

Performance of Angle-and-Range-Based Localization of Wireless Sensors*

Patryk Mazurkiewicz and Kin K. Leung

Imperial College, Exhibition Road, London, SW7 2AZ, U.K.

E-mail: {patryk.mazurkiewicz06, kkleung}@imperial.ac.uk

Abstract

In wireless sensor networks, the accuracy of localization information deteriorates significantly after relaying multiple hops due to accumulation of errors. Techniques exist to enhance the localization information by employing redundant measurements available in networks with high connectivity or from a large number of reference nodes. However, it is challenging to obtain accurate localization information in deployment environments with: 1) unfavorable radio-propagation characteristics that degrade measurements, or 2) sparse topologies that limit the redundancy. One such example is underground tunnels.

In this paper, we study the performance of localization techniques that are applicable to such sparse networks. Particularly we focus on the COBALT method, which is based on cliques of nodes in the network and uses angle of arrival and range estimates. For different techniques, we show the influence of the connectivity among sensor nodes on the accuracy. Our results show that the COBALT method outperforms other general-purpose localization techniques.

1. Introduction

Localization of nodes in wireless sensor networks (WSNs) has attracted considerable research attention recently. Accurate location estimations of sensors are required in numerous industrial, military and public applications of WSN. Sensor networks used for monitoring purposes in general require accurate positioning information in order to link the sensing data to the place where it is gathered. From a wider perspective, localization in WSN can serve as a basis for solving a number of networking problems, like scheduling of the transmission [1] and geographical routing [2].

WSNs used for monitoring applications have to be designed to meet the specific requirements and characteristics of the application environments. This fact is strongly reflected in the indoor systems where operating constraints can vary significantly from one application scenario to another. The particularly sparse network topologies, like those in tunnels, impose constraints on the possible network topology. Deployed in such long structures, the WSN has a linear topology [3, 4, 5]. This limits the usability of certain localization methods or reduces dramatically their accuracy. In fact, the public literature has few studies about methods dedicated to localization in tunnels and thus the challenges for accurate localization techniques remain to be addressed.

The major issue with localization in sparse topology networks is that nodes tend to have a relatively small number of neighboring nodes. For that reason many localization algorithms (like [6, 7]) not always produce a unique solution. We say that the algorithms do not hold rigidity if the node degree of connectivity is below a certain threshold.

Rigidity of localization algorithm is the ability to solve the localization problem uniquely. Rigidity is an equivalent of determinability of the set of localization equations that result from the localization algorithm. Some algorithms not always hold rigidity conditions, hence are unable to localize all nodes uniquely in certain cases. For example, 3-D trilateration is not rigid if node connectivity is less than 4 [7]. Our algorithm referred to as COBALT (Clique-of-Nodes Based Localization Technique) produces a unique solution for any topology of connected nodes. This is because each node gathers enough information about their neighbors. Nodes measure distance-vectors (i.e. spherical angle and distance) to neighbors and the earth gravity direction.

In this paper we continue work presented in [3]. We focus on a localization method called COBALT which employs measurements of radio-wave direction-of-arrival [8], range and earth gravity direction. Our main contribution in this publication is as follows. Firstly, we propose a solution to 3-D localization problem of wireless sensors in sparse networks like tunnels. Secondly, our method produces a rigid estimation of the network for any topology,

*This work is funded by UK EPSRC Research Grant EP/D076838/1, entitled: "Smart Infrastructure: Wireless Sensor Network System for Condition Assessment and Monitoring of Infrastructure".

i.e. COBALT can localize even the nodes whose connectivity is as little as 1. Thirdly, we propose a heuristic for identifying cliques of nodes in a network.

The remaining part of the paper is organized as follows. Our localization algorithm is described in section 2 preceded by definitions of the associated terminology. Following this we discuss the inherent properties of our algorithm - COBALT in the section 3. Simulation results showing the performance of the proposed algorithm and discussion of the results are presented in section 4. Concluding remarks can be found in section 5.

2. Localization Algorithm

In this section we present the COBALT algorithm. This algorithm comprises of three stages: (1) Distributed identification of overlapping cliques of nodes across the whole network. As a result, some nodes belong to more than one clique. (2) Second stage occurs independently in all cliques, possibly in parallel. Members of a particular clique through message exchange and computations agree on their reference system that is intended to be common for all members of this particular clique. Some nodes therefore are aware of more than one reference system: one in each clique they belong to. (3) Progressively, starting at a particular clique, the reference systems between two cliques are agreed between one another and a common reference system for those two cliques is introduced. This process occurs iteratively thus propagating one common reference system throughout the network. Process continues until there are no more cliques to propagate the reference system to. This iterative process is referred to as *stitching*.

The order of the cliques to participate in the iterative process of agreeing on the common reference system (i.e. in the process of stitching) is studied by [9] and we adopt their approach of establishing the stitching-order in our algorithm.

2.1. Alignment & Identity of Ref. Systems

The two Cartesian frames of reference (non-prim and prim) have the property of being aligned iff any vector \mathbf{a} observed in one frame of reference equals $\mathbf{a}' = \mathbf{a} + \mathbf{d}$ in the other frame of reference, where \mathbf{d} is a constant for those two frames of reference and is a distance vector of the prim-system origin from the non-prim origin. We can say of the corresponding axes of aligned frames of reference that they point towards the same direction.

The two Cartesian frames of reference have the property of being identical iff they are aligned and the distance vector between the origins is $\mathbf{d} = \mathbf{0}$. Observation of any vector \mathbf{a} in one of the frames of reference equals the observation of \mathbf{a} in the other frame of reference.

Aligning is achieved by the method presented in [10] that originates from the absolute orientation method [11], but uses all possible distance constraints between aligned reference systems.

2.2. Maps and Reference Systems

We call “map” (of a node, of a clique or global) a set of position-vectors which are estimates of distances from the origin of the reference system of the “map” to positions of nodes represented at a particular “map”. There are three following contexts in which we refer to maps.

- At the beginning of the localization process, just after a node is activated to work; Nodes use *hardware reference system* (hdw_RS) which is associated with the node’s physical construction. The angle and distance measurements conducted in hdw_RS determine position-vectors to the node’s neighbors at node’s *local map*.
- In a clique which is identified in the network during operation of COBALT; The group of n nodes which has been identified to be a clique will proceed to obtain a *clique reference system* (cli_RS) and a *clique map* which both are common to all n members of the clique. In a particular clique c , each membering node $k = 1..n$ possesses a parameter \mathbf{R}_c^k that is a rotation that rotates vectors from node’s k hdw_RS to cli_RS of c . Each node k also has a parameter \mathbf{C}_c^k which is a position vector of this node in the cli_RS of clique c .
- In order to achieve a network-wide localization, an introduction of *global reference system* (glo_RS) is needed. At the end of the localization process glo_RS is common to all nodes. At the beginning of the localization process one clique is chosen (as discussed in [9, 10]) to be the initial part of the *global map*. Other cliques are added iteratively to the global map until there are cliques that can be added. The order of this iterative clique-adding to the global map is defined in [9, 10]. The computations done in order to add a clique to the global map is referred to as *stitching*, and it is equivalent to introducing identity of cli_RS with regards to glo_RS (as described in section 2.1). After clique c is stitched to the global map each node which is a member of c becomes a member of the global map. After stitching all cliques identified in the network of N nodes, each node $K = 1..N$ has two parameters: \mathcal{R}^K and \mathcal{D}^K which align hdw_RS of K to glo_RS, and transfer the origin of hdw_RS of K to this of glo_RS respectively.

Stitching map a' in hdw_RS to map a in glo_RS is a process which introduces identity of a' with regards to a . Precisely,

this process computes the parameters \mathcal{R} and \mathcal{D} that transform hdw_RS in such a way that it is identical with glo_RS . One crucial remark is that \mathcal{R} is chosen in a way that for the earth gravity direction co-located with vector \mathbf{G} the following holds: $\mathbf{G} = \mathbf{G}\mathcal{R}$. This assumption does not introduce an error of any kind, since the earth gravity in both reference systems is oriented in the correct direction - physically in the same direction.

2.3. Identification of Cliques

Below we introduce a distributed heuristics that finds cliques of nodes of the network in a reasonable time.

Clique in a graph is a completely connected sub-graph. The task of finding a clique in a group of nodes is an NP-hard problem. In order to solve the problem within a reasonable time and distributively we propose a heuristic. The heuristic needs $\mathcal{O}(n \log n)$ average time, $\mathcal{O}(n^2)$ worst-case time, and $\mathcal{O}(n)$ messages for finding a clique in a group of n neighbors. The heuristic uses two connected nodes as initial candidates for a clique. This is correct since all nodes of this group of two nodes are connected with each other. The heuristic then proceeds to iterative selection of one most favorable (in terms of conditions explained below) neighbor of all current candidates for a clique and adds it to the clique. The list of the common neighbors is then refreshed and the selection attempted until there are neighbors in common.

The criterion of the selection of a node for adding it to clique candidates is the number of neighbors this node shares with all current clique candidates. The heuristic in pseudo-code is shown in Listing 1.

Listing 1. Partitioning of the network into overlapping cliques.

```

1  FOR EACH PAIR p OF CONNECTED NODES
2    clique=p
3    inCommon=NEIGHBOURS OF ALL FROM clique
4    SORT(inCommon)
5    WHILE inCommon NON-EMPTY
6      n = inCommon(FIRST_NODE_OF_IT)
7      clique = clique OR n
8      inCommon = inCommon AND NEIGHBOURS(n)
9    END
10 END
11 REMOVE REPLICATED CLIQUES

```

The initial selection for the clique members is two linked nodes. In fact, all connected pairs of the network have to belong to a clique in order for the clique maps to overlap. Note that not only “partitioning of a network into maximum cliques” is needed. It is a partitioning which 1) “ensures overlaps of the clique maps” in order to enable building a

connected tree of clique map stitching and 2) “finds possibly maximal cliques” in order to maximize the use of redundancy of measurements.

In line 3 there is a creation of *inCommon* list of nodes that are neighbors of all nodes from the *clique* list-variable.

The dominating operation of the heuristic is sorting the list of neighbors (line 4), hence the optimal complexity can be as good as this of comparison sort algorithms.

There are two criteria for sorting: the more significant criterion is the quality of the geographical information that can be provided by a node. (This criterion is analogous to the bandwidth in [9].) The less significant criterion is the number of neighbors in common with the initial pair of nodes. The sorting order is descending, hence the nodes most favorable for being proceeded with are stored first in the *inCommon* list (hence the instruction in the line 6).

2.4. Clique Reference System

In this sub-section we focus on the procedure of computing \mathbf{R}_c^k and \mathbf{C}_c^k for each node $k = 1..n$ in the clique c comprising n nodes. Those parameters are computed in order to introduce a reference system common to all nodes k of c . Transformations \mathbf{R}_c^k and \mathbf{C}_c^k transform hdw_RS of k to cli_RS of c .

The cli_RS will have its origin in the centroid of the clique. We choose to use the centroid for the origin because node maps have to be aligned through rotations around the centroid [11], so it is straightforward to retain the position vectors as they are needed by this method used in the localization algorithm.

The measurements are carried out in the hdw_RS of k . The position vector of node i measured by node k is noted as $\bar{\mathbf{X}}_i^k$. The local map of node k is referred to as $\bar{\mathbf{X}}^k$. The centroid of the local map is the average of all points on this map, hence:

$$\bar{\mathbf{C}}^k = -\frac{1}{n} \sum_i \bar{\mathbf{X}}_i^k \quad (1)$$

where the minus sign indicates reverting the direction of the node’s measurements in a way that the origin of the node map is in the centroid, not at the origin of hdw_RS . In order to align the hdw_RS of k to hdw_RS of m one has to find a 3-D rotation \mathbf{R}^k such that it rotates vector $[\bar{\mathbf{X}}_i^k + \bar{\mathbf{C}}^k]$ onto vector $[\bar{\mathbf{X}}_i^m + \bar{\mathbf{C}}^m]$ with earth gravity co-linear with the rotation axis or \mathbf{R}^k . For that purpose a minimum square error with regards to \mathbf{R}^k should be computed:

$$\min \sum_{i=1}^n (\bar{\mathbf{X}}_i^m + \bar{\mathbf{C}}^m - (\bar{\mathbf{X}}_i^k + \bar{\mathbf{C}}^k) \mathbf{R}^k)^2 \quad (2)$$

We call this expression peer-to-peer map alignment expression. Minimization is needed because we are not aware of

an algebraic method for finding \mathbf{R}^k . By aligning systems of all nodes in a clique to a hdw_RS of one arbitrary node the orientation of cli_RS is established. The origin of the cli_RS is in the centroid of all node map centroids, hence:

$$\mathbf{C}^k = \frac{1}{n} \sum_{k=1}^n \mathbf{C}^k \mathbf{R}^k \quad (3)$$

and the observations of the physical measurements of node k can be expressed as:

$$\mathbf{X}^k = (\bar{\mathbf{X}}^k + \bar{\mathbf{C}}^k) \mathbf{R}^k \quad (4)$$

Note, that earth gravity direction which is the same in both reference systems restricts \mathbf{R}^k to a rotation with a fixed rotation axis co-located with the earth-gravity-direction vector. Hence the equation 2 can be solved uniquely for as little as one pair of observations: \mathbf{X}_2^1 and \mathbf{X}_1^2 in a clique of two nodes. In other words, rigidity holds even for $i = 1$ (one pair of observations, i.e. two nodes) in equation 2. Without the earth-gravity-direction constraint the expression needs at least three physical nodes in order to hold rigidity (three nodes is a setup where at least two pairs of observations exist).

2.5. Stitching Cliques

In this sub-section we explain the process of stitching clique c to the global map. In other words we explain the procedure of computing the transformations of cli_RS to glo_RS.

For stitching of a clique to the global map to be possible, there have to be at least one node s in the clique which is a member of the global map, i.e. is stitched ($s = 1..t$ and $t \in (1..n)$, where t is the number of nodes already stitched). All nodes s have the parameters transforming their hdw_RS to glo_RS (because they are stitched) as well as the parameters transforming their hdw_RS to cli_RS of c (because they are members of c). From this fact the transform of cli_RS of c to glo_RS with regards to s can be written as:

$$\mathcal{R}^{cs} = (\mathbf{R}_c^s)^T \mathcal{R}^S \quad (5)$$

where $S \in \{1..N\}$ is a node in the network of N nodes we are referring to as s in clique c , hence its global transforms are \mathcal{R}^S and \mathcal{D}^S , and the transpose inverts the transformation.

Of course for $t > 1$ there will be $t > 1$ rotations \mathcal{R}^{cs} , hence requiring “averaging” the rotations which is not straightforward; it consists of finding a rotation that rotates any vector \mathbf{X} onto the vector being: $\frac{1}{t} \sum_{s=1}^t (\mathbf{X} \mathcal{R}^{cs})$.

The global position vector of node M is the average of position vectors of M which are estimated by all possible nodes S and it is therefore \mathcal{D}^M such that

$$\mathcal{D}^M = \frac{1}{t} \sum_{s=1}^t (\mathcal{D}^S + (\mathbf{C}^m - \mathbf{C}^s) \mathcal{R}^{cs}) \quad (6)$$

where M , ($M \in \{1..N\}$) is a node which we are referring to as a member m of the clique c .

3. Discussion on COBALT properties

There are three inherent properties of COBALT which we would like to discuss. Firstly, the approach of COBALT is motivated by the problem existing in map stitching algorithms which stitch maps one-by-one. We call this problem a “problem of cascade of averaging” which is discussed below. Secondly, COBALT has a linearly increasing computational cost with the increase of the degree of node connectivity compared to map stitching. However, we find this to be a very low price for the accuracy gain that COBALT offers. Thirdly, COBALT is rigid for any topology, i.e. it always produces a unique estimation of the topology.

The problem of “cascades of averaging” has been observed in the map stitching algorithms which stitch maps in a series, one-by-one, i.e. not in parallel. When map \mathbf{X}^k of node k is being stitched to the global map G it does not benefit from the redundancy offered by the not-yet-stitched nodes o and their maps \mathbf{X}^o . This leads to a loss of the potentially useful redundancy. Many algorithms do not fix that problem at all, including [9]. One trivial solution is to take the omitted redundant measurements from \mathbf{X}^o into account later in order to refine position of K by the means of averaging. The refinement would be: $\mathbf{X}^k = \text{avg}(\mathbf{X}^o, \mathbf{X}^k)$. However, iterative application of this method to a position vector \mathbf{X}^k leads to degradation of the contribution of the initial position vector: $\mathbf{X}^k = \text{avg}(\mathbf{X}^{o1}, \text{avg}(\mathbf{X}^{o2}, \dots, \mathbf{X}^k))$. This phenomena reduces even further the contribution of the earliest-used position vector in case of a long cascade of averages, whose length is unknown beforehand. COBALT uses several local maps available at a moment in order to compute the clique map, hence produces enhanced maps resilient for the problem at this step.

The increasing computational complexity with increasing connectivity comes from the fact that COBALT builds maps for each pair of connected nodes, hence builds up to $NC/2$ maps (for network of N nodes and degree of connectivity C), hence up to $\frac{C}{2}$ times more than map stitching algorithms. We find this relative increase of the cost is small compared to the accuracy gain.

Rigidity is an ability of localization algorithm to solve the localization problem uniquely. While many wireless sensor localization algorithms available in the public literature require certain degree of connectivity for being rigid, COBALT is rigid in any topology and can localize all nodes uniquely regardless on the degree of connectivity. This is

achieved by rich spatial measurements taken as an input to the algorithm. COBALT uses angle-of-arrival, range and earth gravity direction. This enables peer-to-peer localization, as explained in 2.4, equation 2.

4. Simulation Results

In order to investigate the performance of COBALT we simulate its behavior in three 3-D topologies and compare the results with simulations of two other algorithms on the same set of data. The topologies we used are: C-shaped, highly clustered C-shaped and uniform. The projections on the plane of the first two topologies are illustrated in figure 1.

We regulate the degree of connectivity in the networks by varying the number of nodes deployed in the network area from 150 to 400. Initially in each simulation the nodes have their hardware reference system randomly rotated and origin of this system set to be in the node’s physical position. One node in the network is oriented and positioned correctly in the global reference system. It is in the left bottom corner of the maps in figure 1 (a) and (b).

Each node in our simulations has the same communication range r which is 90 meters in our simulations. The two nodes spaced d from each other succeed to communicate with the probability $p = 1 - \frac{d^2}{r^2}$, for $d \leq r$. Even though this assumption is not a reliable simulation of the communication channel, it is a reasonable initial approach [9].

We assume the uniform distribution of the measurement errors, as in [9, 12]. We used two ranges of measurement errors: ranging error $\pm 7.5\%$ of the communication range and angular error ± 7.5 degrees. The error range which is shown as second in the figure 2 is: $\pm 25\%$ for ranging and ± 15 degrees for angles. The earth gravity direction can be determined in reality using accelerometers with the accuracy as good as a fraction of a degree, hence its error can be neglected in the deliberations where the scale of the error of radio-wave measurements is as high as mentioned above.

The other two algorithms we use for comparison are: map stitching, which has a stitching-order decision-making mechanism the same as COBALT does, and flooding algorithm, which does not employ any inter-node co-operation technique for eradicating the error.

Map stitching algorithm is originally a 2-D localization algorithm [10, 9] that uses only distance estimations for creating node maps by trilateration. We extend the original algorithm in two ways: firstly, we create node maps using angle-of-arrival and range measurements (see section 2.2), secondly we extend the map-stitching procedure to 3-D (see section 2.5), hence extending the algorithm to 3-D. To summarize the description of Map Stitching algorithm, the major difference between Map Stitching and COBALT

is that Map Stitching performs stitching of *node maps* while COBALT performs stitching on *clique maps*.

Flooding algorithm is a trivial solution, which does not use any redundant measurements and any error correction mechanism. The reference system is propagated through the network from node to node, peer-to-peer, until all nodes are localized. Passing the reference system occurs along the shortest path to the reference node. Information about the neighboring nodes is not used in the process of propagation of the reference system.

Figure 2 presents the numerical results. The horizontal axis of each graph shows the average degree of connectivity of the instance of network simulated. The vertical axes show the average localization error normalized over the communication range. The communication range in our simulations was set to 90 meters.

The average localization error for COBALT oscillates around $1/3$ of the communication range in the most cases, only slightly improving for higher degrees of connectivity, as can be seen in figure 2 (b) for dash-line performance plot. The accuracy of the other algorithms depends on the degree of connectivity to a higher extent.

It can be seen from most of the results illustrated in figure 2 that the performance of map stitching remains relatively constant compared to the performance of the flooding algorithm. Invariant gain of map stitching algorithm may be due to the problem of cascades of averages, which reduces the increase of the accuracy gain despite the increase of the redundancy. In the meanwhile, the performance of flooding improves as the network becomes more dense and the minimum degree of connectivity increases, because the higher connectivity boosts chances for long-haul links (due to how the communication channel is modeled) and consequently reduces the number of hops to the reference node. Less hops means better localization accuracy.

The adverse effect of the sparse topology (highly clustered C-shape) on general-purpose localization algorithms (like map stitching and flooding) can be seen from the figure 2 (c). The accuracy of these algorithms in the experiment on clustered topology depends highly on the accuracy of the input (measurements) data, unlike in the other experiments, where the algorithms performed relatively similarly for both error ranges.

5. Conclusions

In this paper we present a distributed and scalable algorithm referred to as COBALT which solves the problem of localization for 3-D networks of wireless sensors outperforming other methods suitable for solving this problem. Our focus in this work is on enhancing the local maps prior to stitching them to the global map. Enhancing the local maps results in achieving much better accuracy, as can be

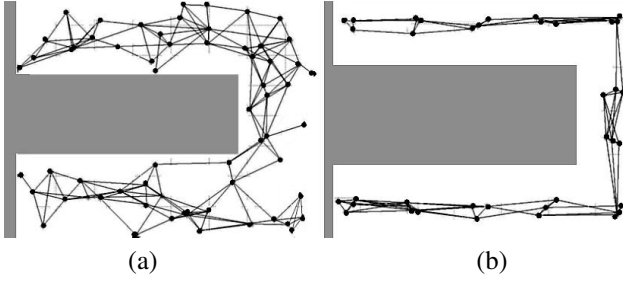


Figure 1. Projections on the plane of example 3-D network topologies that we used: (a) C-shape, (b) C-shape highly clustered.

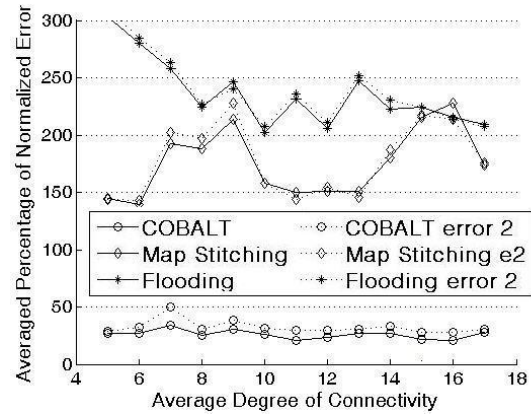
seen in section 4 and figure 2. The algorithm takes the following measurements as input: distance, radio wave angle of arrival and earth gravity direction. The method used for eradication of the measurement error is minimum square error, as in equation 2. The nodes which are being processed in order to eradicate the error belong to groups of fully connected nodes, i.e. cliques.

We have proposed a fast heuristic that finds cliques in the connectivity graph of the sensor network. The heuristic is applicable to be implemented in a distributed way and performed by sensor nodes.

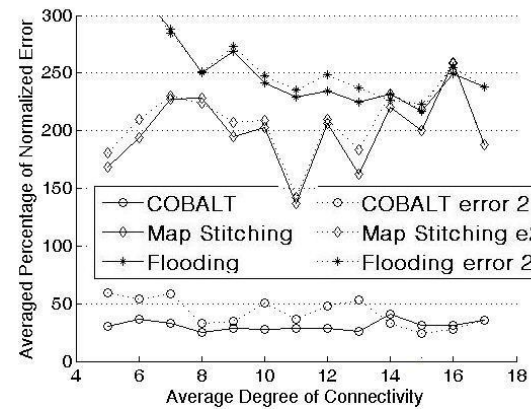
We have two concluding remarks about COBALT. Firstly, the approach of stitching a few maps collectively has the accuracy advantage over stitching of a single map at a time. This is due to avoiding the problem of cascade of averages. This problem appears when some initial values are averaged with each other and are later averaged with yet another value. The contribution of the initial values into the overall average is thus not proportional and may be further disrupted if the cascade has many steps. COBALT mitigates this problem by using cliques for computing stitching parameters. Secondly, COBALT has no rigidity conditions, i.e. the network is always rigid, thus allowing for localization of all nodes connected. This property was achieved by using rich spatial measurements including: range, angle of arrival as well as earth gravity direction. The property of rigidity is crucial for localization in sparse networks like those deployed on tunnels.

6. Acknowledgments

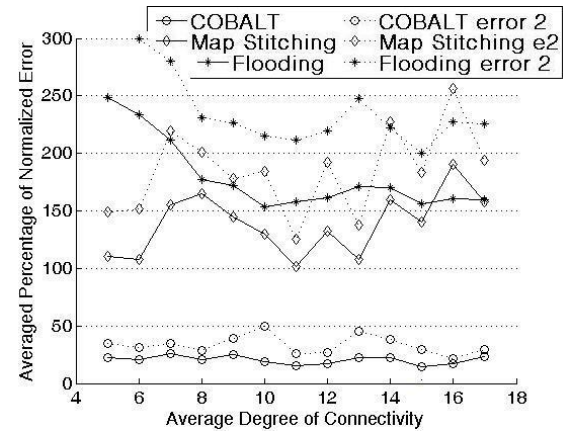
Authors would like to thank Dr. Abdelmalik Bachir for reading the draft of this paper and for his valuable comments on the presentation of this work.



(a) uniform topology



(b) C-shape topology



(c) C-shape topology, highly clustered

Figure 2. Localization error with varying degree of connectivity for different topologies. Two measurement error setting used are: $[\pm 7.5\%, \pm 7.5^\circ]$ and error 2: $[\pm 25\%, \pm 15^\circ]$ ([normalized to the communication range, and the angular error]).

References

- [1] S. M. Hur, S. Mao, Y. T. Hou, K. Nam, and J. H. Reed, "A location-assisted mac protocol for multi-hop wireless networks," in *IEEE Wireless Communications and Networking Conference 2007*, 11-15 March 2007, pp. 322 – 327.
- [2] Z. Sun, R. Yu, and S. Mei, "Robust power-aware routing in wireless sensor networks with special concern about localization error," in *Asia-Pacific Conference on Communications*, 2006, pp. 1–5.
- [3] P. Mazurkiewicz and K. K. Leung, "Clique-based location estimations for wireless sensors in gps-free environments," in *17th International Conference on Computer Communications and Networks*, U.S.A., 3-7 August 2008.
- [4] K. Soga, N. Graham, K. K. Leung, and et al., "Smart infrastructure: Wireless sensor network system for condition assessment and monitoring of infrastructure," EPSRC Research Grant EP/D076838/1, <http://gow.epsrc.ac.uk/ViewGrant.aspx?GrantRef=EP/D076838/1>, Research Project Proposal, October 2006.
- [5] A. Magnani and K. K. Leung, "Self-organized, scalable gps-free localization of wireless sensors," in *IEEE Wireless Communication and Networking Conference*, Hong Kong, China, 2007.
- [6] D. Niculescu and B. Nath, "Geometry and positioning: Error characteristics of ad hoc positioning systems (aps)," in *ACM MobiHoc*, Roppongi Hills, Tokyo, Japan, 2004, pp. 20–30.
- [7] V. P. Sadaphal and B. Jain, "Localization accuracy and threshold network density for tracking sensor networks," in *IEEE International Conference on Personal Wireless Communications*, 23-25 January 2005, pp. 408 – 412.
- [8] A. Kalis and T. Antonakopoulos, "Direction finding in ieee802.11 wireless networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 5, 2002.
- [9] O.-H. Kwon and H.-J. Song, "Localization through map stitching in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, pp. 93–105, January 2008.
- [10] —, "A new map stitching method for anchor-free localization in wireless sensor networks," in *6th IEEE International Conference on Computer and Information Technology*, 2006.
- [11] B. K. P. Horn, H. M. Hilden, and S. Negahdaripour, "Closed-form solution of absolute orientation using orthonormal matrices," *Journal of the Optical Society of America A*, vol. 5, no. 7, pp. 1127–1135, 1988.
- [12] S. Basagni, M. Battelli, M. Iachizzi, C. Petroioli, and M. Salehi, "Limiting the propagation of localization errors in multi-hop wireless networks," in *IEEE International Conference on Pervasive Computing and Communications Workshop*, 13-17 March 2006.