Fukunaga, S.; Kitayama, K. An Effect of Anchor Nodes Placement on a Target Location Estimation Performance. In *Proceedings of the TENCON 2006. 2006 IEEE Region 10 Conference*, Hong Kong, China, November 2006; pp. 1–4.

45. Demartines, P.; Hrault, J. Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Trans. Neural Netw.* **1997**, *8*, 148–154.

46. The MathWorks, I. Procrustes Analysis. 2010. Available online: http://www.mathworks.com/help/toolbox/stats/procrustes.html (accessed on 5 June 2012).

47. Dorst, L. First order error propagation of the procrustes method for 3D attitude estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 221–229.

48. Larsen, R. Generalized procrustes 2D shape alignment. *J. Math. Imaging Vision* **2008**, *31*, 189–194.