Fuzzy Centroid Localization Schemes for Unbalanced Deployments of Wireless Sensor Networks

Songyut Permpol, Chudapa Thammasakorn, Kanokmon Rujirakul and Chakchai So-In
Applied Network Technology (ANT) Department of Computer Science, Faculty of Science, Khon Kaen University
songyut_p@kkumail.com

***Abstract*—This paper proposes a novel methodology to mitigate the effect of unbalanced known nodes' positions for location approximation in wireless sensor networks. In a practical deployment, some nodes may not properly be in uniform places, and perhaps, due to unequal power consumption of large-scale networks while performing sensing, computing, and transmitting tasks. K-means clustering is applied to select a representative of the known nodes where their positions are close together, and each of which will be then fed into fuzzy logic systems to determine a proper weight to finally use in the actual location determination process with weighted Centroid. The effectiveness of our methodology is evaluated via a large scale simulation with regard to node density, coverage, and topology, against a traditional Centroid, its fuzzy systems, and DV-Hop.**

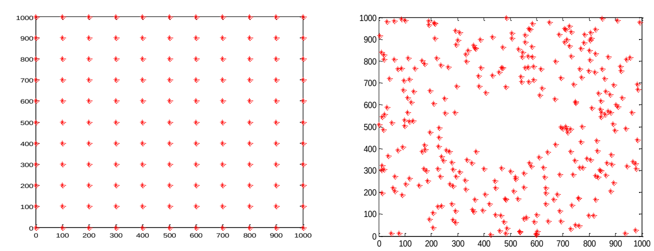
***Keywords*—Centroid; Clustering; Deployment; Fuzzy Logic; K-Means; Non-Uniform; Unbalance; Wireless Sensor Networks.**

I. INTRODUCTION

Recently, Internet of Things (IoT) has become widely used and implemented in many areas, such as smart home, smart city, and even smart grid. Many applications include monitoring, tracking, tracing, and controlling, with respect to smart and automatic schemes [1-2]. The feasible of this IoT is caused by the advance in micro-electro-mechanical systems with regard to the use of tiny sensors equipped with computing, storage, and transmission units, integrating with dedicated power. This sensor can perform multi-functions and interact with each other to form the sensor networks, and wireless sensor networks (WSNs) while considering the communication wirelessly [3-4].

For decades, there have been researches for enhancing WSNs' usages, especially with awareness of power constraint. Several issues have been brought, such as reliability, scalability, routing, and quality of services, one of its challenge includes localization, especially with the absent of GPS signals [4-5], or if equipped, additional costs in terms of hardware logic, size, and budget, are put in the design consideration.

A range-free based localization scheme is promising with its key advantage on low cost [6-7] but with (high) location estimation error trade-off, some of which include Centroid, Approximate Point-In-Triangulation Test (APIT), Distance Vector Hop (DV-Hop), and Amorphous [8]. Generally, the first one, Centroid [9], has been used with its key benefit on simplicity while using a triangulation method over the information of known nodes' positions (anchor). Similarly, however, together with center of gravity (CoG), APIT can consume more energy even with precision gain [10].



(a) Grid Distribution

(b) Non-Uniform Distribution (five holes)

Figure 1: Node Distribution Deployment (cross = anchor nodes)

Instead of the only position information, other approaches can also use various information like number of hops (DV-Hop and Amorphous). DV-Hop can yield better estimation precision in low density deployments of anchor nodes [11]. The estimation can be derived from the average hop size over the number of hops. Amorphous only considers the nodes with less hop size to reduce the complexity [12]. Note that both could present high estimation error in high density of the reference nodes.

For years, there are several approaches proposed to enhance the location estimation performance; however, one of the pioneers, i.e., Centroid, is commonly used for precision improvement, especially with an additional weight. One of its derivations is based on fuzzy logic systems (FLS) because it can yield higher accuracy without high computational complexity tradeoff [13].

Although there exists some optimized techniques to deal with this adjustable weight, most proposals do not consider the diversity of topology where it is norm for practical deployments. Some of the examples include the deployed area where is close to the border or where the position of anchor nodes is unbalance due to a specific characteristic of the actual environment or even with the unequal power consumption of nodes leading to a non-uniform distribution of dead nodes. Figure 1 shows examples of such scenarios, i.e., grid and non-uniform deployments.

Thus, this research investigates the behavior of unbalanced structure of anchor nodes, especially when more numbers of anchor nodes are close to each other; but in the actual computation, these nodes can deviate the estimation precision on unknown node position. We then apply K-means as a clustering technique to determine one such representative of these nodes so as to be used further for weight calculation. Due to the simplicity suitable for distributed sensor networks, FLS is also our selection to generate a proper weight given a proper set of anchor nodes before applying the weight into the approximation process of weighted Centroid.

This paper is organized as follows: Section 2 provides a literature survey of wireless sensor network localizations, especially with respect to Centroid and its weight derived from FLSs. Then, Sections 3 presents our methodology including a detailed description. In Section 4, the performance evaluation of our proposal is then evaluated against some existing techniques, such as Centroid, its fuzzy weights, and DV-Hop. Finally, Section 5 contains our conclusion and possible future work.

II. RELATED WORK

As briefly discussed, for decades, there are several techniques to enhance WSN localization approximation [6-8]. Again, Centroid is one of the pioneers with its key advantage on simplicity but with (high) location estimation error trade-off [13]. Several techniques have been proposed to enhance its performance, especially with weight adjustments. One of which, recently, is based on soft-computing [14], such as Neural Networks (NNs), Fuzzy Logic (FL), and Evolutional Computing (EC), Support Vector Machine (SVM) [14]. However, here, we focus on FL due to its key advantage on simplicity – low computational complexity [14].

In 2005, Yun, S., *et al.* [13] investigated how to adjust the weight of Centroid given received signal strength indication (RSSI) as inputs using Takagi-Sugeno-Kang FL; however, genetic algorithm, one traditional class of the EC, was also used to adjust the shape of membership function to explicitly differentiate the inputs but with high computational complexity trade-off.

In addition, in 2009, the same research group [15] evaluated the estimation performance in comparisons between FL and NN, and then reported that although NN's precision is better, the computational time is high, perhaps, not properly used for the sensor node with limited power as constraint. The evaluation is at the centralized base station.

Two year later, Kumar, V. *et al.* [16] performed the performance comparison of FL types, i.e., between Mamdani and Sugeno; and with their average - there was an improvement of the precision but with computational complexity trade-off. The use of each type is typically based on the complexity of the problem. For example, Sugeno is suitable for mathematical optimization but if simplified problem for Mamdani [17].

In addition, in 2012, Larios, D.F. et al. [18] investigated various inputs including the RSSIs from each anchor nodes' coverage as another computational step. There are two fuzzy logic classes (2 input groups). The first class consists of three membership functions (for RSSIs) and the other is nine functions for each anchor node within its coverage. With this additional step induces higher computational complexity.

From all techniques, FL derivations, considered only grid deployments which these may require more investigation in practical field like non-uniform deployments or unbalanced structure of node distribution. Thus, considering the probable approach for meeting this requirement, Shang, Y. *et al.* [19] investigated a mathematical model using cosine and linear algebra to approximate the unknown node position for random deployment, called Multidimensional Scaling; however, the complexity of this algorithm is very high, not suitable for distributed computing for WSNs.

The other promising type of soft-computing is based on SVM for WSN localization. Tran, D. A. *et al.* [20] applied a binary SVM to segment the deployment area into sub-area. Then, the unknown node position will be determined in which sub-area will be covered based on the number of hops. The authors reported that the algorithm is robust to the irregularity of the topology; however, the computation is centralized at the base station, not practical for distributed sensor networks.

III. FUZZY CENTROID OPTIMIZATIONS FOR WIRELESS SENSOR NETWORK LOCALIZATIONS

There are four main steps of our protocol optimization as follows:

1. Broadcasting: given the unknown node coverage, this node listens for broadcasting beacons from anchor nodes. Typically, the beacon is periodic and contains the position of the anchor node including its RSSI.
2. Clustering: once there is a set of known position of different anchor nodes corresponding to the signal strength, the unknown node will perform K-means clustering such that the representative of each cluster will be selected based on its RSSIs. In this research, K is 3 (maximum) so as to have enough nodes for location approximation.
3. Fuzzy Centroid: based on the selected RSSIs among each group, the unknown node will then compute the weight to adjust the Centroid.
4. Location Estimation: at this step, the node will actually perform the location approximation resulting into the predicted location in (x, y) coordinates.

A. K-means Clustering

One of the pioneer clustering methods is K-means with its key advantage of its simplicity [21]. This technique can be considered as a partition method which makes advantage of the data averaging into the same group or cluster. This average can be used as a representative of each cluster. There are three main steps of K-means as follows:

1. Base on a predefined K as the maximum number of groups, it starts from randomly select the data K sets, each of which will be initially used as a center of each cluster.
2. Then, subsequently, more data sets will be fed into the computation process based on the similarity or the distance to the center.

3. With more data sets, the averaging will be recomputed until no more data left and the data within the network has not changed to different clusters.

Suppose there is a set of $X = (x_1, x_2, \ldots, x_n)$, each of which represents the vector (in real values) in d dimensions. To cluster the data in K-means, the set of X will be grouped into K clusters such that $K \leq n$. The member of each cluster denotes $S_i \mid 1 \leq i \leq K$. The selection criterion is based on the minimization within the cluster sum of squares as stated in equation below. Here, μ_i is the average of each point within the same group.

$$\min_{S} \sum_{i=1}^{k} \sum_{x \in S_i} \|x - \mu_i\|^2 \tag{1}$$

Once applying K-means to cluster the anchor nodes, there are three main steps as follows. Note that after the completion of clustering process, the representative (anchor node) of each group will be selected based on the maximum signal strength, i.e., perhaps, this node will be close to the particular unknown node.

1. Given a predefined K clusters, here is three because it is probably required to have the predicted location within the triangular shape of the anchor nodes.
2. The unknown node randomly selects K (3) anchor nodes as the cluster centers.
3. Subsequently, each known position of each anchor node will be arranged to the group which has the center close to the node using the minimum distance.
4. The center of K positions will be recomputed based on the averaging of all locations of the anchor nodes within the same group.
5. All three steps will be repeated until no change in the center.

It should be noted that we also run various simulations - varying the values of K. While ranging the number of nodes and coverages, K=2 is the worst; with K larger than 4 is also worse. With high density of nodes, although K=4 is higher in accuracy, the precision improvement is not significant compared to K=3 but with higher complexity.

B. *Fuzzy Weight Derivation*

After RSSIs are ready selected from the proper anchor nodes, before performing the actual location approximation using Centroid computation, this research also apply an additional weight derived from FLS [17]. Those represented RSSIs will be used as fuzzy inputs to derive the output weights in range of 0 to 1, and finally generate the output from the fuzzy process. Note that the input RSSIs are also normalized in range between -1 and 1 before feeding into the computational process.

In this research, we used the fuzzy inference systems with Sugeno since it is suitable for optimization and mathematical analysis [17]. Moreover, the selection of membership function is based on our intensive evaluation; that is Triangular function [22]. With the recommendation provided by Yun, S. *et al.* [15], there are five membership functions, i.e., Very low, Low, Medium, High and Very high, as inputs (See Figure 2) and five rules as shown in Table 1.

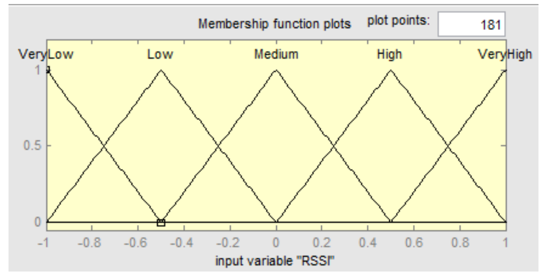


Figure 2: Fuzzy Logic - Triangular Function of Input (RSSI)

Table 1
Fuzzy Rule: Example

Rule	IF: State of RSSI	THEN: State of Weight
1	Very low	Very low
2	Low	Low
3	Medium	Medium
4	High	High
5	Very high	Very high

C. *Fuzzy Weighted Centroid*

With the weights (aka output as fuzzy definition) derived from the fuzzy process, the actual node approximation will be computed over the weighted Centroid, as shown in equation (2). Here, the anchor nodes (x_i, y_i) with corresponding weights (w_i) are used to derive the estimated location (x_{est}, y_{est}) [15].

$$(x_{est}, y_{est}) = \left(\frac{\sum_{i=1}^{n} w_i \times x_i}{\sum_{i=1}^{n} w_i}, \frac{\sum_{i=1}^{n} w_i \times y_i}{\sum_{i=1}^{n} w_i} \right) \tag{2}$$

D. *Performance Evaluation*

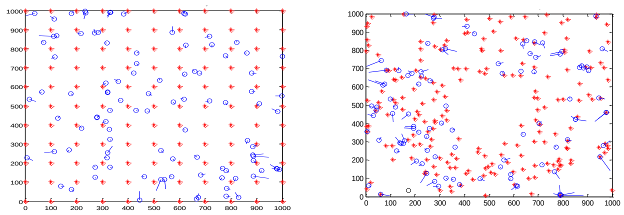
This section discusses the performance of our proposal, K-means Fuzzy Centroid, (for the sake of paper length limitation, more investigation and simulation are for future work) in comparisons with a traditional Centroid, its fuzzy system, and DV-Hop.

i. *Simulation Configurations*

The performance evaluation is based on a standard testbed. To evaluate and justify the correctness, Matlab framework was used with standard libraries including recommendations provided by Gu, S. *et al.* [23]. Our simulated machine is Windows 7 64-bit with Intel Core Q8400 2.66 GHz, 4 GB DDR-SDRAM, and 320GB 7200 rpm hard disk.

To consider the effect of large-scale networks including in our intensive simulation, a simulation parameter includes a variation of anchor nodes in range of 121, 196, and 441, corresponding to the grid deployment requirement of 100×100, 75×75, and 50×50 m^2, over 1000×1000 m^2. Here, there are two main topologies, i.e., grid or non-uniform (with five holes) [20] as examples shown in Figure 3. Note that the number of unknown nodes is fixed at 100 nodes.

In addition, each of the topology will be simulated with the reflection of signal coverage (radius), i.e., 100 and 200 meters. Once deployed, there is no mobility involved. Note that the signal propagation model follows a log-distance path loss model [23]. Here, the energy consumption of the computing node and transmission logic is not considered in this localization model evaluation [24-25]. There is also assumption that the routing cost is not on our focus, i.e., no limitation of routing protocol selection. Here, the location estimation is within the node coverage in distributed manners.



(a) Grid Distribution

(b) Non-Uniform Distribution (Five Holes)

Figure 3: Node Distribution Deployment (cross = anchor; circle = unknown; line = absolute error)

In this evaluation, Average Location Error (ALE) (meters) is mainly used as a metric to measure the location approximation error [22] as shown in equation (3). Here, (x_{sensor}, y_{sensor}) represents the actual position of unknown nodes and (x_{est}, y_{est}) for their estimations. This computation is over ten trials resulting the approximation average. Again, our proposal was evaluated against a traditional Centroid, its fuzzy, and DV-hop.

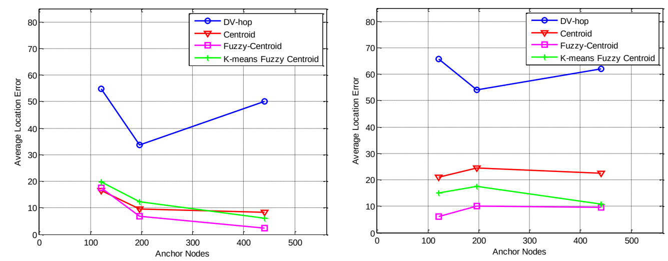
$$ALE = \frac{\sum \sqrt{(x_{est} - x_{sensor})^2 + (y_{est} - y_{sensor})^2}}{\#Sensor\ Nodes} \quad (3)$$

ii. Simulation Results

Figure 4 shows the estimation performance in terms of ALE in grid distribution deployments with 100 and 200 m signal radius. In general, conceptually, with an unrealistic grid distribution, the performance of Centroid should be outstanding, i.e., the unknown will be placed in the square of anchor nodes. However, once the radius is higher, the estimation error should be also higher since more numbers of (non-significant) anchors will be included in the computation process. In the opposite, the error of DV-Hop cannot beat the Centroid. Note that with higher numbers of anchor nodes can bring less location estimation errors.

Specifically considering small coverage (100 m radius), Figure 4a shows that in general, the estimation error of DV-Hop is very high, i.e., around 60 m but with only less than 20 m for the rest (Centroid derivations). In particular, at 121 nodes, the performance of Centroid is superior (16.5 m vs. 17.4, 19.7, and 54.8 m for Centroid, Fuzzy Centroid, K-means Fuzzy Centroid, and DV-Hop, respectively). However, once the number of anchor nodes is higher, its fuzzy system becomes better. For example, at 441 nodes, ALEs are in order of 8.3, 2.4, 6.1, and 50, respectively. This is the fact that the fuzzy takes an effect with additional weights.

Figure 4b also shows the other scenario – larger coverage (200 m radius). Similar trends as in 100 m radius are applied; however, with higher estimation errors since the unnecessary anchor nodes are included in the computation. Here, ALEs are in order of Fuzzy Centroid, K-means Fuzzy Centroid, Centroid, and DV-Hop, i.e., in average around 8, 14, 22, and 60 m respectively. This is also justifiable due to the fuzzy process effect (proper additional weights). Note that our scheme outperform the others except Fuzzy Centroid since the grid deployment may not require another step of unbalanced structure consideration.



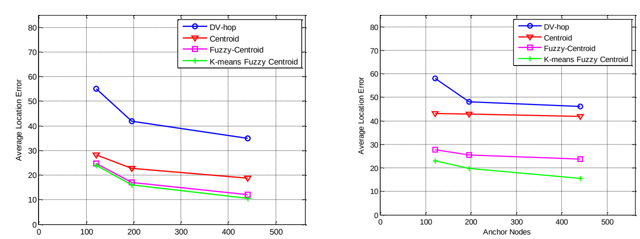
(a) signal radius = 100 meters

(b) signal radius = 200 meters

Figure 4: Average Location Error with Grid Distribution

Figure 5 shows the evaluation results of non-uniform distribution deployment, i.e., with 100 and 200 m radius. Similar to those of Figure 4, the trends follow the grid deployment. However, with unbalanced structure effects, the errors are higher. The performance of our proposal is superior in both coverages, and in order of Fuzzy Centroid, Centroid, and DV-Hop, respectively. In addition, increasing numbers of anchor nodes maintains higher accuracy and with higher radius lowers the estimation precision.

In particular, considering a small radius (100 m), Figure 5a, again, similar trends will be applied. At 441 anchor nodes, the precision of each technique is in order of 10.5, 12, 18.6, and 35 m and 23, 25, 28, and 55 m, with 121 anchor nodes for K-means Fuzzy Centroid, Fuzzy Centroid, Centroid, and DV-Hop, respectively. The errors tend to be higher than those of the grid distribution (Figure 4a) which is in fact justifiable.



(a) signal radius = 100 meters

(b) signal radius = 200 meters

Figure 5: Average Location Error with different topology: Non-Uniform Distribution

With 200 m radius, Figure 5b shows a similar trend of that of Figure 4b. Increasing numbers of anchor nodes may not significantly affect the error. It is noticeable that, here, DV-Hop can mitigate the error effects in non-uniform distribution, i.e., around 50 m. The impact of unbalanced structure is important, and so the estimation errors of the other techniques except our proposal tend to be high, i.e., in average, 25, 42, and 50, respectively, compared to the only 19 m for our proposal.

V. CONCLUSIONS

To improve the performance of Centroid, in particular, with adjustable weights, we investigated the use of weights derived from fuzzy logic systems. We also enhanced its weighted Centroid by applying K-means clustering to figure out a representative of a close-together known nodes, especially when the actual deployment is non-uniform, and all of these is called K-means Fuzzy Centroid. The superior performance is due to the effect of unstructured deployment, e.g., some unnecessary anchor nodes may be included in the fuzzy computation process. Based on our intensive simulation evaluation in large-scale networks and while ranging the signal radius and node density, the performance of our scheme is outstanding, i.e., 44.92%, 16.80%, and 61.72%, better against a traditional Centroid, its fuzzy system, and DV-Hop, respectively. However, although our proposal can confirm the effectiveness, i.e., low location estimation error, more investigation is still needed in other scenarios and constraints. Comprehensive simulation and analysis can be intensively investigated, such as a scalability consideration, network density and diversity, network dimension, diverse irregular topologies, and various signal propagation models considering additional transmission protocol overheads. It is also noted that the effective routing protocol with awareness of node location can be further studied. Another dimension of the study can be also investigated such as the effect of node mobility, and all of these are for possible future work.

REFERENCES

[1] Atzori, L., Iera, A., Morabito, G.: The Internet of things: a survey. Comput. Netw. vol. 54, pp. 2787--2805 (2010).

[2] Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of things (IoT): a vision, architectural elements, and future directions. Future Gener. Comput. Syst. vol. 29, pp. 1645--1660 (2013).

[3] Lindroos, V., Tilli, M., Lehto, A., Motooka, T.: Handbook of Silicon Based MEMS Materials and Technologies. William Andrew, Burlington, MA, USA (2010).

[4] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A survey on sensor networks. IEEE Commun. Mag. vol. 40, pp. 102--114 (2002).

[5] Yick, J., Mukherjee, B., Ghosal, D.: Wireless sensor network survey. Comput. Netw. vol. 52, pp. 2292--2330 (2008).

[6] Han, G., Xu, H., Duong, T. Q., Jiang, J., Hara, T.: Localization algorithms of Wireless Sensor Networks: a survey. Telecommun. Syst. vol. 52, pp. 2419--2436 (2013).

[7] Cheng, L., Wu, C., Zhang, Y., Wu, H., Li, M., Maple, C.: A survey of localization in Wireless Sensor Network. Int. J. Distrib. Sensor Netw. vol. 2012, pp. 1--12 (2012).

[8] Alrajeh, N.A., Bashir, M., Shams, B.: Localization techniques in wireless sensor networks. Int. J. Distrib. Sensor Netw. vol. 2013, pp. 1--9 (2013).

[9] Bulusu, N., Heidemann, J., Estrin, D.: GPS-less low-cost outdoor localization for very small devices. IEEE Pers. Commun. vol. 7, no. 5, pp. 28--34 (2000).

[10] He, T., Huang, C., Blum, B. M., Stankovic, J. A., Abdelzaher, T.: Range-Free localization schemes for large scale sensor networks. In: Int. Conf. on Mobile Comput. Netw. pp. 81--95. ACM, CA, USA (2003).

[11] Niculescu, D., Nath, B.: DV based positioning in ad hoc networks. Telecommun. Syst. vol. 22, pp. 267--280 (2003).

[12] Nagpal, R.: Organizing a global coordinate system from Local information on an amorphous computer. In: Proc. A.I. Memo MIT Artificial Intell. Lab. (1999).

[13] Yun, S., Lee, J., Chung, W., Kim, E.: Centroid localization method in wireless sensor networks using TSK fuzzy modeling. In: Proc. Int. Symp. on Advanced Intell. Syst. pp. 971--974, (2005).

[14] Jang, H., Topal, E.: A review of soft computing technology applications in several mining problems. Applied Soft Comput. vol. 22, pp. 638--651 (2014).

[15] Yun, S., Lee, J., Chung, W., Kim, E., Kim, S.: A soft computing approach to localization in wireless sensor networks. Expert Syst. with Appl. vol. 36, no. 4, pp. 7552--7561 (2009).

[16] Kumar, V., Kumar, A., Soni, S.: A combined Mamdani-Sugeno fuzzy approach for localization in wireless sensor networks. In: Proc. Int. Conf. & Workshop on Emerging Trends in Technol. pp. 798--803. ACM, NY, USA (2011).

[17] Zadeh, L. A.: Fuzzy algorithms. Info. and Control. vol. 12, no. 2, pp. 94--102 (1968).

[18] Larios, D. F., Barbancho, J., Molina, F. J., León, C.: LIS: localization based on an intelligent distributed fuzzy system applied to a WSN. Ad Hoc Netw. vol. 10, no. 3, pp. 604--622 (2012).

[19] Shang, Y., Ruml, W., Zhang, Y., Fromherz, M. P. J.: Localization from mere Connectivity. In: Proc. 4th ACM Int. Symp. on Mobile Ad Hoc Netw. & Comp. pp. 201--212. ACM, MD, USA (2003).

[20] Tran, D.A., Nguyen, T.: Localization in wireless sensor networks based on support vector machines. IEEE Trans. Parallel and Distrib. Syst. vol. 19, pp. 981--994 (2008).

[21] Lloyd, S.: Least Squares Quantization in PCM. IEEE Trans. on Info. Theory. vol. 28, pp. 129--137 (1982).

[22] So-In, C., Permpol, S., Rujirakul, K.: Soft computing-based localizations in wireless sensor networks. Pervasive and Mobile Comput. (in press) (2015).

[23] Gu, S., Yue, Y., Maple, C., Wu, C.: Fuzzy logic based localization in Wireless Sensor Networks for disaster environments. In: Proc. 18th Int. Conf. on Automation & Comput. pp. 1--5. IEEE, CA, USA (2012).

[24] Lee, H., Chung, K., Jhang, K.: A study of Wireless Sensor Network Routing protocols for maintenance access hatch condition surveillance. J. Info. Process. Syst. vol. 9, pp. 237--246 (2013).

[25] Pantazis, N. A., Nikolidakis, S. A., Vergados, D. D.: Energy-efficient routing protocols in wireless sensor networks. IEEE Commun. Survey & Tutorials. vol. 15, pp. 551--591 (2013).