

HEA-Loc: A Robust Localization Algorithm for Sensor Networks of Diversified Topologies

Yuanyuan Hong, King-Shan Lui and Yik-Chung Wu

Department of Electrical and Electronic Engineering

The University of Hong Kong, Hong Kong

Email: {yyhong, kslui, ycwu}@eee.hku.hk

Abstract—In recent years, localization in a variety of Wireless Sensor Networks (WSNs) is a compelling but elusive goal. Several algorithms that use different methodologies have been proposed to achieve this goal. The performances of these algorithms depend on several factors, such as the sensor node placement, anchor deployment or network topology. In this paper, we propose a robust localization algorithm called *Hybrid Efficient and Accurate Localization* (HEA-Loc). HEA-Loc combines two techniques, Extended Kalman Filter (EKF) and Proximity-Distance Map (PDM) to improve localization accuracy. It is distributed in nature and works well in various scenarios as it is less susceptible to anchors deployment and the network topology. Furthermore, HEA-Loc has strong robustness and it can work well even the measurement errors are large. Simulation results show that HEA-Loc outperforms existing algorithms in both computational complexity and communication overhead.

I. INTRODUCTION

The main goal of Wireless Sensor Networks (WSNs) is to monitor an area of interest, but several prerequisites have to be achieved in order to reach the objective. Localization, the problem of finding the locations of the sensors, is one of these prerequisites. It has attracted significant research effort in recent years [1]–[6]. Despite many techniques have been developed, there is still no clear consensus on a particular mechanism that would be suitable in a wide range of topologies. One of the reasons is that the placement of sensor nodes has dramatic impact on the effectiveness and efficiency of localization [7]. Most algorithms require the uniform placement of sensor nodes to achieve a high accuracy, but it may not be feasible for some networks. Also, the number and locations of anchors can have great influence on the network performance. Distributed algorithms, such as APS [1] and PDM [3] usually require large number of uniformly distributed anchors to achieve a high accuracy. Centralized algorithms, like Semidefinite Programming (SDP) [8], MDS-MAP [2] and Isomap [6] can perform well based on small number of anchors, for example, three anchors for 2D space and four anchors for 3D space, but they may not work well when the anchors are not uniformly distributed. Another disadvantage of centralized mechanisms is the huge computational complexity, which far exceeds the limit of a small sensor.

In this paper, we propose a hybrid localization algorithm called Hybrid Efficient and Accurate Localization (HEA-Loc), which improves localization accuracy and efficiency with low cost in various kinds of networks. Simulation results show

that it can perform well with limited and clustered anchors in both isotropic networks and anisotropic networks, like C-shape topologies. Furthermore, it results in significant savings in computational and message overhead. Robustness is another major concern for the localization algorithm, which reflects the tolerance to measurement error. HEA-Loc has strong robustness that it still works reasonably well when the measurement error is large.

Two localization techniques are applied in HEA-Loc: Extended Kalman Filter (EKF) [9] and PDM. In the localization problem, sensors are scattered in an area and each node needs to find out its own location. There are some sensors who know their positions by external means such as GPS, and they are called *anchors*. Due to cost issue, there are only a few anchors in the network. Other nodes find out their locations based on communicating with each other and the information provided by the anchors. In our algorithm, the nodes who know their positions in the beginning are called *primary anchors*. Some nodes are selected to localize themselves using EKF. These nodes are called *secondary anchors*. After the secondary anchors know their positions, other unknown nodes can localize themselves together based on the information provided by both the primary anchors and secondary anchors by using the distributed algorithm PDM.

The paper is organized as follows. In Section II, we introduce some related works of localization in WSNs. In Section III, we give the details of HEA-Loc. Section IV presents the simulation results and we conclude the paper in Section V.

II. RELATED WORKS

Localization algorithms can be classified into centralized or distributed. Centralized algorithms collect all the data at a central point and the central point computes a global map. They usually can achieve a high accuracy in position estimation. However, information of the whole network needs to travel to a coordinator. It leads to a very high message overhead, and high computational power in the coordinator is required. Distributed algorithms allow nodes to compute their positions based on the local information. They do not need the global knowledge and have lower communication overhead. The disadvantage is that large number of anchors are needed. We now introduce some representative centralized and distributed localization algorithms.

A. Centralized Algorithms

MDS-MAP [2] makes use of distance information between every pair of nodes to compute a global map. Multidimensional scaling [10] (MDS) is a data analysis method to transform a set of objects characterized by pair-wise distances [11] between them into an embedding in the lower dimensional space. It only needs three anchors for a global map in 2D space. However, it requires all pair-wise distances and the computational complexity is high. It involves Singular Value Decomposition (SVD), and its operation complexity is $O(N^3)$, where N is the number of nodes. The uniform distribution of anchors is a prerequisite for the accurate performance.

Wang [6] proposed a centralized algorithm based on Isomap [12]. It is similar to MDS-MAP and the difference lies in the input of MDS. MDS-MAP uses the shortest path distance between each pair of nodes, while Isomap considers the weighted shortest path distances. Like MDS-MAP, it gets the relative map by applying classical MDS and uses the anchors to transform the relative map into an absolute map. The performance is sensitive to the locations of the anchors.

B. Distributed Algorithms

Lim *et al.* [3] proposed the Proximity-Distance Map (PDM) algorithm to estimate the sensor locations in anisotropic networks. In an anisotropic network, like the one in Figure 2, the shortest path distance between two nodes can be very different from their actual geographic distance. The authors devised a linear transformation that transforms a shortest path distance into a geographic distance. This mapping between shortest path distance and the geographic distance is called proximity distance mapping. Only the anchors, but not all the nodes, are involved in expensive Singular Value Decomposition (SVD), and the computational complexity is less than that of MDS-MAP. It requires no global topology information, but accuracy is seriously affected when anchors are limited or clustered.

Cheng *et al.* [5] presented a hybrid localization algorithm named HyBrid Localization (HyBloc) which combines two techniques, MDS-MAP and PDM. There are two phases in the algorithm. Some sensors are selected as secondary anchors which are localized through MDS-MAP. To pick secondary anchors, each primary anchor sends an invitation packet, which contains a counter that specifies how many secondary anchors are still needed, to its neighbors. Normal sensor receiving this packet will make a decision whether to be a secondary anchor or not by a Bernoulli trial. The counter is updated if the node has become a secondary anchor. The node forwards the invitation packet to its neighbors if more secondary anchors are still needed. After all the secondary anchors are found, in the second phase, other nodes are localized through PDM. The process to select the secondary anchors increases the message overhead.

In summary, we found out that although many localization mechanisms have been developed, it is not easy to identify a single one that is suitable for wide range of network topologies. This motivates us to develop a new localization scheme.

III. OUR APPROACH: HEA-LOC

There are two steps in HEA-Loc. First, the direct neighbors of every primary anchor are selected as the secondary anchors. Extended Kalman Filter (EKF) [9] is used to localize the secondary anchors based on the locations of the primary anchors. This can alleviate the impact of limited clustered primary anchors to the accuracy of location. In the second step, other nodes localize themselves based on the anchors (including secondary and primary together) using PDM, which is a fully distributed algorithm. Compared to other two-step approaches, HyBloc [5] and SHARP [4], which use MDS-MAP in the first step, EKF is chosen because it is relatively simpler and is more robust [9]. EKF only requires the distances between directly connected node pairs to calculate the positions while MDS-MAP requires the distances between every pair of nodes. The computational overhead of EKF is also less than that of MDS-MAP, and we will describe the details in subsection D. Because secondary anchors are direct neighbors of the primary anchors, a few messages would be enough to identify them. The message overhead is very small when compared with HyBloc and SHARP.

A. Procedure For the Algorithm

In this section, we describe the two steps in details, focusing on how nodes acquire the information for localization.

Step 1: Identification of Secondary Anchors

Each primary anchor q initiates the process to choose the secondary anchors by broadcasting a probing packet containing its ID and location to its neighboring nodes. When a node r receives the message, it knows it is a direct neighbor of a primary anchor and is selected to be a secondary anchor. r then replies its information to its primary anchor neighbor q . The information includes the ID of r , the neighbors of r , and the measured distances from r to r 's own neighbors. The distances can be obtained by any measurement method, such as RSSI, ToA or TDoA [13]. After collecting the information from all of its neighbor nodes, which are now secondary anchors, q constructs the connectivity information among itself and the secondary anchors it selects. Definitely, q knows the distance between itself and a direct neighbor. From the information provided by the secondary anchors, q also knows whether two secondary anchors are direct neighbors by themselves. Note that the neighbors of secondary anchor r which are not direct neighbors of primary anchor q are not included in the connectivity information. q then sends the connectivity information and its location to other primary anchors through a network-wide broadcast.

After a primary anchor has received the messages from all other primary anchors, it can identify the set of all primary anchors with known location at (a_{xl}, a_{yl}) , $l = 1, 2, \dots, M$, and the set of secondary anchors, whose locations are represented by (x_l, y_l) , $l = 1, \dots, Q$, where M and Q are the numbers of primary anchors and secondary anchors. The location vector of the secondary nodes is defined as $\mathbf{x} = [x_1, y_1, \dots, x_Q, y_Q]^T$. The measurement distance vector

is modeled as $\mathbf{z} = [z_1, z_2, \dots, z_L]^T$, where L is the number of neighboring node pairs. Based on \mathbf{z} and (a_{xl}, a_{yl}) , $l = 1, 2, \dots, M$, each primary anchor can calculate \mathbf{x} , the positions of the secondary anchors, using EKF. The primary anchor can then inform its secondary anchors their locations.

Step 2: Localization of Other Nodes

To localize other nodes in the network, the PDM mechanism is adopted. Both the primary and secondary anchors participate in the construction of the proximity distance mapping. Each anchor, primary and secondary, broadcasts a message containing its ID and location to every node in the network. The proximity distance mapping is broadcast to every node as well. Then, a node which still does not know its position can get the path distance to each of the secondary and primary anchors. This node can then calculate its own position based on the information by multilateration.

B. Kalman Filter

We employ EKF to localize the secondary anchors. EKF has been very popular for mobile localization and there are many studies in recent years [14]–[16]. However, they usually consider using EKF to track the location of one mobile target at one time. On the other hand, in HEA-Loc, we localize all secondary anchors by EKF at once, and is more efficient.

First, we define the function for calculating Euclidean distance between two points (x_1, y_1) and (x_2, y_2) as

$$h((x_1, y_1), (x_2, y_2)) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}. \quad (1)$$

Assuming there are totally L_1 direct connections between primary anchors and secondary anchors. Then, all the measured distances between primary anchors and secondary anchors can be grouped into a $L_1 \times 1$ vector $\mathbf{z}_1 = h_1(\mathbf{x}) + \mathbf{v}_1$ with elements of $h_1(\mathbf{x})$ given by $h((x_p, y_p), (a_{xl}, a_{yl}))$ where (x_p, y_p) represents the unknown location of a secondary anchor, and (a_{xl}, a_{yl}) represents the known location of the l^{th} primary anchor. The vector \mathbf{v}_1 is the measurement noise vector with independent and identically Gaussian distributed elements. Similarly, we can put the L_2 measured distances between different directly connected secondary anchors into a vector as $\mathbf{z}_2 = h_2(\mathbf{x}) + \mathbf{v}_2$ with elements of $h_2(\mathbf{x})$ given by $h((x_p, y_p), (x_q, y_q))$, and \mathbf{v}_2 represents the noise vector with statistical properties the same as \mathbf{v}_1 . Furthermore, \mathbf{z}_1 and \mathbf{z}_2 can be stacked into a vector of length $L = L_1 + L_2$ as

$$\mathbf{z} = \begin{pmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{pmatrix} = \begin{pmatrix} h_1(\mathbf{x}) \\ h_2(\mathbf{x}) \end{pmatrix} + \underbrace{\begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix}}_{\triangleq \mathbf{v}} \triangleq f(\mathbf{x}, \mathbf{v}). \quad (2)$$

From equation(1)(2), it is clear that $f(\mathbf{x}, \mathbf{v})$ is a non-linear function of the Q unknowns in \mathbf{x} , so we use EKF in HEA-Loc. We introduce a vector \mathbf{x}_k , which represents the state of \mathbf{x} at times k . Using the framework of EKF, we can write the system model as

$$\mathbf{x}_k = \mathbf{x}_{k-1} = \mathbf{x}. \quad (3)$$

$$\mathbf{z} = f(\mathbf{x}_k, \mathbf{v}). \quad (4)$$

Equation (3) is the evolution model of the dynamic system and equation (4) is the measurement equation.

Following the derivation of EKF, it can be shown that the estimate of the unknown vector \mathbf{x} can be obtained iteratively as follows (starting with $k = 1$):

$$\mathbf{K}_k = \mathbf{P}_{k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k-1} \mathbf{H}_k^T + \mathbf{M}_k \mathbf{R} \mathbf{M}_k^T)^{-1}. \quad (5)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1} + \mathbf{K}_k (\mathbf{z} - f(\hat{\mathbf{x}}_{k-1}, 0)). \quad (6)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k-1}. \quad (7)$$

where \mathbf{R} is a $L \times L$ diagonal matrix with the diagonal elements being the variance of the measurement noise; \mathbf{H}_k and \mathbf{M}_k are defined as the following matrices:

$$\mathbf{H}_k = \left. \frac{\partial f}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_{k-1}}, \quad \mathbf{M}_k = \left. \frac{\partial f}{\partial \mathbf{v}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_{k-1}}.$$

Note that EKF is iterative and it needs to be initialized with $\hat{\mathbf{x}}_0$ and \mathbf{P}_0 . In Kalman filter literature, usually, $\hat{\mathbf{x}}_0$ is taken as a $2Q \times 1$ zero vector and \mathbf{P}_0 is taken as a $L \times L$ identity matrix. The algorithm alternates among equations (5)(6)(7) until $\|\hat{\mathbf{x}}_k - \hat{\mathbf{x}}_{k-1}\| \leq 0.01$.

C. Proximity-Distance Map (PDM)

After the secondary anchors obtain their locations, all secondary anchors and primary anchors help the remaining nodes to localized themselves by PDM. Given that there are $M' = M + Q$ anchors, the shortest path distance from node i to M' anchors are collected in the proximity vector, $\mathbf{p}_i = [p_{i1}, \dots, p_{iM'}]^T$, where p_{ij} is the shortest path distance measured from the i^{th} node to the j^{th} anchor. Similarly, denoting the geographic actual distance from the i^{th} node to the j^{th} anchor as l_{ij} , the geographic distance vector for node i is defined as $\mathbf{l}_i = [l_{i1}, \dots, l_{iM'}]$.

Now, assuming the indices for the primary and secondary anchors are $i = 1, \dots, M'$, then, with $\mathbf{P} \triangleq [\mathbf{p}_1, \dots, \mathbf{p}_{M'}]$, and $\mathbf{L} \triangleq [\mathbf{l}_1, \dots, \mathbf{l}_{M'}]$, we can compute a linear transformation \mathbf{T} defined as

$$\mathbf{T} = \mathbf{L} \mathbf{P}^T (\mathbf{P} \mathbf{P}^T)^{-1}. \quad (8)$$

Finally, given a node s with unknown location, but can obtain its proximity vector \mathbf{p}_s by shortest path calculation, its actual geographic distances from all the anchors can be estimated as:

$$\hat{\mathbf{l}}_s = \mathbf{T} \mathbf{p}_s. \quad (9)$$

To ensure numerical stability when computing equation (8), $(\mathbf{P} \mathbf{P}^T)^{-1}$ is usually calculated by the Singular Value Decomposition (SVD). After getting an estimate of the true distances from anchors, we can use conventional methods, such as trilateration, to obtain the location of the node s .

D. Complexity Analysis

To analyze the complexity, we consider an instance that there are M primary anchors and N nodes to be localized. Suppose that there are Q secondary anchors. We consider the complexity in terms of the communication cost and computational cost.

• **Communication cost**

In the first step, primary anchors broadcast the messages to the direct neighbors to select the secondary anchors. Furthermore, HEA-Loc only requires the distances of neighboring nodes, and each secondary node just sends the neighboring information to its primary anchor instead of flooding the message to the whole network.

In the second step, other nodes conduct the localization based on the secondary anchors and primary anchors by PDM, which is distributed. Each node just has to know the shortest path distance to the anchors. The secondary anchors and primary anchors flood the message in the networks, which gives a lower bound of $(M + Q) * N$ to the communication cost.

Compared to the fully-distributed algorithms, the secondary anchors introduce some extra messages. In our problem, the clustered primary anchors are limited, so the number of secondary anchors is small compared to the total number of nodes. Centralized Algorithm MDS-MAP requires the pair-wise distance between any two nodes, and its communication cost is $O((M + N + Q)^2)$. Therefore, the message overhead of our algorithm is comparable to that of PDM and far less than that of MDS-MAP.

• **Computational cost**

The dominant operation of EKF is the matrix inversion. The scale of the matrix is determined by the number of directly connected node pairs among primary and secondary anchors, which is small when compared to the scale of the network (around 20). PDM in the second step involves the computation of Singular Value Decomposition (SVD) of a matrix, whose complexity is $O((M + Q)^3)$. For the original PDM, the complexity is $O(M^3)$, since there is no secondary anchor involved. For MDS-MAP, the main operation is the computation of the Singular Value Decomposition (SVD) of a matrix. The complexity is $O((M + N + Q)^3)$.

From the discussion above, our algorithm complexity is far smaller than that of MDS-MAP and comparable to the PDM in term of communication cost and computational cost, when there are only a few anchors.

IV. SIMULATION

To study the performance of our algorithm, we conducted extensive simulations. We ran different algorithms on various topologies of networks in MatLab Version 7.2. There are two main types of topologies generated: uniform and anisotropic. In uniform networks, we randomly placed sensors in a square area of 100x100. In anisotropic topologies, sensors are randomly put in a C-shaped area. Figure 1 and Figure 2 are examples of the two types of networks. We tested 100 different topologies for each kind of networks, each contains 200 nodes.

We compared HEA-Loc with the centralized algorithm MDS-MAP, distributed algorithm PDM, and two-step approach HyBloc. Nodes calculate the locations based on the distances measured between the directly connected neighbors.

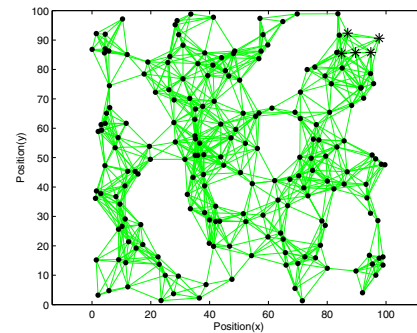


Fig. 1. 200 nodes randomly placed in a square

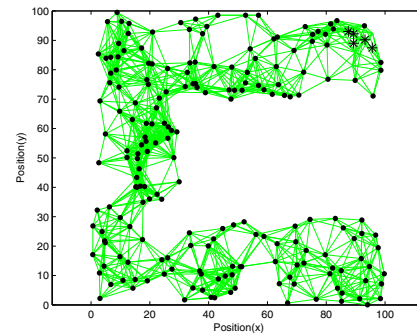


Fig. 2. 200 nodes randomly placed in a C-shape area

Measurements are subjected to the random noises. The measurement noise is modeled as i.i.d. Gaussian random variable with zero mean and variance σ^2 . To compare HEA-Loc with other algorithms, the errors are normalized to the communication range (R). In the following section, we measured the performance in terms of accuracy, noise sensitivity, and message overhead.

A. Location Accuracy

Accuracy implies the closeness of the location estimate to the ground truth, which is an important issue to evaluate the localization algorithms. Figure 3 shows the performances of different algorithms in the uniform networks with 3 anchors

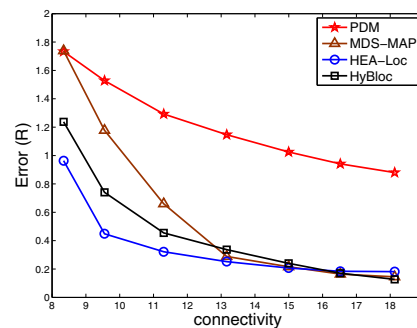


Fig. 3. uniform network, 3 random anchor placement, $\sigma = 0.1$

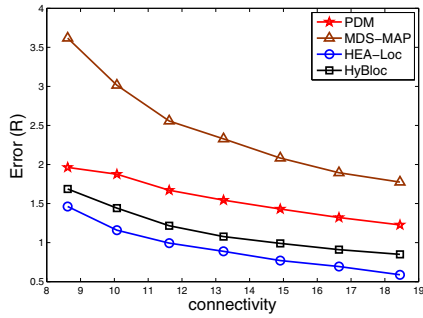


Fig. 4. C-shape network, 3 random anchor placement, $\sigma = 0.1$

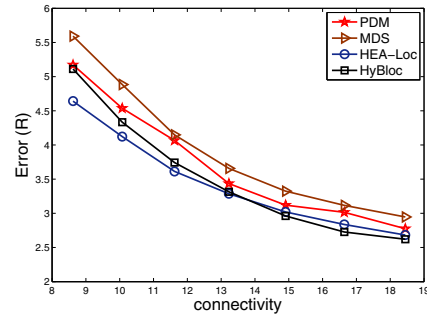


Fig. 6. C-shape network, 5 anchors clustered in the corner, $\sigma = 0.2$

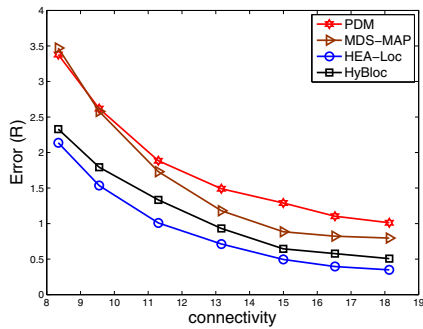


Fig. 5. uniform network, 5 anchors clustered in the corner, $\sigma = 0.2$

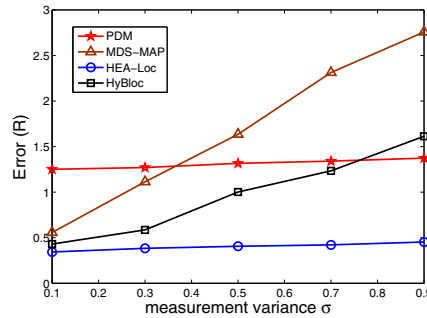


Fig. 7. uniform network with random anchors, connectivity=11.2

randomly placed. We varied the communication range between 13m to 18.3m so that the connectivity level falls between 8.1 to 18.2. With the connectivity increases, the performances of algorithms are all improved. However, the rates of improvement are different. When the connectivity is less than 13, MDS-MAP performs worse than HEA-Loc. When the connectivity becomes larger, the accuracy of MDS-MAP increases a lot. HEA-Loc is less susceptible to the connectivity. HyBLoc performs better than MDS-MAP and PDM. When the connectivity is small, its error is larger than that of HEA-Loc. With the connectivity increases, its error reduces a lot, because it uses MDS-MAP to localize the secondary anchors and MDS-MAP works well when connectivity is large. As for the PDM, HEA-Loc is better than it by 0.8R consistently.

Figure 4 shows the performances in the C-shape networks with 200 nodes. There are 3 anchors randomly placed in the C-shape area and noise standard deviation σ is 0.1m. MDS-MAP does not perform well in anisotropic networks and its error is between 1.6R to 3.6R in our simulations. PDM adapts to localize the nodes in the anisotropic networks, but it requires more anchors. For 3 anchors are limited, its performance is not good and error is between 1.1R to 2R. HyBLoc outperforms MDS-MAP and PDM but works worse than HEA-Loc. HEA-Loc is better than HyBLoc by 0.2R.

Figure 5 shows the performances of different algorithms with the clustered anchors. An example network is shown in Figure 1 where the anchors represented as stars are at the top right hand corner. The measurement noise variance is $\sigma=0.2m$.

The error of PDM is between 3.5R and 1R and MDS-MAP varies between 3.5R and 0.7R. HyBLoc outperforms PDM and MDS-MAP approximately by 1.3R. HEA-Loc performs better than PDM, MDS-MAP and HyBLoc consistently.

Figure 6 presents the performances in the C-shape network topologies, which is shown in Figure 2. There are 5 anchors clustered in the corner, represented as stars in Figure 2. HEA-Loc error is less than that of PDM and MDS-MAP consistently. When the connectivity is small, HEA-Loc performs better than HyBLoc. The error of HyBLoc decreases fast with the connectivity increasing. The reason is that HyBLoc uses the MDS-MAP which depends on the connectivity. It outperforms HEA-Loc for a small degree when the connectivity is large, but it needs more message overhead, which is illustrated in Section C.

B. Robustness to Noise

Robustness is another major concern for the localization algorithm, which consists of tolerance to communication problems and inaccurate distance. Figure 7 and Figure 8 present the performances in uniform topologies under different degrees of measurement errors. Figure 7 shows the performances in the uniform networks with random anchors. The measurement noise standard deviation varies from 0.1m to 0.9m. With the noise increasing, the estimation error increases for every algorithms. However, the effects of the noise are different. MDS-MAP is more sensitive to the noise. When the measurement error is small, its performance is better than that of PDM. When the measurement error becomes large, its performance

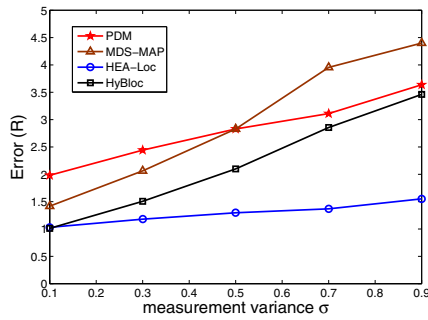


Fig. 8. uniform network with clustered anchors, connectivity=11.2

deteriorates fast. The reason is that the MDS-MAP requires all-pair-distance matrix, which depends on the noise. PDM is less dependent on the noise. HyBloc is affected by the measurement error because it uses the MDS-MAP to localize the secondary anchors. HEA-Loc is less susceptible to the noise, which increases 0.1R with the noise increasing. For the networks with clustered anchors, Figure 8 shows that the trend is the same as that with random anchors. HEA-Loc can still tolerate the noise deviation from 0.1m to 0.9m.

C. Message Overhead for Secondary Anchors

We have analyzed the complexity of different algorithms theoretically in the previous section. In this section, we compared HEA-Loc with HyBloc through the simulation. For HyBloc and HEA-Loc both use two-step approach and use PDM in the second step, the differences of communication cost lie in the message overhead used for identifying and localization of the secondary anchors in the first step. There are 2 kinds in message overhead. One is the secondary anchor selection and the other is for collecting information for localization. In HyBloc, primary anchors send the invitation packet traveled in the network to select the secondary anchors. HEA-Loc chooses the nodes directly connected to the primary anchors as the secondary anchors, so each primary anchor just broadcasts one message, which reduces the message overhead by 70%. On the other hand, HyBloc requires the all pair distances between anchors by using MDS-MAP, so anchors (including secondary and primary) have to floods the messages to get the shortest path. Every node receiving the message at the first time will store the proximity. If the message from a particular anchor has been received before, the node will check whether the new proximity is larger than the stored proximity. If so, the message will be discarded. Otherwise, the value will be updated and the message will be forwarded to other neighbors. For example, suppose there are 200 nodes and 20 anchors (including secondary and primary), and each node has to send one message for each anchor at least, which gives a lower bound to 20×200 . EKF only needs the distances between directly connected neighbors that each secondary anchor broadcasts one message and exchange the information between primary anchors. HEA-Loc can reduce the message overhead a lot for secondary anchors and cut

Different Algorithms	random anchors		clustered anchors	
	Selection	Information	Selection	Information
HyBloc	17	7331	39	6560
HEA-Loc	3	226	5	214

TABLE I
NUMBER OF MESSAGES GENERATED IN THE FIRST STEP

down the message overhead in the whole network.

V. CONCLUSION

In this paper, we have developed a robust localization algorithm to improve the accuracy with low cost in various networks, especially for the networks with limited clustered anchors. It works well in various scenarios, including both isotropic and anisotropic networks, because it is less susceptible to the anchors deployment and the topology of networks. In HEA-Loc, we combined the Extended Kalman Filter and PDM together, which makes a tradeoff between the performance and the cost. Compared to other algorithms, simulation results show that the accuracy can be improved by HEA-Loc. Furthermore, HEA-Loc is less sensitive to the measurement noise. We evaluated HEA-Loc under different degrees of measurement noise in the simulation. It can still work well when the noise is large. Overall, HEA-Loc provides a less expensive and more accurate alternative for localization in WSN.

REFERENCES

- [1] D. Niculescu and B. Nath, "Ad hoc positioning system (APS)," in *IEEE Globecom*, 2001, pp. 2926 – 2931.
- [2] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, "Localization from mere connectivity," in *ACM MobiHoc*, 2003, pp. 201–212.
- [3] H. Lim and J. C. Hou, "Localization for anisotropic sensor networks," in *IEEE INFOCOM*, 2005.
- [4] A. A. Ahmed, H. Shi, and Y. Shang, "Sharp: A new approach to relative localization in wireless sensor networks," in *IEEE ICDCS*, 2005.
- [5] K.-Y. Cheng, K.-S. Lui, and V. Tam, "Hybloc: Localization in sensor networks with adverse anchor placement," in *Sensors*, vol. 9, no. 1, January 2009, pp. 253–280.
- [6] C. Wang, J. chen, Y. Sun, and X. Shen, "Wireless sensor networks localization with isomap," in *International Conference on Communications*, July 2009.
- [7] S. S. Dhillon and K. Chakrabarty, "Sensor placement for effective coverage and surveillance in distributed sensor networks," in *IEEE WCNC*, 2003.
- [8] P. Biswas and Y. Ye, "Semidefinite programming for ad hoc wireless sensor network localization," in *IEEE IPSN*, 2004, pp. 46–54.
- [9] D. Simon, *Optimal State Estimation*. Wiley-Interscience, 2006.
- [10] J. B. Kruskal and M. Wish, "Multidimensional scaling," in *Sage Publications*, 1977.
- [11] L. R. Ford and D. R. Fulkerson, *Flows in Networks*, Princeton: Princeton University Press, 1962.
- [12] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric frameworks for nonlinear dimensionality reduction," in *SCIENCE*, Dec 2000, pp. 2319–2323.
- [13] B. Hofmann-Wellenho, H. Lichtenegger, and J. Collins, *Global Positioning System: Theory and Practice*. Springer-Verlag, 2001.
- [14] S. Kwon, K. Yang, and S. Park, "An effective kalman filter localization method for mobile robots," in *International Conference on Intelligent Robots and System*, Oct 2006, pp. 1524–1529.
- [15] K. B. Purvis, K. J. Astrom, and M. Khammash, "Estimation and optimal configurations for localization using cooperative uavs," vol. 16, no. 5, pp. 947–958, Sep 2008.
- [16] G. Binazzi, L. Chisci, F. Chiti, R. Fantacci, and S. Menci, "Localization of a swarm of mobile agents via unscented kalman filtering," in *International Conference on Communications*, July 2009.