LaMSM: Localization Algorithm with Merging Segmented Maps for Underwater Sensor Networks

Eunchan Kim, Seok Woo, Chungsan Kim, and Kiseon Kim

Department of Information and Communications,
Gwangju Institute of Science and Technology (GIST),
1 Oryong-dong, Buk-gu, Gwangju, 500-712, Republic of Korea
{tokec, swoo, only2442, kskim}@gist.ac.kr

Abstract. Underwater sensor networks (UWSNs) are considered a cost-effective solution to ocean applications, such as the acquisition of natural resources in oceans, protection from underwater disasters, etc. These applications basically require location information of nodes to identify the venue of reported events. To locate more accurately the position of nodes, multidimensional scaling (MDS) is widely used because of its good tolerance to errors in measured distances. MDS requires measured distances between every pair of nodes but in practice, only distances between nodes within a communication range can be measured. Hence, the well-known MDS-MAP(P) [6] calculates unmeasured distances for MDS but these calculations result in large errors. In this paper, we proposed a localization algorithm with merging segmented maps (LaMSM) that constructs many reliable segmented maps composed of only nodes within a communication range, and then merges them together based on their common nodes. The segmented maps are built from only the measured distances and as a result, LaMSM provides more accurate node positions than MDS-MAP(P).

Key words: sensor networks, optimization, localization, multidimensional scaling.

1 Introduction

Recently, many countries have been turning their interest toward underwater applications to acquire natural resources in oceans, to monitor pollution, to prevent disasters, etc. For such applications, there are many underwater systems using autonomous underwater vehicles (AUVs) which explore and gather geological features from the ocean floor. However, traditional underwater devices usually have difficulties in real-time monitoring, high expense, and recovery from failure. To overcome these difficulties, underwater sensor networks (UWSNs) are emerging as a cost-effective solution by monitoring events near sensor nodes deployed underwater [1]. The major difference between UWSNs and terrestrial sensor networks is the communication channel among sensor nodes. Due to the limited transmission range of high radio frequency (RF) signals, acoustic signals are alternatively used for UWSNs because of their long transmission range: up to 1Km, 10Km, and 100Km depending on the used bandwidth and frequency.

Acoustic signals experience high delays due to the low speed of sound underwater: about 1.5×10^3 m/s [1, 2]. High delays of acoustic signals can provide more accurate distances between nodes through resolving the delays called time of flight (ToF) than in the case of RF signals. Once distances between nodes are secured, sensor nodes can be localized with useful traditional localizations. Among the basic methodologies for localization, multidimensional scaling (MDS) is a popular mathematical tool due to its good tolerance to errors in measured data. MDS computes the positions of nodes from both connectivity and distance between nodes [3]. Applying MDS to localization needs distance information between every pair of nodes in a network but in practice, a node can measure only the distances to nodes within a communication range.

For complete distance information, MDS-MAP calculates the unmeasured distance by summing measured distances along the shortest multi-hop path between nodes [4]. It then constructs a global map where all nodes are located and relocates the global map with the given beacon nodes. However, errors in calculating the unmeasured distance becomes greater as the number of hop counts in the shortest path increases, which considerably affects the localization error of nodes. To patch MDS-MAP, MDS-MAP(P) restricts the hop count to 2 or 3 hops in calculating unmeasured distances, which prevents errors in distances from increasing [5, 6]. Hence, MDS-MAP(P) first builds local maps for 2-hop or 3-hop areas and then merges them together to construct a global map. It performs well but there are still open problems to improve accuracy: how to obtain more accurate local maps and how to merge them with minimum error.

In this paper, we propose a localization algorithm with merging segmented maps (LaMSM) which constructs segmented maps with only fully connected nodes, merges segmented maps to build a local map, and merges local maps again to form a global map. Because the segmented maps are the basic units for a global map, it is important to reduce localization error in a segmented map. To build a reliable segmented map with MDS, LaMSM uses only measured distances by grouping only the nodes fully connected to one another. The rest of this paper is organized as follows: Section II summarizes literature related to our current research, Section III explains our proposed algorithm in detail, and Section IV presents the simulation results under varying errors in distance measurements. Finally, we conclude this paper in Section V.

2 Related Works

In this section, we explain several distance-based localizations that construct a local map with measured distances and then merge them together. These are well known for their good performance using a small number of beacon nodes.

MDS-MAP(P) proposed by Shang et al [5] uses multidimensional scaling (MDS) and local distance information. Most local distances between nodes can be obtained by resolving received packets but there still remain unmeasured distances in a local area. Building local maps with the MDS requires fully complete distance information for all nodes in a local area so that MDS-MAP(P) calculates the sum of measured distances along the shortest path between two nodes instead of unmeasured distances.

Moore et al presented a distributed localization which constructs robust quadrilaterals for local maps and merges them together [7]. The quadrilateral is constructed with a trilateration method using distances among four nodes. When two quadrilaterals are merged, a map is sometimes reflected and merged into another map because of a serious position error in quadrilaterals called flip ambiguity. To prevent flip ambiguity, Moore et al's scheme collects only robust quadrilaterals in which four nodes should have full connections to one another and greater than a pre-defined angle between them. This restriction could reduce the possibility of flip ambiguity but it also excludes the chance for other node maps to be used as quadrilaterals. Only robust quadrilaterals are merged together using the closed-form solution suggested by Horn et al [9].

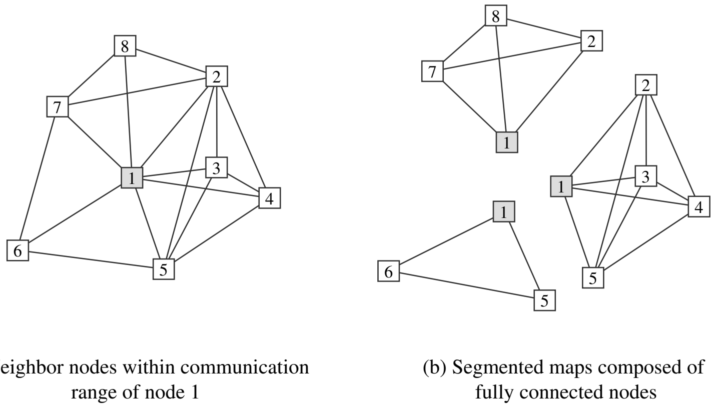
While the previous localizations take into consideration local map construction with constraints and basic mathematical tools, e.g., MDS or trilateration, Kwon et al turned their focus to a merging method to overcome flip ambiguity in their localization [8]. They utilized a merging method based on the closed-form solution of Horn et al and additional distance information. In addition, they investigated performance between maps with the MDS method and the multilateration. According to their results, merging MDS-based local maps shows better localization accuracy than merging multilateration-based local maps. However, the MDS-based local maps are constructed with the same mechanism as the MDS-MAP(P), so that the local maps also have errors due to the calculated distances as well as errors due to measurement error.

3 Proposed Localization Algorithm

This section discusses the causes of localization inaccuracy of local maps in MDS-MAP(P) and describes the proposed localization algorithm which also follows the overall structure of MDS-MAP(P) but adopts different mechanisms to construct local maps and to merge them.

3.1 Problem Statement

Building a local map with MDS, MDS-MAP(P) calculates all distances between every pair of nodes within a local area by summing the measured distances along the shortest path between them. In the calculated distance between distant nodes, errors arising from indirect paths degrade the accuracy of nodes' positions in a local map rather than errors in measurements. Fig. 1(a) shows the real deployment of node 1 and its neighbors, where a line between two nodes stands for a direct connection. For instance, node 2 cannot directly communicate with



(a) Neighbor nodes within communication range of node 1

(b) Segmented maps composed of fully connected nodes

Fig. 1. Node deployed example within communication range of node 1: (a) real deployment of node 1 and its neighbor nodes, (b) segmented maps of node 1 of the LaMSM.

node 6, so that it cannot measure the distance to node 6. In this case, MDS-MAP(P) calculates the distance as $d_{26} = \min(d_{21} + d_{16}, d_{27} + d_{76}, d_{25} + d_{56})$: whatever it selects is an indirect distance for the real distance between node 2 and node 6.

3.2 LaMSM: Localization Algorithm with Merging Segmented Maps

In the proposed LaMSM, a segmented map is newly defined as a basic unit to make a reliable local map. LaMSM is composed of two phases: building a local map with segmented maps at every node and constructing a global map with collected local maps at a special node.

A. Building Segmented Maps from Measured Distances Each node measures distances from itself to neighbor nodes by resolving delays at received acoustic packets, and then sends and receives the distance information to neighbor nodes. Then, each node realizes most distances between neighbor nodes but still has some unmeasured distances between them. To avoid calculating the unmeasured distance, it segments measured distances into sub-distance sets. Elements in a sub-distance set are distances between nodes which are fully connected to one another, so that there is no unmeasured distance in a sub-distance set. Applying MDS to each sub-distance set, each node can immediately build a few segmented maps which are the basic units to construct a local map. For instance, node 1 can build three segmented maps with sub-distance sets, as shown in Fig. 1(b). All nodes in each segmented map are connected to one another, so that they can measure all distances between them. Hence, node 1 does not need to calculate any unmeasured distance like d_{26} to build segmented maps but it is necessary to merge three segmented maps together for a local map in Fig. 1(a).

B. Merging Segmented Maps for a Local Map Once segmented maps are secured, the next work of each node is to merge them for a local map. Each node selects a segmented map which has the most nodes among the segmented maps, as a reference map. The other segmented maps are merged into the reference map sequentially by relocating them on the coordinates of the reference map. A merging order is determined with the number of common nodes between a reference map and the other segmented maps. Hence, it is important to minimize errors arising in every merging time, otherwise errors continuously increase at the next merging time.

Let's consider two maps: a reference map and a segmented map, which have m common nodes in the p-dimensional Euclidean space, usually $p = 2$ or 3. The position matrices for m common nodes are represented as $\boldsymbol{A} = [\boldsymbol{a}_j]_{p\times m}$ in a segmented map and $\boldsymbol{B} = [\boldsymbol{b}_j]_{p\times m}$ in a reference map, respectively. While merging the segmented map to a reference map, each node computes three parameters: a scaling parameter s, an orthonormal rotating (optionally reflecting) matrix $\boldsymbol{R}$, and a translating vector $\boldsymbol{t}$, in order to minimize the following discrepancy error $E(s, \boldsymbol{R}, \boldsymbol{t})$ between two common node sets:

$$E(s, \boldsymbol{R}, \boldsymbol{t})^2 = \sum_{j=1}^{m} ||s\boldsymbol{R}\boldsymbol{a}_j + \boldsymbol{t} - \boldsymbol{b}_j||^2 \ . \tag{1}$$

For the parameters, we adopt Umeyama's method [10] which determines the rotating matrix $\boldsymbol{R}$ with the singular value decomposition. According to Umeyama's method, three parameters can be expressed as

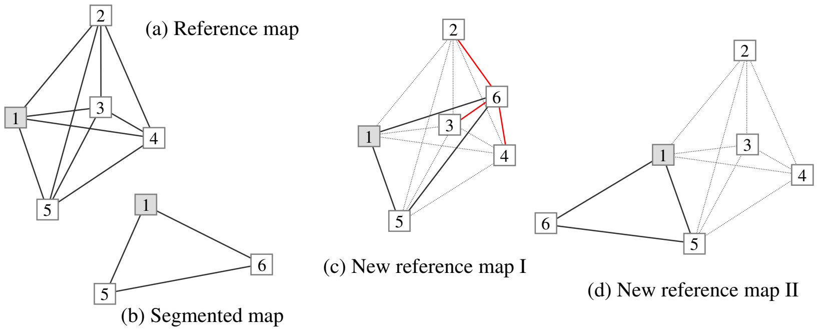
$$\begin{aligned} \boldsymbol{R} &= \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T \ , \\ s &= \frac{\mathrm{trace}(\boldsymbol{D}\boldsymbol{S})}{\frac{1}{m}\sum_{j=1}^{m} ||\boldsymbol{a}_j - \bar{\boldsymbol{a}}||^2} \ , \\ \boldsymbol{t} &= \bar{\boldsymbol{b}} - s\boldsymbol{R}\bar{\boldsymbol{a}} \ , \end{aligned} \tag{2}$$

where $\bar{\boldsymbol{a}} = \frac{1}{m}\sum_{j=1}^{m} \boldsymbol{a}_j$, $\bar{\boldsymbol{b}} = \frac{1}{m}\sum_{j=1}^{m} \boldsymbol{b}_j$, $\mathbb{A} = [\boldsymbol{a}_j - \bar{\boldsymbol{a}}]_{p\times m}$, $\mathbb{B} = [\boldsymbol{b}_j - \bar{\boldsymbol{b}}]_{p\times m}$, $\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T = \mathrm{svd}(\mathbb{B}\mathbb{A}^T)$, and

$$\boldsymbol{S} = \begin{cases} \boldsymbol{I} & \text{if } \det(\mathbb{B}\mathbb{A}^T) \geq 0 \\ \mathrm{diag}(1, 1, \cdots, 1, -1) & \text{if } \det(\mathbb{B}\mathbb{A}^T) < 0 \ . \end{cases} \tag{3}$$

While merging two segmented maps, we do not use the scaling parameter in Eq. 2 to preserve the scale of all segmented maps as well as a reference map.

Because some segmented maps are given as being reflected in the coordinates of a reference map, rotating matrix $\boldsymbol{R}$ needs to determine whether it reflects a segmented map or not, as well as to rotate it. However, Umeyama's method does not provide such a determination to reflect in the rotating matrix $\boldsymbol{R}$. Furthermore, it is hard to determine reflection of a segmented map only with common nodes when there are outliers in common nodes or when common nodes are almost placed on a straight line.



(a) Reference map

(b) Segmented map

(c) New reference map I

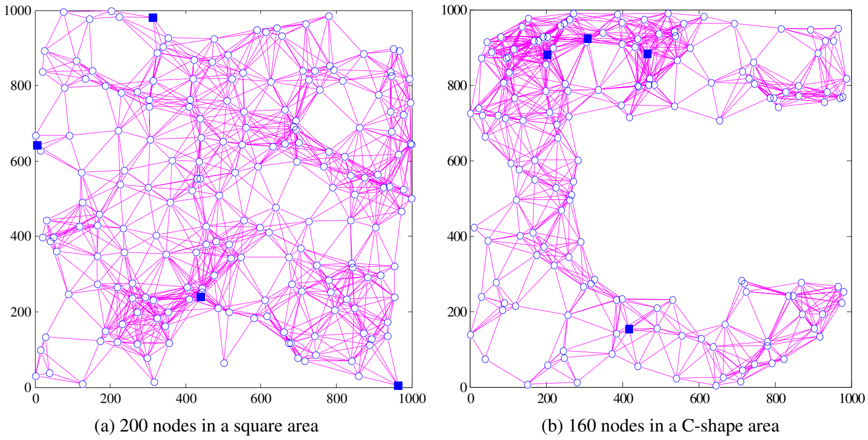
(d) New reference map II

Fig. 2. Merging segmented maps for a local map: (a) reference map at node 1, (b) segmented map overlapped with a reference map at node 1, (c) new reference map I after merging an original segmented map, (d) new reference map II after merging a reflected segmented map.

To determine reflection of a segmented map, LaMSM considers two kinds of segmented maps: an original segmented map and a reflected segmented map. LaMSM relocates two segmented maps on the coordinates of a reference map with three parameters in Eq. 2, respectively. Then, LaMSM can obtain independently two new reference maps for a segmented map. It then selects a new reference map that has less direct connections between excluded nodes. The excluded nodes mean the other nodes which remain in a segmented map and a reference map except common nodes. Because excluded nodes in a segmented map are placed out of the communication range of excluded nodes in a reference map, they do not have any direct connection to each other.

For example, a reference map and a segmented map are shown in Fig. 2(a) and 2(b). Their common nodes are nodes 1 and 5, excluded nodes of a reference map are nodes 2, 3, and 4, and an excluded node of a segmented map is node 6. With original and reflected segmented maps, two new reference maps are obtained: one is the new reference map I after merging an original segmented map, and the other is the new reference map II after merging a reflected segmented map, as shown in Fig. 2(c) and 2(d). While excluded node 6 of a segmented map has connections to excluded nodes 2, 3, and 4 in Fig. 2(c), it has no connections to any excluded nodes in Fig. 2(d). Hence, we can properly relocate a segmented map on the coordinates of the reference map by selecting the new reference map II, as shown in Fig. 2(d).

C. Merging All Local Maps for a Global Map After building a local map by merging segmented maps, each node sends the local map information to a powerful node, such as a monitoring center, a sink node or a base station. Among the collected local maps, the powerful node selects the local map which has the largest number of nodes, as a reference map for a global map. Then, the other local maps are merged into the reference map with the same mechanism as the



(a) 200 nodes in a square area

(b) 160 nodes in a C-shape area

Fig. 3. Real deployments with 4 beacons marked as black square nodes: (a) 200 nodes in a square area, (b) 160 nodes in a C-shape area.

one used to merge segmented maps. The reference map becomes a global map after merging all local maps. Then, the powerful node relocates the global map with the given beacon nodes in order to assign all nodes the absolute positions, which can be achieved with Umeyama's method by regarding the given beacon nodes as common nodes of a real map. Here, the scaling parameter in Eq. 2 is used because the global map could be scaled up or down compared to a real map. Finally, we can get a global map where all nodes have their absolute positions. The main advantages of such a centralized algorithm using a powerful node are to alleviate network burden because it is not necessary to forward absolute positions to all nodes, and to improve accuracy in nodes' position using the global information of sensor nodes.

4 Simulation Results

To evaluate the proposed localization algorithm with merging segmented maps (LaMSM), we consider two distinct placements of nodes, as shown in Fig. 3: one is a regular placement where 200 nodes are randomly deployed over a 1000m × 1000m square area, and the other is an irregular placement where 160 nodes are randomly distributed over a C-shape area, where dark square nodes are beacons which are supposed to be attached to buoys. Also, we assume that communication between nodes is possible within acoustic transmission range, $R = 150$m. The measured distance between nodes i and j contains a range error, which is modeled as $d_{ij} = v_a(t_{ij} + t_\epsilon) = d^*_{ij} + v_a t_\epsilon$ where $v_a = 1.5 \times 10^3$m/s is the speed of sound underwater, d^*_{ij} is the real distance between nodes i and j, t_{ij} is actual time for d^*_{ij}, and t_ϵ is a range error which is a random value with the Gaussian distribution $N(0, \sigma^2)$. We ran the simulation 100 times to get the

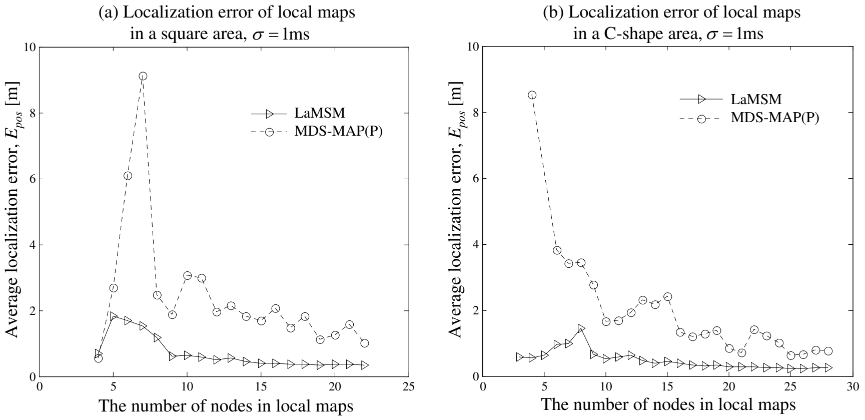


Fig. 4. Localization error in local maps built with LaMSM and MDS-MAP(P) under $\sigma = 1$ms in (a) a square area and (b) a C-shape area, respectively.

results under a different range error each time. To indicate average position error between positions of a real node and its estimated node, the average localization error of a node is defined as

$$E_{pos} = \frac{1}{n}\sum_{i=1}^{n} ||x_i^* - x_i||, \tag{4}$$

where n is the number of nodes, x_i^* and x_i are the real position and the estimated position for a node i, respectively.

First, we compare the average localization error in local maps built with LaMSM and MDS-MAP(P). The number of nodes in a local map depends on node density. The average number of nodes in a local map is 12.3 in a square area and 12.5 in a C-shape area. The simulations of Fig. 4 were carried out under 1ms range error in time of flight (ToF). As shown in Fig. 4, local maps of LaMSM have lower and more regular localization error than those of MDS-MAP(P) because LaMSM never uses unmeasured distances which MDS-MAP(P) does. The average localization error of all local maps in LaMSM is 0.63m in a square area and 0.61m in a C-shape area, while it is 2.26m and 2.12m in MDS-MAP(P), respectively. Additionally, localization error of MDS-MAP(P) gradually declines with the increasing number of nodes in a local map; that is, the number of measured distances relatively increases rather than the number of the calculated distances with increasing node number, so that the effect of errors in the calculated distance is reduced correspondingly.

Fig. 5 shows the average localization error in a global map by varying the standard deviation of a range error from 0.2ms to 2ms. Results show that the accuracy of LaMSM is about 7.8 times better than MDS-MAP(P) in a square area and about 10.4 times better in a C-shape area at 0.2ms range error, as shown in Fig. 5(a) and 5(b), respectively. Even at 2ms range error, the accu-

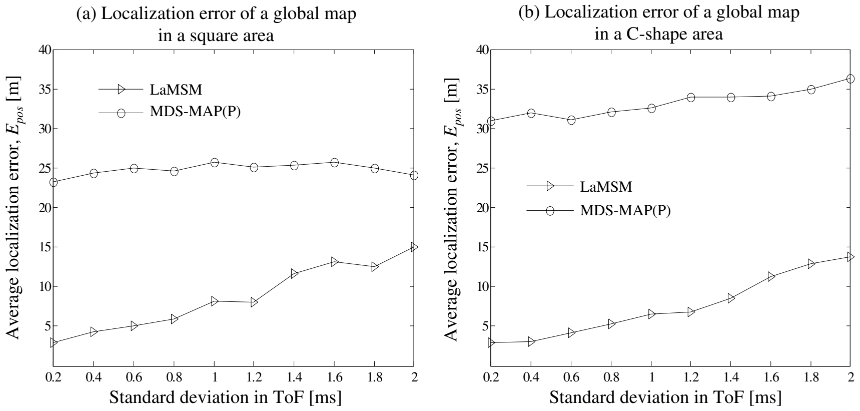


Fig. 5. Average localization errors for all nodes estimated with LaMSM and MDS-MAP(P) in (a) a square area and (b) a C-shape area.

racy of LaMSM is still better than MDS-MAP(P); that is, about 1.6 times in a square area and 2.6 times in a C-shape area. Overall, the cause of relatively high localization error in MDS-MAP(P) is that errors arising from calculating unmeasured distances are dominant over range errors when it constructs local maps. Furthermore, the performance of LaMSM is analogous in both node deployments, while MDS-MAP(P) performs better in a square area than a C-shape area.

5 Conclusions

We presented the localization algorithm with merging segmented maps (LaMSM) for underwater sensor networks (UWSNs). Underwater sensor nodes use acoustic signals to communicate due to its long transmission range, which makes it possible to measure accurately distances between nodes by resolving the delay of acoustic signals.

As a basic unit for a global map in LaMSM, a segmented map is composed of the only fully connected nodes; that is, the map is built with only the measured distances. Hence, it can avoid calculating unmeasured distances that result in errors in accuracy. In fact, LaMSM has about 3.5 times better accuracy in a local map than the well-known MDS-MAP(P). Furthermore, LaMSM provides a closed-form solution to minimize localization errors arising from merging maps, where it utilizes node connections to correctly decide whether a map is reflected or not. According to our simulation results, LaMSM outperforms MDS-MAP(P) with respect to localization accuracy: at least 1.6 times in a square area, and 2.6 times in a C-shape area, even at 2ms range error.

Accurate position information is useful to various ocean applications, such as exploring geographical features, developing natural resources underwater, protecting from ocean disasters, etc. To secure accurate position information, more elaboration is required in resolving the delay of time of flight (ToF), time synchronization, and localization algorithms. In the future our work will be to investigate LaMSM under realistic environment, three-dimensional Euclidean space.

Acknowledgment

This work was supported by the Center for Distributed Sensor Network at GIST.

References

1. Akyildiz, I.F., Pompili, D., Melodia, T.: Underwater Acoustic Sensor Networks: Research Challenges. Ad Hoc Networks Journal, (2005) 257–279
2. Cui, J., Kong, J., Gerla, M., Zhou, S.: The Challenges of Building Scalable Mobile Underwater Wireless Sensor Networks for Aquatic Applications. IEEE Networks Mag., (2006) 12–18
3. Oh, M., Raftery, A.: Bayesian Multidimensional Scaling and Choice of Dimension. Journal of the American Statistical Association, Vol. 96, No. 455, (2001) 1031–1044
4. Shang, Y., Ruml, W., Zhang, Y., Fromherz, M.: Localization from Mere Connectivity. Proc. of the 4th ACM Intr. Symp. on Mobile and Ad-Hoc Networking & Computing, (2003) 201–212
5. Shang, Y., Ruml, W.: Improved MDS-Based Localization. Proc. of IEEE INFOCOM, Vol. 4, (2004) 2640–2651
6. Shang, Y., Ruml, W., Zhang, Y., Fromherz, M.: Localization from Connectivity in Sensor Networks. IEEE Trans. Parallel and Distributed Systems, Vol. 15, No. 11, (2004) 961–974
7. Moore, D., Leonard, J., Rus, D., Teller, S.: Robust Distributed Network Localization with Noisy Range Measurements. Proc. of ACM SenSys'04, (2004) 50–61
8. Kwon, O., Song, H.: A New Map Stitching Method for Anchor-free Localization in Wireless Sensor Networks. IEEE Conf. on Computer and Information Technology, (2006) 236–236
9. Horn, B.K.P., Hilden, H.M., Negahdaripour, S.: Closed-form Solution of Absolute Orientation Using Orthonormal Matrices. Journal of the Optical Society of America A, Vol. 5, No. 7, (1988) 1127–1135
10. Umeyama, S.: Least-Squares Estimation of Transformation Parameters between Two Point Patterns. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 13, No. 4, (1991) 376–380