

A Dynamic MDS-Based Localization Algorithm for Mobile Sensor Networks

Changhua Wu

Math and Science Dept.
Kettering University
Flint, MI 48504

Email: cwu@kettering.edu

Weihua Sheng

Electrical and Computer Engineering Dept.
Oklahoma State University
Stillwater, OK, 74078

Email: weihua.sheng@okstate.edu

Wen-Zhan Song

School of Engineering and Computer Science
Washington State University
Vancouver, WA 98686

Email: songwz@wsu.edu

Abstract—In this paper, we proposed a dynamic mobility-assisted MDS-based localization algorithms for sparse mobile sensor network. For sparse networks, the assumption of the existing MDS-based localization algorithms is not necessary valid and the network may even be nonrigid, which significantly affects the application and accuracy of the existing MDS-based algorithms. In the proposed algorithm, we utilize the mobile capability of sensors in a mobile network. By moving the sensors in a random direction and recording the distances to their neighbors during the movement, virtual nodes are added. The distances between virtual nodes and real nodes provide more information about network, which leads to significantly better localization than existing methods for sparse networks. Experiments and evaluation of the proposed algorithm are provided.

Index Terms—sensor network localization MDS

I. INTRODUCTION

Recent advancements in wireless communication and micro-electro-mechanical systems (MEMS) have made possible the deployment of wireless sensor networks for many real world applications, such as environmental monitoring, search and rescue, military surveillance, and intelligent transportation, etc [1], [2], [3]. The ability of a sensor node to determine its physical location is of fundamental importance in sensor networks. In recent years, various sensor localization methods have been developed for ad hoc wireless sensor networks. Most of the node localization algorithms are based on range measurement, through either time of arrival (TOA) [4], time difference of arrival (TDOA) [5], or received signal strength (RSS) [6]. For example, In the Picoradio project [7] at UC Berkeley, a geolocation scheme for an indoor environment is provided based on RF received signal strength measurements and pre-calculated signal strength maps. The AHLoS (Ad-Hoc Localization System) [8] proposed by Savvides *et. al* enables sensor nodes to discover their locations using a set distributed iterative algorithms. An RF based proximity method was developed by [6], in which the location of a node is given as a centroid generated by counting the beacon signals transmitted by a set of beacons pre-positioned in a mesh pattern. Other methods that do not rely on range measurements were also developed. For example, the count of hops is used as an indication of the distance to the beacon nodes in some applications [4].

However, most of the localization algorithms are developed for stationary sensor networks where the sensor nodes do not move once they are deployed. Recent years have seen the growing interest in mobile sensor networks [9] where all or partial of the sensor nodes have motion capability endowed by robotic platforms. Mobile sensor networks have more flexibility, adaptivity and even intelligence compared to stationary sensor networks. Mobile sensors can dynamically reposition themselves to satisfy certain requirements on monitoring coverage, network connectivity, or fault tolerance. Localization for mobile sensor networks is very important to facilitate the information collection and the movement of mobile sensors. Researchers solving the localization problem for mobile sensor networks usually approach it from a robotics perspective, which heavily relies on the sophisticated sensors such as GPS, sonar, laser ranger finder, or camera onboard the mobile platforms. However, most of the mobile sensors have very stringent constraints on the cost and complexity. Therefore it is not practical to assume the availability of these sensors. On the other hand, localization algorithms developed for stationary sensor networks may not be a best fit for mobile sensor networks. The mobility of the sensors inspires us to address the sensor localization in a different perspective. New localization algorithms should be developed to utilize the mobility of sensors to enhance the localization performance.

To the best of our knowledge, only very limited work has been done on mobile sensor network localization. Tilak *et al.* [10] developed dynamic localization protocols for mobile sensor networks. However, their main interest is on how often the localization should be carried out in a mobile sensor network and not on the localization algorithm itself. Recently, Hu and Evans [11] proposed sequential Monte Carlo (SMC) localization method to solve the localization problem and they found that the mobility of the sensors can be exploited to improve the accuracy and precision of the localization. The SMC localization method has two steps, prediction step and filtering step. In the prediction step, the nodes use the transition distribution to predict its possible locations based on previous samples and its movement. In the filtering step, the nodes use new information received to eliminate predicted locations that are inconsistent with observations. Obviously this method is derived from the mobile robot localization

algorithms.

In this paper we are going to develop a new localization algorithm based on a distributed multidimensional scaling approach, MDS-MAP(P). In our algorithm, additional information of the network is obtained by moving sensors in a random direction and adding virtual nodes during the movement. The distances between the virtual nodes and the surrounding nodes are stored in an adjacency tables. The information from virtual nodes puts more constraints on the localization, thus leading to better performance. This paper is organized as follows: In section II we introduce MDS-MAP(P) algorithm. Section III presents a mobility assisted MDS-MAP(P) algorithm, denoted as MA-MDS-MAP(P), for mobile sensor networks. Section IV provides detailed comparison of the performance between the MDS-MAP(P) and MA-MDS-MAP(P) methods in networks of four different topologies. The influence of noise and the effect of the number of virtual nodes on the accuracy of localization are also discussed in this section. Section V concludes this paper.

II. MDS ALGORITHM FOR NODE LOCALIZATION

Our new algorithm is based on the localization algorithm developed for stationary sensor networks by Shang et al. [12]. Here we briefly review the MDS-MAP and MDS-MAP(P) algorithms.

A. MDS-MAP

MDS-MAP is based on a technique called classical multidimensional scaling (MDS) [13], which is a data analysis technique that display the structure of distance-like data as a geometrical picture. MDS takes one or more matrices representing distances or dissimilarities between objects and finds a placement of points in a low-dimensional space (two- or three-dimensional), where the distances between the points resemble the original dissimilarities. Analytical solutions are derived from the distance matrix efficiently through singular value decomposition and provide the best low-rank approximation (e.g., 2-D space) in the least squared error sense. MDS-MAP algorithm has the following three steps: [12]

- 1) Compute shortest paths between all pairs of nodes in the region of consideration. The shortest path distances are used to construct the distance matrix for MDS.

- 2) Apply the classical MDS to the distance matrix, retaining the first 2 (or 3) largest eigenvalues and eigenvectors to construct a 2-D (or 3-D) relative map.

- 3) Given sufficient anchor nodes (3 or more for 2-D networks, 4 or more for 3-D networks), the coordinates of the anchors in the relative map are mapped to their absolute coordinates through a linear transformation. The best linear transformation between the absolute positions of the anchors and their positions in the relative map is computed.

Classical MDS requires the distance between every pair of nodes. It assumes that the shortest path between two nodes provides an estimate of the Euclidean distance. This estimate is fine when the networks are dense and uniform, but is not good for very irregular ones. When the estimation is off,

the result of classical MDS is not good. The MDS-MAP(P) methods presented in the next section address this issue.

B. The MDS-MAP(P) Method

In the MDS-MAP method (namely, MDS-MAP(P)), each node computes its own local map using its local information and then the local maps are merged to form a global map. In MDS-MAP(P), each node applies MDS to compute a local map that includes only nearby nodes. For example, only those nodes within two communication hops are considered. Local maps are then merged together based on the common nodes according to the best linear transformation that transforms the coordinates of the common nodes in one map to those in the other map. The steps of MDS-MAP(P) are as follows: [12]

- 1) Set the range for local maps, R_{lm} . For each node, neighbors within R_{lm} hops are involved in building its local map.

- 2) Compute local maps. Each node does the following:
 - a) Compute shortest paths between all pairs of nodes in range R_{lm} . The shortest paths are used to construct the distance matrix for MDS.
 - b) Apply the classical MDS to the distance matrix and retain the first 2 (or 3) largest eigenvalues and eigenvectors to construct a 2-D (or 3-D) local map.
 - c) Refine the local map. Using the node coordinates in the MDS solution as the initial point, a least squares minimization is performed to make the distances between nearby nodes match the measured ones.

- 3) Merge local maps. Local maps can be merged either sequentially or in parallel.

- 4) Given sufficient anchor nodes (3 or more for 2-D networks, 4 or more for 3-D networks), transform the global map to an absolute map based on the absolute positions of the anchors.

III. MOBILITY ASSISTED MDS-MAP(P) ALGORITHM

One assumption of the MDS-based methods is that the shortest path between nodes is approximately proportional to their Euclidean distance. While this may be true if the network is dense and uniform. In the situation where a dense network is not possible due to limited resources or the network topology is not uniform, this assumption is not necessarily true. For example, in Fig. 1-(a), while the shortest path between some nodes, such as $B \rightsquigarrow F$, $B \rightsquigarrow E$, are approximately proportional to the Euclidean distances, the shortest paths between A and D , and D and E are, however, significantly larger than their Euclidean distances. In this situation, the assumption of the MDS methods is not valid anymore, which may lead to inaccurate localization. Moreover, in this graph, node A actually can be anywhere in the circle centered at B with radius equal to $|AB|$. This is due to the fact that this graph is not rigid, which means the existing constraints in the network are not enough to unambiguously determine the position of the nodes in it. In this situation, the traditional MDS-based method can't guarantee correct results. The nonproportional relation between pairwise shortest paths and the Euclidean distances, and the non-rigidity of the network are two challenges for the traditional

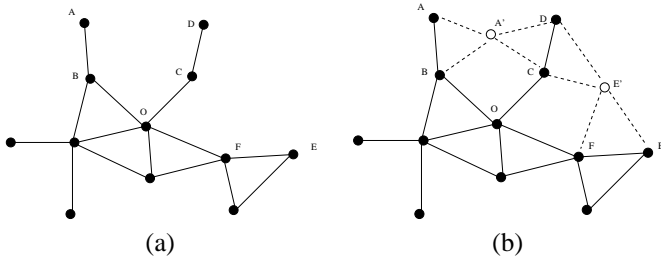


Fig. 1. Improvement on the distance estimate with node movement. (a) a sparse network. The shortest path between A and D is $A \rightarrow B \rightarrow O \rightarrow C \rightarrow D$, whose length is much larger than the Euclidean distance between A and D . The same is true for nodes D and E . (b) If we can insert a node at A' and a node at E' , then the shortest path between A and D now becomes $A \rightarrow A' \rightarrow D$. The shortest path between D and E is $D \rightarrow E' \rightarrow E$. Both of the two new shortest paths are much better approximation than those in the original network. The two new nodes can be inserted by moving A to A' and E to E' temporarily.

MDS-based methods. The first challenge affects the accuracy of the localization and the second one affects the correctness of the results.

In some type of networks, such as mobile robotic networks, each node can move around within a certain range. For this type of networks, we can utilize the mobile capability of the nodes to get additional information about the networks, thus improving the accuracy of localization. For example, in Fig. 1-(b), suppose A moves straight to a new position A' , and E moves to E' . If A' can communicate with B , C , D , and E' can communicate with D , C , and F , then several new edges, AA' , $A'B$, $A'C$, $A'D$, $E'D$, $E'C$, $E'F$, $E'E$, shown as dashed lines in Fig. 1-(b) can be added into the network. With these new edges, the shortest distances between nodes, such as $A \rightsquigarrow D$, $D \rightsquigarrow E$, $A \rightsquigarrow C$, $C \rightsquigarrow E$ are greatly improved. With the improved shortest distance between nodes, we expect the accuracy of the MDS-based localization will get better.

A. The movement of nodes

Based on the above discussion and the random walk mobility model [14], [15], we propose a mobility assisted MDS-based localization scheme for mobile sensor network, in which each sensor moves straight in a random direction for a maximum distance. During the movement, the sensor sends messages to its neighbors to update its position. After moving for the maximum distance, it tries to return back to its original position. This process is repeated for all sensors in the network. More detailed and formal description of the proposed scheme is discussed as follows.

We assume that distance between sensors within communication range can be measured reliably and all nodes in the network are mobile. During the localization process, a node can be in two status: *Moving* or *Rest*. Before the localization starts, all nodes are in *Rest* status. During the movement, a node may send messages to its neighbor at several positions, including the initial position, to add virtual nodes when certain condition is met. Each step of the proposed scheme is discussed as follows.

- A node in the network broadcasts *Start-Localization* message to start the localization process. This node can be any node in the network that discovers the necessity to start a localization process.
- Upon receiving the *Start-Localization* message, each node, denoted by v_i , in the network starts moving in the following way.
 - At the initial position or any intermediate position during the movement, v_i sends a message *AddVirtualNodes*(vid) to all of its neighbors, where vid is identification number of the virtual node to be added at this position.
 - When a neighbor, denoted by v_j , receives a *AddVirtualNodes*(vid), it measures the distance d_{ij} between v_j and v_i and sends an message *ACK*(j, d_{ij}) back to v_i .
 - When v_i receives more than three *ACK* messages from its neighbors, it sends a *ConfirmVirtualNodes*(vid) to all the neighbors and for each *ACK*(j, d_{ij}) received, it adds an entry $\langle vid, j, d_{ij} \rangle$ in the local adjacency table to record the distance between the virtual node and the neighbor. Otherwise, it sends an *AbortVirtualNodes*(vid) to its neighbors. Upon receiving a *ConfirmVirtualNodes*, the neighbor, v_j , that has sent an *ACK* message to v_i adds a new entry $\langle j, vid, d_{ij} \rangle$ into its adjacency tables. If receiving an *AbortVirtualNodes*(vid), the neighbors simply delete all message records relevant to the potential virtual node vid .
 - v_i continues moving and repeats the above steps until it finishes the movement. Then v_i sends a message *MoveStoped*(i) to its neighbors. Upon receiving this message, its neighbors update the distances to v_i in their adjacency tables. v_i also updates the distances to its neighbors after sending the *MoveStoped*(i) message. Finally, it changes its status to *Rest*.
- Proceed the localization of the network in a revised MDS-MAP(P) method, which will be discussed below.

B. The merging of local maps

After the movement of all nodes, there are two types of nodes in the network: one type is real nodes, the other type is virtual nodes added during the movement of nodes. The virtual nodes only exist in the adjacency tables of the real nodes. There is no communication between a virtual node and any other nodes. However, the distances kept in the adjacency tables provide additional information of the network. Based on these information, more precise localization can be obtained through a revised MDS-MAP(P) method. The details are discussed as follows:

- Step 1: Each node v sends requests to the nodes in its two-hop neighborhood for the adjacency tables and combine the adjacency tables of its neighbors into a local table T . After removing the duplicated entries (entries

describing the same edge), node v constructs a local graph by identifying its two-hop neighbors, shown in Fig. 2-(a). Let the two hop neighbors be denoted by S_v . v then constructs the distance matrix for nodes in S_v using only edges between nodes in S_v . The elements in the distance matrix are the shortest path between every two nodes in S_v . Obviously, S_v contains all the real nodes within two hop distance of v and some virtual nodes.

- Step 2: v builds its local relative map using the classical MDS method for nodes in S_v
- Step 3: v refines the location of nodes, including the virtual nodes, in its local relative map via least squares minimization using the distance information in T and the coordinates from the step 2 as the initial point. Let $\langle i, j, d_{ij} \rangle$ denote the entries in T and p_{ij} denote the Euclidean distance between v_i and v_j based on their coordinates. The formulation of the minimization is

$$\min \sum_{i,j \in S_v} w_{ij} (d_{ij} - p_{ij})^2 \text{ for all } \langle i, j, d_{ij} \rangle \text{ in } T \quad (1)$$

where w_{ij} is the weight for the distance between v_i and v_j . To compensate for the potential noise introduced in the movement of nodes, higher weight can be assigned to the distances between real nodes.

- Step 4: Merge the local maps until all the real nodes are included in a core map, which is grown by merging it with the local maps of neighboring real nodes. The local map with the maximum number of common nodes, including virtual nodes, with the core map is chosen to be merged with the core map. The merging can be done sequentially [12] or in a distributed way [16].
- Step 5: Transform the core map into an absolute map based on the absolute positions of anchors. Obviously, in this step, we only have to transform the coordinates of the real nodes.

Let k be the average number of nodes, including virtual nodes, in a local map. The overall complexity for computing each local map is $O(k^3)$. The total complexity for step 2 and 3 is $O(k^3n)$. Similar to the discussion in [12], the complexity of step 4 is $O(k^3n)$. For r anchors, the complexity of step 5 is $O(r^3 + n)$. So the total complexity of this revised MDS-MAP(P) method is also $O(n)$.

IV. SIMULATION RESULTS

A. Performance comparison with MDS-MAP(P) method

Simulation experiments have been carried out to evaluate the localization improvement after introducing movement under various network settings using Matlab. As we know, the accuracy of the MDS method is highly dependent on the density/degree of the network. The higher the degree of each node, the more accurate the localization is. Therefore, we would focus the experiments in evaluating the improvement of the proposed scheme in the localization of sparse networks, whose degree is under 5. We compared the performance of MA-MDS-MAP(P) with MDS-MAP(P) in four particular

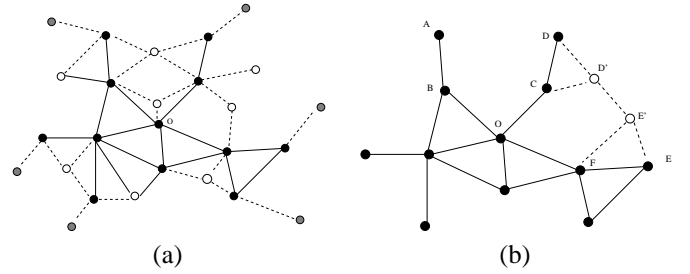


Fig. 2. (a) Only the real nodes and the virtual nodes, shown as nonfilled circles, within two hops from the center node O are kept in the local map of O . The virtual nodes shown in gray spheres are not included because they are more than two hops away from O . (b) Node D and E move to D' and E' respectively and get into communication range of each other. If they are both in *Stationary* status, then the distance between them is kept in the adjacency table of both D and E .

networks of both uniform and irregular topologies. Each column of Fig. 3 shows the error comparison on one type of network topologies. The first row shows the ground truth network, the second row shows the result of MDS-MAP(P), and the third row shows the result of MA-MDS-MAP(P). The errors are shown by a line segment starting from the actual position towards the estimated one. The longer the line segments are, the larger the error is in localization.

The field size of the four networks evaluated in this experiment is $10 * 10$. The communication range is 1.0 in all the four networks. Two virtual nodes are added per movement. The random uniform network has 200 nodes randomly positioned within the field. To prevent the nodes from getting too close to each other, the minimum distance between any two nodes is larger than 0.5. The average degree is 4.72. The regular uniform network has 144 nodes regularly positioned. There are 12 nodes in each row/column. To model the error in grid placement, the position of the nodes are adjusted by a uniformly distributed noise. The mean of the noise is 0, the range is between -0.09 and 0.09 . The average degree is 4.39. The localization for irregular networks is much more challenging than that for uniform networks because the error made in any step during the merging stage is easily propagated to the whole networks due to the limited paths of merging (the merging has to follow the topology of the networks). The average degrees of the regular C shape and random C shape networks are 3.96 and 4.12 respectively. The regular C shape network has 112 nodes regularly positioned in a C-shape field. The position of each node is adjusted by the same noise as that in the experiment of regular uniform network. The random C shape contains 150 nodes. The minimum distance between any two nodes is 0.5. As we can see from Fig.3, the error were reduced by more than 90% on the random uniform, regular uniform and regular C shape networks, and about 75% on the random C shape network, which demonstrates the improvement of the MA-MDS-MAP(P) method.

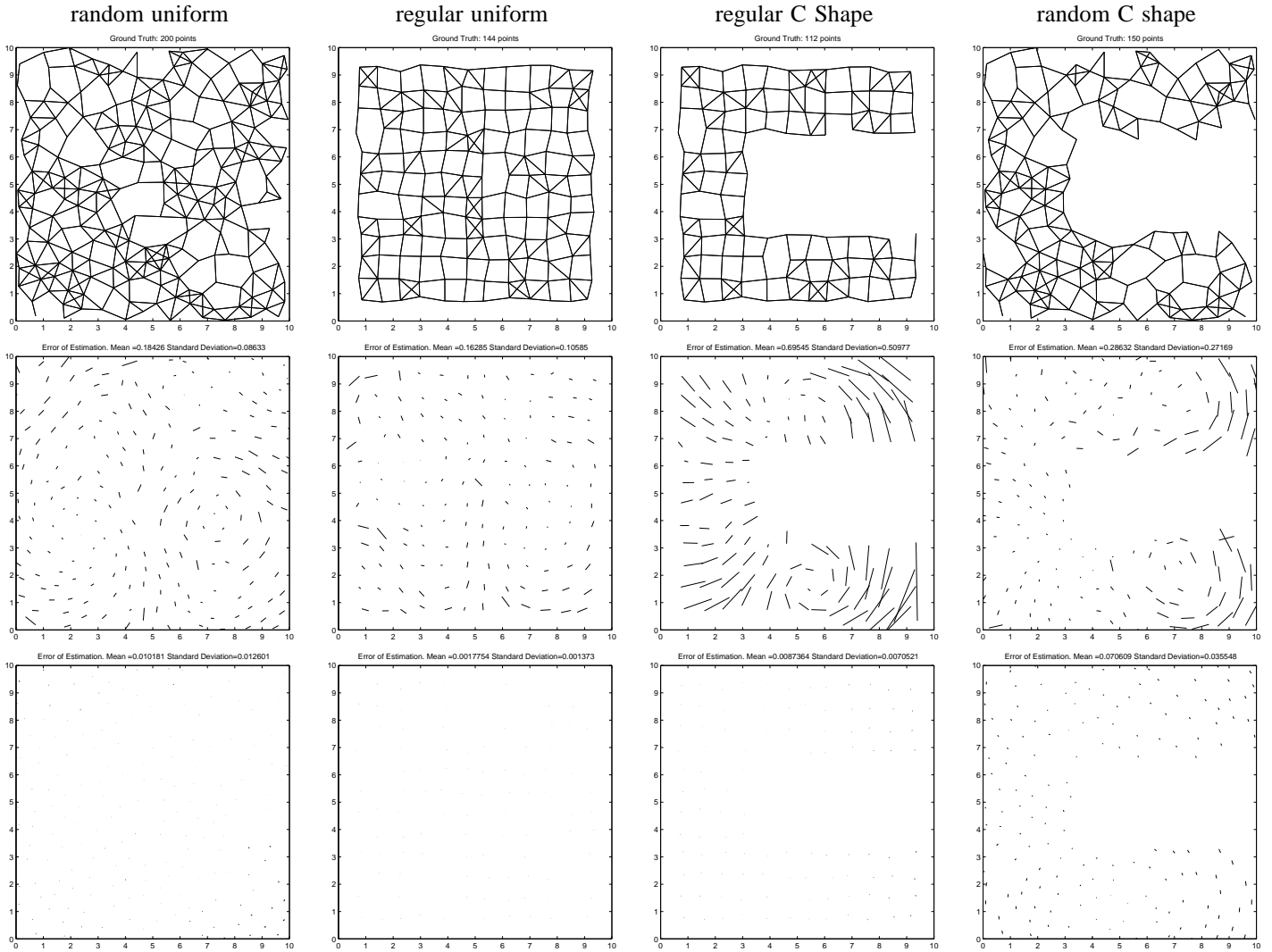


Fig. 3. Error comparisons four types of networks

B. Analysis of reliability to noise

The above evaluation is based on the assumption that the sensors can get back to their original positions after movement. However, in real application, due to various reasons, this is not necessarily true. If sensors can not get to their original position, the entries recording the distances between virtual nodes and real nodes become inaccurate. To evaluate how well the MA-MDS-MAP(P) algorithm adapts to the deviation of the sensors from their original positions, we carried out a set of experiments with various levels of additive noise to the measured distances between virtual nodes and real nodes. The mean of the noise is 0 and the deviation of the noise is from 0 to 0.15. Fig. 4 shows the results of the experiments. We can see that for uniform network, under 6 percent of additive noise, both the mean and standard deviation of the error is less than 0.05, which is 5 percent of the communication range. For C-shape networks, we see more vibrations. However, the mean and the standard deviation are

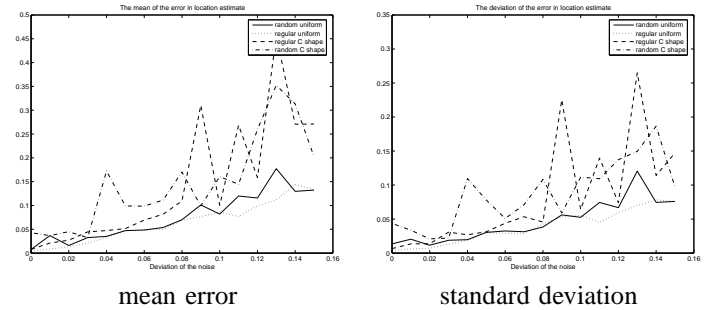


Fig. 4. Evaluation of the effect of the noise in sensor movement. The horizon axis is the deviation of the noise. The vertical axis is the estimate error.

still quite small if the deviation of the noise is less than 0.03, which is 3 percent of the communication range.

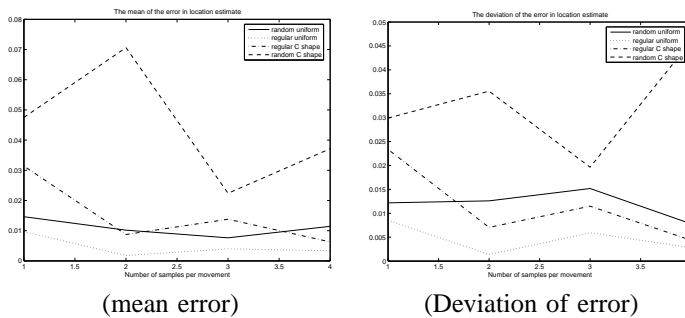


Fig. 5. Analysis of the influence of number of samples on the accuracy of localization.

C. Analysis of the number of samples per movement

The number of virtual nodes added per movement is an important factor in the localization of nodes. Usually, more virtual nodes lead to more accurate localization but also cause more communication and computation cost. Given the limited resources of sensors, we want to add as less virtual nodes as possible and at the same time get accurate results. This experiment tries to evaluate the number of virtual nodes per movement on the accuracy of localization. Similarly to the experiment of noise evaluation, we tried the experiments on four types of network topologies: regular uniform, random uniform, regular C-shape, random C-shape. The configuration (such as the number of nodes, the communication range) of each network is the same as the networks discussed in the previous experiments. The results are shown in Fig. 5. As we can see, the random C-shape is vibrating more than the other three topologies, but the mean error is still less than 0.07, about 7% of the communication range. We can also see that with the increased number of virtual nodes per movement, the localization generally gets more accurate. However, the improvement is getting less obvious as the number of virtual nodes per movement increases. Therefore, it is not necessary to add many virtual nodes per movement. Two to three virtual nodes per movement should be sufficient to get accurate localization.

V. CONCLUSION

In this paper, we proposed a mobility-assisted MDS-MAP(P) localization approach for mobile sensor network. The MDS-MAP(P) highly depends on the degree of the network, which leads to poor performance in sparse network. Based on the fact that sensors in a mobile network have limited mobile capability, in the proposed approach, more information is obtained by moving the sensors in random directions and adding virtual nodes during the movement. The distances between the virtual nodes and the real nodes are kept in adjacency tables. The virtual nodes are incorporated in building and merging the local maps. Evaluation of the performance of the proposed approach is carried out on four types of network: random uniform, random C-shape, regular uniform, and regular C-shape. The results have shown significant improvement over the MDS-MAP(P) approach on

low-degree networks. Future work may include the evaluation of various movement patterns and experiments on real mobile sensor networks to validate the simulation results.

VI. ACKNOWLEDGMENT

We give special thanks to Dr. Ying Zhang in Palo Alto research center and Professor Yi Shang in the University of Missouri-Columbia for kindly sharing with us the Matlab simulation code. The communication with Dr. Ying Zhang is especially helpful.

REFERENCES

- [1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Commun. Mag.*, 40:102–114, 2002.
- [2] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson. Wireless sensor networks for habitat monitoring. In *Proceedings of Wireless Sensor Network and Applications*, 2002.
- [3] S. N. Simic and S. Sastry. Distributed environmental monitoring using random sensor networks. In *Proceedings of the 2nd International Workshop on Information Processing in Sensor Networks*, pages 582–592, 2003.
- [4] F. Zhao and L. Guibas. *Wireless Sensor Networks: An Information Processing Approach*. Elsevier and Morgan Kaufmann Publishers, 2004.
- [5] C. Savarese, J.M. Rabaey, and J. Beutel. Locationing in distributed ad hoc wireless sensor networks. In *Proc. 2001 Int'l Conf. Acoustics, Speech, and Signal Processing (ICASSP 2001)*, volume 4, pages 2037–2040, May 2001.
- [6] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. Technical Report 00-729, Computer science department, University of Southern California, Los Angeles, CA, 2000.
- [7] J. Beutel. *Master Thesis, Geolocation in a PicoRadio Environment*. UC Berkeley, 1999.
- [8] A. Savvides, C. C. Han, and M. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *ACM SIGMOBILE*, 2001.
- [9] J. Cortes, S. Martinez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. In *IEEE Transactions on Robotics and Automation*, pages 243–255, 2004.
- [10] S. Tilak, V. Kolar, N. B. Abu-Ghazaleh, and K. D. Kang. "dynamic localization control for mobile sensor networks". In *Proceedings of IEEE International Workshop on Strategies for Energy Efficiency in Ad Hoc and Sensor Networks*, April 7-9 2005.
- [11] L. Hu and D. Evans. Localization for mobile sensor networks. In *Proceedings of MobiCom'04*, pages 45–57, 2004.
- [12] Y. Shang and W. Ruml. Improved mds-based localization. In *Proceedings of IEEE InfoCom'04*, 2004.
- [13] I. Borg and P. Groenen. *Modern Multidimensional Scaling, Theory and Applications*. Springer-Verlag, New York, 1997.
- [14] Kuo-Hsing Chiang and N. Shenoy. A 2-d random-walk mobility model for location-management studies in wireless networks. *IEEE Transactions on Vehicular Technology*, 53(2):413– 424, March 2004.
- [15] Michael McGuire. Stationary distributions of random walk mobility models for wireless ad hoc networks. In *Proceedings of the International Symposium on Mobile Ad Hoc Networking and Computing archive*, pages 90 – 98, Urbana-Champaign, IL, USA, 2005.
- [16] Y. Shang, W. Ruml, and M. Fromherz. Positioning using local maps. "Ad Hoc Networks" *Journal of Elsevier*, 4:240–253, 2006.