

Distributed Localization in Wireless Sensor Networks based on Force-Vectors

Viet-Duc Le ¹, Viet-Hung Dang ², Sungyoung Lee ³, Sang-Hoon Lee ⁴

¹ Department of Computer Engineering, KyungHee University

Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, Korea, levieduc@oslab.khu.ac.kr

² Department of Computer Engineering, KyungHee University

Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, Korea, dangviethung@oslab.khu.ac.kr

³ Department of Computer Engineering, KyungHee University

Seocheon-dong, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, Korea, sylee@oslab.khu.ac.kr

⁴ Information Services Division, Korean Institute of Construction Technology

2311, Deahwa-dong, Ilsanseo-gu, Goyang-si, Gyeonggi-do 411-712, Korea, jason813@kict.re.kr

Abstract

Node Localization is a fundamental requirement for deploying real wireless sensor network applications such as hospital logistics, environmental monitoring, and target surveillance. However, recently proposed algorithms still could not attain enough accuracy and are too costly. To improve the previous work, we describe a low cost, high accuracy, scalable, distributed localization algorithm which based on distance vectors. We assume that only few sensor nodes have known-locations (named as beacons) and the remaining nodes have unknown-locations (named as normal nodes). Each node updates its location from currently estimated distance vectors and given pairwise distances which are derived from Received Signal Strength (RSS) measurements or Time of Arrival (TOA), and then passes this new location to neighbors until achieving enough convergence. Analysis, simulation and experimental results show that the proposed algorithm outperforms the other range-based algorithms. Specially the proposed algorithm on real-world experiment measurements, which contain unpredictable noise and shadowing, achieves better accuracy than the previous work do. The proposed method can perform well even with only few reference devices or anchors.

1. INTRODUCTION

Nowadays, there is a growing trend of developing wireless sensor network applications in which nodes localization or self-organization is an essential and crucial issue. Thanks to the near real-time monitoring, low cost and easy deployment, wireless sensor networks (WSNs) may be applied for US water utilities [1], environmental research [2], zebras tracking [6], and active volcano monitoring [4]. To permit deployment of such applications, sensor data must be with its physical location, in other words, "sensing data without knowing the sensor location is meaningless" [5]. Therefore, node localization schemes which have low cost, low energy consumption, high accuracy, robustness are necessary to be developed to meet these applications.

Many localization algorithms have been proposed to fulfill these needs. Such localization algorithms are categorized into range-based schemes and range-free schemes. The range-based schemes estimate the coordinates of nodes from pairwise distances determined by special hardware which can measure time of arrival (TOA), time of difference of arrival (TDOA), angle of arrival (AOA), received signal strength (RSS). Depending on the location accuracy requirements which vary with applications, these types of measurements can be used: TOA and AOA for high accuracy (in order of centimeters); RSS for low accuracy (in order of meters). However, TOA and AOA often add size, cost and energy requirements of devices. If the first priority is low cost, RSS is typically used. To improve the accuracy in case of using RSS, Patwari et al. proposed novel localization algorithms based on maximum likelihood relative estimation (MLE) [10], manifold learning [12]. Recently Costa et al. [11] introduced the scalable, distributed weighted-multidimensional scaling (dwMDS) algorithm.

In the range-free schemes, the locations of sensors are estimated from their connectivity, whether or not two sensors can communicate with each other. Therefore, the range-free schemes do not require that sensor nodes are equipped with such additional hardware to determine the pairwise distances. In this way, it significantly reduces cost, energy consumption, and is suitable for the resource-limited WSNs. These range-free algorithms include Isomap/MDS-MAP [15] [18], the area-based approach APIT [13], the DV based positioning (DV-hop) [14], the Convex Position Estimation [17], and the Mobile and Static sensor network Localization (MSL) [16]. These algorithms estimate the locations of normal nodes (without location information) by gathering the location information from anchor nodes or reference nodes (with known location). Recently, some novel approaches to range-free localization were proposed such as the distributed localization scheme with improved grid-scan and vector-based refinement [21], and the connectivity-based for complex shape networks [19].

Obviously, the range-free schemes take precedence over the range-based schemes where large scale, low cost, low power

consumption requirements are the main concern. However, the current range-free algorithms are either too complex, or not accurate enough and also not scalable. Therefore, the range-free schemes are still unrealistic, particularly in WSNs' applications which have beyond 10,000 nodes. According to a recent survey [3], most of future WSN applications would be applied on small and medium networks with less than 100 nodes. We also realized that the Received Signal Strength Indication (RSSI) are available for most sensors deployed in indoor applications. In addition, the existing range-based algorithms still could not overwhelm the noise of radio frequencies and fading channel in RSS measurements to achieve enough accuracy.

Motivated by these considerations, we propose a light weight and distributed range-based localization scheme which significantly improves the accuracy, removes the bias due to measurement noise and artifacts. As our best knowledge, the proposed algorithm is radically different from the current range-based algorithms. The algorithm estimates the locations of nodes by modeling the error distances into push and pull forces. So we name the algorithm as Push-Pull Estimation (PPE). More details of the PPE algorithm will be analytically described in Section 3. In section 4, by using database of a real experiments [7], we show that the PPE algorithm performs well even when the network has irregular noise and fading channels due to the presence of walls, corridors, furniture, or other obstacles in the deployment area. The experimental results show that the proposed algorithm can give better accuracy than existing algorithms do.

2. PROBLEM STATEMENT

We consider a small to medium network of n sensors which are randomly deployed in d -dimensional space ($d=2$ or 3) and have unknown-location information. To obtain the absolute coordinates, very few m anchors, which have known-location information, need to add in ($m \ll n$). Let $N=n+m$ be the total number of sensors in the considering WSN, $X = \{x_i : i = 1..N\}$, $x_i \in \mathbb{R}^d$, be the actual vector coordinates of sensors and $\tilde{X} = \{\tilde{x}_i : i = 1..N\}$, $\tilde{x}_i \in \mathbb{R}^d$, be the estimate vector coordinates of sensors. The pairwise range measurements, $\{\delta_{ij} : i, j = 1..N\}$, are given or measured by specific hardware, for example, RSSI. The localization problem is the estimation of the vector coordinates $\{\tilde{x}_i : i = 1..n\}$ given the vector coordinates of the anchor nodes, $\{x_i : i = n+1..N\}$, and measured pairwise vector distances, $\{\delta_{ij} : i \neq j \text{ and } i, j = 1..N\}$. We assume that all measured pairwise vector distances, $\{\delta_{ij} : i \neq j \text{ and } i, j = 1..N\}$, are available and $\|\delta_{ij}\| = \|\delta_{ji}\|$, ($\|\cdot\|$ is the 2-norm). This assumption doesn't restrict the application of the proposed algorithms. The more pairwise vector distances are given, the higher accuracy is achieved. Note that our method is developed to adapt any type of range measurements, for instance, RSS, TOA, or AOA. However, in this paper, we aim at RSS measurement because of its low cost and also for its availability for small and medium WSN applications.

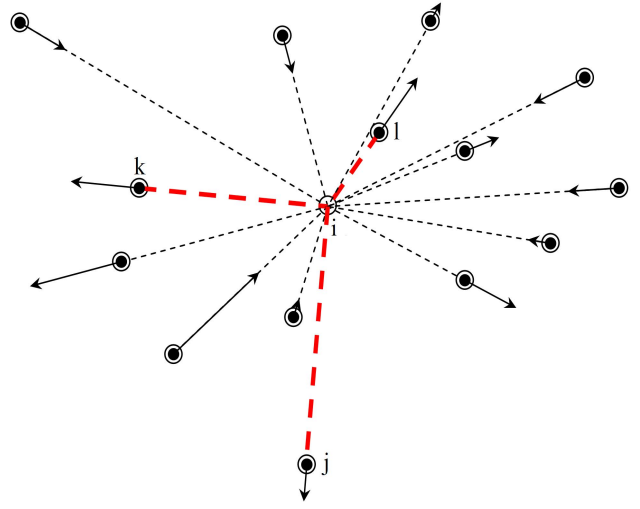


Fig. 1: Errors between the given and estimate distances respected to node i are modeled as vector errors

3. PUSH-PULL ESTIMATION

The original of the method comes from the thought that there must be some way to remove noise of measurements by averaging even this kind of noise depends on the real peer-to-peer distances and directions. The method we found, based on geometry, estimates the coordinates of sensors by modeling the noisy measurements into push and pull vectors as force-vectors. We call the method Push-Pull Estimation (PPE) as its fundamental concept.

A. Model and Analysis

Assuming that the noise we consider is Gaussian noise, we will step by step analyze the PPE algorithm, hereby. Without loss of generality, let's consider an arbitrary node i in a wireless sensor network. If the coordinates of all other nodes are indicated, it is easy to calculate the coordinate of node i from three neighboring nodes that form a triangle, say j , k , and l (Fig. 1), and the true distances between i and i , k , l . However, noises which always present in measurements make the pairwise distances measured either longer or shorter than the real ones, respectively. Thus, we surely can not gain a good estimation for location of node i unless we use the location information of more than three neighboring nodes. Obviously, if we use all the pairwise distances measured from node i , we would get the best location estimation.

Let $\{d_{ij} : i \neq j \text{ and } j = 1..N\}$ and $\{\delta_{ij} : i \neq j \text{ and } j = 1..N\}$ be the actual and measured pairwise vector distances of which starting points are at node i , respectively. Then for an arbitrary node i , we have a set of vector errors that models the error of measurements in both the distance and the direction is:

$$\Delta_{ij} = \delta_{ij} - d_{ij}, \forall i \neq j, j = 1..N. \quad (1)$$

Since the magnitudes of some pairwise vector distances could be several orders of that of the other pairwise vector distances, it is critical to normalize the errors to the magnitudes

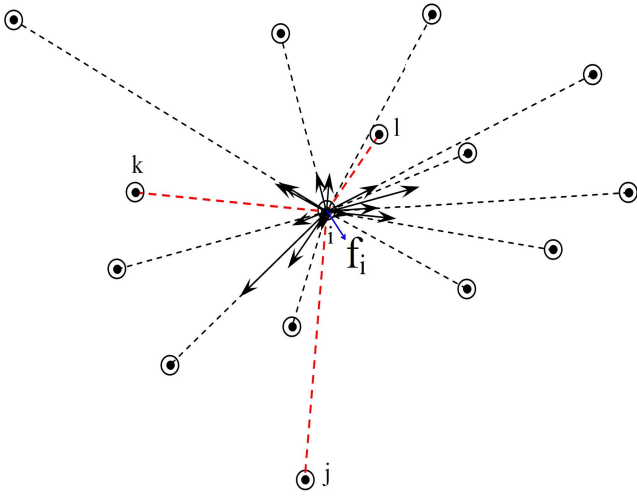


Fig. 2: A sum of the vector errors affecting node i forms a push-pull vector f_i of which beginning point is at i

of measured pairwise vector distances. We propose a method of normalization by dividing the errors by the magnitudes of actual pairwise vector distances, respectively. Let $\{\hat{\Delta}_{ij} : i \neq j \text{ and } j = 1..N\}$ denote these normalized vector errors, we have:

$$\hat{\Delta}_{ij} = \Delta_{ij} \frac{\|\delta_{ij} - d_{ij}\|}{\|d_{ij}\|}, \quad \forall i \neq j, j = 1..N \quad (2)$$

where $\|\cdot\|$ is the 2-norm (Euclidean norm).

if $\hat{\Delta}_{ij} > 0$ then it forms a push vector, and if $\hat{\Delta}_{ij} < 0$ then it forms a pull vector. The sum of these push and pull vectors, f_i , will describe the sum of measurement errors that affect the estimation of node i ,

$$f_i = \sum_{i \neq j, j=1}^N \hat{\Delta}_{ij} = \sum_{i \neq j, j=1}^N \Delta_{ij} \frac{\|\delta_{ij} - d_{ij}\|}{\|d_{ij}\|}. \quad (3)$$

Consequently this sum of push-pull vectors, f_i , will drag the sensor i to a new location (see Fig. 2).

Now let's consider the functions of these push and pull vectors in a situation where the pairwise range measurements include bias of noise and the locations of most sensors are unknown. What we have is not the the actual pairwise vector distances $\{d_{ij} : i \neq j \text{ and } j = 1..N\}$, but the estimate $\{\tilde{d}_{ij} : i \neq j \text{ and } j = 1..N\}$. Where

$$\tilde{d}_{ij} = \tilde{x}_j - \tilde{x}_i, \quad \forall j \neq i, j = 1..N. \quad (4)$$

From (2) we have the push-pull vectors,

$$\hat{\Delta}_{ij} = \Delta_{ij} \frac{\|\delta_{ij} - \tilde{d}_{ij}\|}{\|\tilde{d}_{ij}\|}, \quad \forall i \neq j, j = 1..N \quad (5)$$

from (3) and (4) we have the sum of push-pull vectors,

$$f_i = \sum_{i \neq j, j=1}^N \hat{\Delta}_{ij} = \sum_{i \neq j, j=1}^N \Delta_{ij} \frac{\|\delta_{ij} + \tilde{x}_i - \tilde{x}_j\|}{\|\tilde{x}_i - \tilde{x}_j\|}. \quad (6)$$

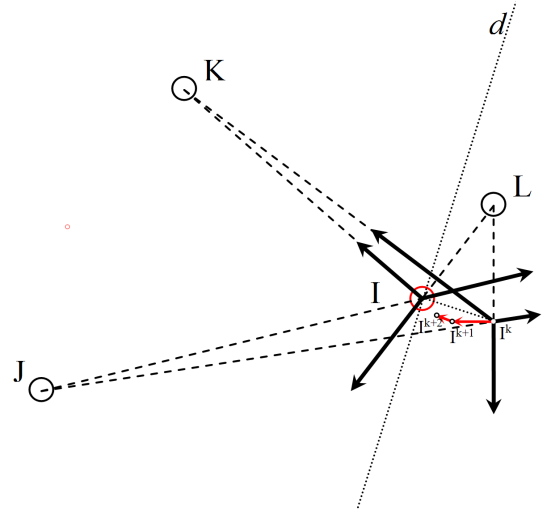


Fig. 3: An arbitrary node i moves along a curve line to its optimal location, iteration by iteration.

Clearly, for any node in the network, the sum of all push vectors is not equal to that of pull vectors ($f_i \neq 0$). In other words, these push-pull vector forces will make all nodes gradually move to where the state of every node is most likely in balance. The lesser norm $\|f_i\|$, the better localization estimation. The problem of localization now becomes the minimization of f_i .

$$\tilde{x}_i = \operatorname{argmin}_{\tilde{x}_i} f_i(\tilde{x}_i) \quad (7)$$

or

$$\tilde{x}_i = \operatorname{argmin}_{\tilde{x}_i} \sum_{i \neq j, j=1}^N \Delta_{ij} \frac{\|\delta_{ij} + \tilde{x}_i - \tilde{x}_j\|}{\|\tilde{x}_i - \tilde{x}_j\|}. \quad (8)$$

To examine that whether the sum of push-pull vectors of node i , f_i will drag node i to its optimal location, without loss of generality, we study a case of an arbitrary node i and three of its arbitrary neighbors, for instance, j , k , and l in Fig. 3. Assume that nodes j , k , and l are still and the sum of their push-pull vectors on node i is zero ($f_i = 0$) at location I (the optimal location). Let's begin with a current estimate location of node i at I^k . Draw a straight line (d) , $(d) \perp II^k$, which divides the plane into two sides. Nodes j and k on the opposite side of I , respectively, tend to generate a weaker push-force, $\|\hat{\Delta}_{ij}^k\| < \|\hat{\Delta}_{ij}\|$, and a stronger pull-force, $\|\hat{\Delta}_{ik}^k\| > \|\hat{\Delta}_{ik}\|$. Meanwhile, node l on the same side of I generates a stronger push-force, $\|\hat{\Delta}_{il}^k\| > \|\hat{\Delta}_{il}\|$. The push-pull vector applying on node i at location I^k will be

$$f_i^k = \hat{\Delta}_{ij}^k + \hat{\Delta}_{ik}^k + \hat{\Delta}_{il}^k.$$

Hence, under the impact of this push-pull force, the estimate location of node i will directly move from I^k to a new position I^{k+1} . Similarly we obtain the next estimate location I^{k+1} and so on, until it gets to the optimal location I (see Fig. 3).

We remark that the estimate location of node i will not run along the line II^k , but on a curve. The reason is that the

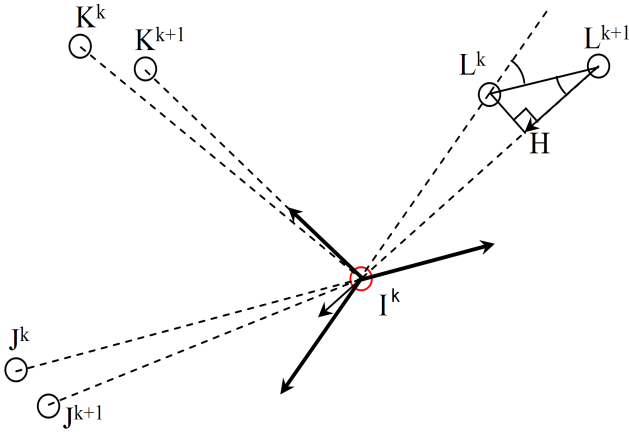


Fig. 4: The impact of node l on node i when node l moves to its next update location.

sum of push-pull vectors, f_i^k , changes in both magnitude and direction with each iteration k .

In addition, our goal is to develop a distributed localization algorithm allowing every node in the wireless sensor network to run independently and simultaneously. Therefore, what we have studied above with the assumption that neighbors of the considering node are still (do not update their own location) is just the necessary condition of the problem. Let's move to the sufficient condition, that is, the location updating of any node in WSN is not affected by that of other nodes. We realize that node i mentioned above (Fig. 3) still move towards its optimal location, I , even when nodes j , k , and l change their locations.

Let's prove the sufficient condition by assuming that node i is estimating its own location at iteration k , meanwhile, its neighbor nodes already updated their own locations at iteration $k+1$ (see Fig. 4). We may without loss of generality, examine node l where $L^k L^{k+1} = \vec{d}$. Because most nodes in the network can be considered to be far enough from node i , $IL^k \approx IH$ if $L^k H \perp IH$. The new push-pull vector, which node j applies on node i , will change in both direction and reduce magnitude with a value proportional to $L^{k+1}H = d \cos \alpha$. The new push-pull vectors of other nodes apply on node i are defined analogously. In addition, as the bias of estimate locations is caused by Gaussian noise and random step-angles, these adding vectors, for instance, $L^{k+1}H$, have Gaussian distribution,

$$d \cos \alpha \sim N(d_0, \sigma), \quad \alpha \sim U(-\pi, \pi). \quad (9)$$

Therefore, the push-pull vector on node i only changes the direction and the magnitude a bit. In other words, node i still keeps moving to its optimal location iteration by iteration. Consequently, the optimal locations of all sensors will pervade over the whole network and the distributed localization algorithm gradually converges.

B. Algorithm

The Push and Pull Estimation (PPE) algorithm proposed hereby is a distributed localization algorithm. In other words, this method allows for parallel and distributed implementations on every sensor in wireless sensor networks.

Algorithm 1 Push and Pull Estimation (PPE)

Inputs $k, n, \alpha, \varepsilon, \{\delta_{ij} : i \neq j \text{ and } i, j = 1..N\}$

Initialize $k = 0, N = n + m, \tilde{x}_i^0$.

send \tilde{x}_i^0 to neighbors of node i

receive $\tilde{x}_j^0, j \neq i, j = 1..N$ from neighbors of node i

form \tilde{X}^0

repeat

$k \leftarrow k + 1$

for $j = 1$ to N **do**

if $j \neq i$ **then**

$$\tilde{d}_{ij}^k = \tilde{x}_j^k - \tilde{x}_i^k$$

$$\Delta_{ij}^k = \delta_{ij} - \tilde{d}_{ij}^k$$

$$f_i^k \leftarrow f_i^k + \Delta_{ij}^k \frac{\|\delta_{ij} - \tilde{d}_{ij}^k\|}{\|\tilde{d}_{ij}^k\|}$$

end if

end for

$$\alpha = \frac{1}{N}$$

$$\tilde{x}_i^k \leftarrow \tilde{x}_i^{k-1} + \alpha \cdot f_i^k$$

send \tilde{x}_i^k to neighbors of node i

receive \tilde{x}_j^k from neighbors of node i

update \tilde{X}^k

until $\|\tilde{x}_i^k - \tilde{x}_i^{k-1}\| < \varepsilon$

To start running the algorithm, it is necessary to generate "raw" initial vector coordinates \tilde{X}^0 . This step can be done by any other localization estimation method such as classical MDS [20]. However, we recommend using the PPE algorithm with k -nearest beacons ($k \geq 3$) to initialize a raw vector coordinate for each node which is then passed along the network until \tilde{X}^0 is completely built. We realized that this method was robust and had much less computations compared with other raw coordinate initializations.

It was assumed that all measured pairwise distances are available. However, it is possible to gain the estimate coordinates even if some measured pairwise distances are not given.

In the description of the algorithm, it was assumed for notational convenience, that a node i must update the entirely estimate coordinates \tilde{X}^k before starting a new iteration step ($k+1$). Nevertheless, a new iteration of node i can start no matter how many estimate coordinates of neighbors of node i are updated. This ensures that every sensor in the network can run independently and parallel.

4. EXPERIMENTAL RESULTS

The most challenging aspect of the localization problem in a real Wireless Sensor Network is the bias caused by effects of shadowing: irregularly distributed power of signals, occlusion, collisions, and multipath effects [22] [8]. These factors create fading channels that dramatically affect on the accuracy.

TABLE 1: THE RMSE OF LOCALIZATION ESTIMATES IN THE EXPERIMENTAL WIRELESS SENSOR NETWORK

	Classical MDS	MLE	dwMDS	PPE
RSS	4.26m	2.18m	2.48m	2.44m
TOA	1.85m	1.23m	1.12m	1.10m

Therefore, we use the real-world RSS and TOA measurements freely available at Patwari's website [7] to evaluate the performance of the proposed algorithm, PPE. In the experiment, there are 44 wireless sensors deployed in an area of 14 by 13 square meters which are partitioned by cubicle walls 1.8 meters in height. For comparable convenience, we also chose the same four sensor devices near the corners as reference devices or anchor nodes, that is, devices marked as 3, 10, 35, and 44 in the Fig. 5 and Fig. 6. The remaining 40 devices are normal devices with unknown location information. The real database provides the input of above algorithms, four anchor coordinates and the pairwise distances computed from either RSS or TOA measurements. The RSS-based pairwise distance was estimated by bias-corrected MLE from the average of 10 measured RSS for each pair of devices. The TOA-based pairwise distance was computed from the average of 10 measured time delays, 5 with the transmitter at i and receiver at j , and 5 with the transmitter at j and receiver at i . See Patwari et al. [10] for more details. We remark that the RSS-based pairwise distances measured are often very larger than their actual distances, several are even more than 200% due to fading channels.

In the simulation, we use the PPE method for each node with anchors to initialize the rough coordinates of normal sensors. In such way, we can utilize the same modules of software and a real implementation is more practical. To visualize accuracy of the proposed algorithm, we draw lines connecting the estimate and actual coordinates of sensors respectively. The Root Mean Square Error (RMSE) was used to compute the error between estimate and actual coordinates for all devices. The experimental results are shown in Fig. 5 for RSS and in Fig. 6 for TOA. The actual and estimated locations of sensors are marked by 'o' and '▽', respectively, while the anchors are marked by '◇' in red. The convergence of the PPE algorithm is plotted in Fig. 7. The RMSE for RSS-based measurements is 2.4401 m after 75 iterations, and that for TOA-based measurements is 1.0876 m after 5 iterations. Naturally, the RMSE of TOA is more accuracy than that of RSS because TOA gives measured pairwise distances that is more similar to actual distances. In addition, we compare the accuracy of PPE to that of currently range-based algorithms, including the classical MDS, MLE [10], and dwMDS [11]. Table 1 shows that the results of PPE are as good as those of the most recent and novel work, dwMDS. Note that we refer the results of MLE and dwMDS in [11].

We also emphasize that PPE has a poor performance at the edges of the network where there are fewer nearby neighbors as the previous work such as dwMDS [11] or MLE [10] do. The devices close to the center are located more accurately than the devices on the edges, particularly in the RSS case,

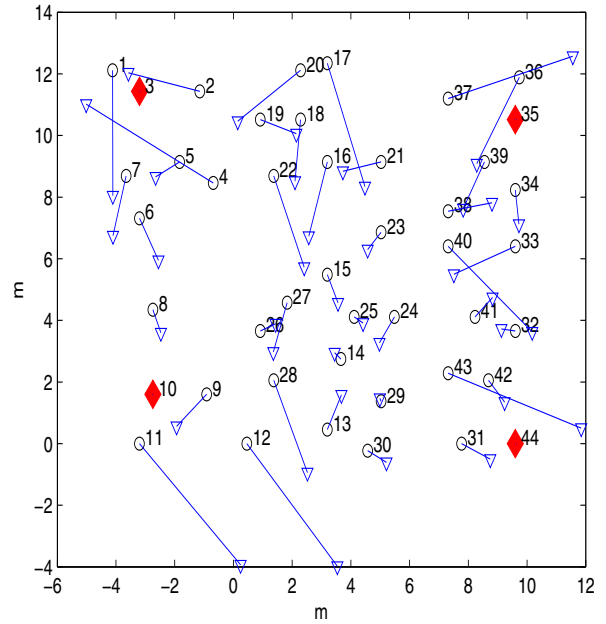


Fig. 5: Actual and estimated locations with PPE algorithm for RSS case, RMSE = 2.4381m. Actual and estimated locations are marked by 'o' and '▽' respectively. Anchors are marked by '◇'.

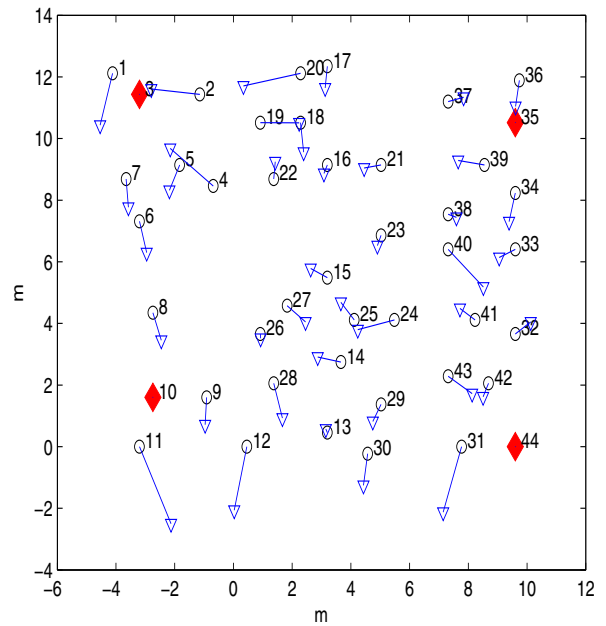


Fig. 6: Actual and estimated locations with PPE algorithm for TOA case, RMSE = 1.1028m. Actual and estimated locations are marked by 'o' and '▽' respectively. Anchors are marked by '◇'.

even those devices nearby anchors. However, the proposed algorithm has lower complexity and it is easier to implement in real-world WSN' applications.

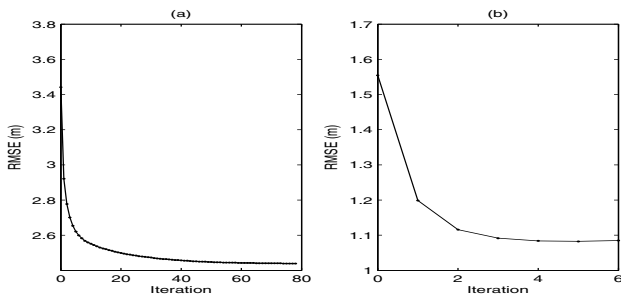


Fig. 7: Convergence of the PPE algorithm applied on: (a)RSS, (b)TOA

5. CONCLUSION

Overwhelming the noise and shadowing in real WSN's applications is the essential challenge for localization algorithms. To reduce the effects of such elements, this paper proposes a distributed range-based localization algorithm based on push and pull force-vectors. In the proposed method, so-called the Push and Pull Estimation (PPE), each normal device (with unknown-location) collects the estimate distances from it to its neighbor devices and then subtracts the given distances from those estimate ones to form a set of vector errors. Under the "force" generated by these vector errors, the estimating node will gradually move towards its own optimal location, where the "force" is minimized. By doing that, every normal node estimates its own coordinates independently and parallel. Analyzing the algorithm shows that the proposed algorithm is fully distributed, lightweight and efficient. Furthermore, we also implement PPE with measured database of a real-world sensor network that is freely available at [7]. The experimental results show that the proposed algorithm can achieve better accuracy than the other localization algorithms, particularly when there are few anchors and a lot of irregular noises. Therefore, PPE is the most suitable among other localization algorithms for real-world wireless sensor networks where the cost of implementation and devices are the priority.

Future research must also investigate and analyze the PPE model and algorithm with statistically simulated models at different constraints such as the number of anchors, the density of networks, the ratio of noise, in order to improve the PPE algorithm. Other implementation and developments with our own finishing testbed can also be used to improve the algorithm's performance.

ACKNOWLEDGMENTS

This research was supported by the MKE (Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2008-C1090-0801-0002). Also, this work was supported by the Korea Research Foundation Grant Funded by the Korean Government (MOEHRD) (KRF-2007-313-D00663).

REFERENCES

- [1] I. Stoianov, L. Nachman, S. Madden, "PIPELINE: a wireless sensor network for pipeline monitoring", In: *Proc. Information Processing in Sensor Networks*, 2007, pp. 264-273.
- [2] L. Selavo, A. Wood, Q. Cao, A. Srinivasan, Y. Wu, W. Kang, J. Stankovic, D. Young, J. Porter, "LUSTER: wireless sensor network for environmental research", In: *Proc. Embedded Network for Sensor Systems*, Sydney, 2007, pp. 103-116.
- [3] BB Electronics and Sensicats. August 2005 wireless survey results, <http://www.bb-elec.com/wirelessurvey/>.
- [4] G. Werner-Allen, K. Lorincz, M. Welsh, O. Marcillo, J. Johnson, M. Ruiz, J. Lees "Deploying a wireless sensor network on an active volcano", *IEEE Internet Computing*, Vol. 10, No. NN, 2006, pp. 18-25.
- [5] J. Rabaey, M. Ammer, J. da Silva, Jr, D. Patel, S. Roundy "Picoradio supports ad hoc ultra-low power wireless networking", *IEEE Comput.*, Vol. 33, 2000, pp. 42-48.
- [6] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, D. Rubenstein, "Energy efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet", In: *Proc. Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2002, pp. 96-107.
- [7] Neal Patwari, *Wireless Sensor Network Localization Measurement Repository*, <http://www.eecs.umich.edu/~hero/localize/>.
- [8] J. Bachrach, C. Taylor *Localization in sensor networks*, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139.
- [9] N. Patwari, J. Ash, S. Kyperountas, A. Hero III, R. Moses, N. Correal "Locating the nodes: cooperative localization in wireless sensor networks", *Signal Processing Magazine, IEEE*, Vol. 22, No. 4, 2005, pp. 54-69.
- [10] N. Patwari, A. Hero III, M. Perkins, N. Correal, R. O'Dea, "Relative Localization Estimation in Wireless Sensor Networks", *IEEE Trans. Signal Processing*, Vol. 51, No. 8, 2003, pp. 2137-2148.
- [11] J. Costa, N. Patwari, A. Hero III, "Distributed weighted-multidimensional scaling for node localization in sensor networks", *ACM Transactions on Sensor Networks*, Vol. 2, No. 1, 2006, pp. 39-64.
- [12] N. Patwari, A. Hero III, "Adaptive neighborhoods for manifold learning-based sensor localization", In: *Proc. IEEE International Conference on Acoustic Speech and Signal Processing*, 2004, pp. 1098-1102.
- [13] T. He, C. Huang, B. Blum, J. Stankovic, T. Abdelzher, "Range-free localization schemes for large scale sensor networks", In: *Proc. ACM MobiCom*, 2003, pp. 81-95.
- [14] D. Niculescu, B. Nath, "DV based positioning in ad hoc networks", *Telecommunication Systems*, Vol. 22, 2003, pp. 267-280.
- [15] Y. Shang, W. Ruml, Y. Zhang, M. Fromherz, "Localization from mere connectivity" In: *Proc. Mobihoc*, 2003, pp. 201-212.
- [16] M. Rudafshani, S. Datta, "Localization in wireless sensor networks", In: *Proc. Information Processing in Sensor Networks (IPSN)*, 2007, pp. 51-60.
- [17] L. Doherty, K. Pister, L. Ghaoui, "Convex position estimation in wireless sensor networks", In: *Proc. IEEE INFOCOM*, 2001, pp. 1655-1663.
- [18] Y. Shang Niculescu, W. Ruml, Y. Zhang, M. Fromherz, "Localization from connectivity in sensor networks", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 15, No. 11, 2004, pp. 961-974.
- [19] S. Lederer, Y. Wang, J. Gao, "Connectivity-based localization of large scale sensor networks with complex shape", In: *Proc. IEEE INFOCOM*, 2008, pp. 789-797.
- [20] V. Vivekanandan, V. Wong, "Ordinal MDS-based localization for wireless sensor networks", In: *Proc. IEEE Vehicular Technology Conference*, 2006, pp. 1-5.
- [21] Jang-Ping Sheu, Pei-Chun Chen, Chih-Shun Hsu, "A distributed localization scheme for wireless sensor networks with improved grid-scan and vector-based refinement", *IEEE Transactions on Mobile Computing*, Vol. 7, No. 9, 2008, pp. 1110-1123.
- [22] N. Patwari, P. Agrawal "Effects of Correlated Shadowing: Connectivity, Localization, and RF Tomography", In: *Proc. Information Processing in Sensor Networks, IPSN*, 2008, pp. 82-93.