International Journal of Artificial Intelligence and Applications for Smart Devices
Vol.1, No.1 (2013), pp.1-14
http://dx.doi.org/10.14257/ijaiasd.2013.1.1.01

Uncertainty Handling for Sensor Location Estimation in Wireless Sensor Networks using Probabilistic Fuzzy System*

Mahnaz Kadkhoda[1] and M.-R. Akbarzadeh-T[2]

Center of Excellence on Soft Computing and Intelligent Information Processing
Department of Computer Engineering
Ferdowsi University of Mashhad

[1]Ma.Kadkhoda@stu-mail.um.ac.ir, [2]Akbarzadeh@ieee.org

Abstract

With the increasing use of location-sensitive applications, a variety of localization techniques in wireless sensor networks (WSN) have been introduced to date; each aiming to improve accuracy of location estimation. However, most of these studies do not consider the factor of uncertainty which is naturally present in the localization problem. Focusing on range-free localization, this research examines RSS-based location estimation models. Due to environmental conditions, RSS can be influenced by several stochastic factors at the receiving node. In this research, we propose an algorithm using Probabilistic Fuzzy Logic Systems (PFLS) for the first time to address randomness and uncertainty in the location estimation problem. After optimizing PFLS by Genetic Algorithm (GA), the performance of both PFLS and the GA-optimized PFLS are compared with several other competing strategies under varying noise levels. Experimental results clearly indicate the improved efficiency of the proposed algorithm, particularly at higher noise levels.

Keywords: *Localization, Uncertainty, Fuzzy logic Systems, Probabilistic Fuzzy Systems, Genetic algorithms, Neural Networks*

Nomenclature:

RSS	Received signal strength
RSS_{max}	Maximum RSS value between two sensor nodes
k	A constant which takes into account carrier frequency and transmitted power
d_{ij}	The distance between the *i*th sensor node and the *j*th adjacent anchor node
R_{ij}	RSS value between the *i*th sensor node and the *j*th adjacent anchor node
α	The attenuation exponent
SNR	Signal-to-noise ratio

(X_a , Y_a)	Actual position of sensor node
(X_{est} ,Y_{est})	Estimated position of sensor node
LE	Location Error- The distance between the estimated position and the actual position of the sensor node
ALE	Average Location Error - The average distance between the estimated position and the actual position of all sensor nodes
W_{max}	Maximum weight considered between two sensor nodes

* *This paper is a revised and expanded version of a paper entitled "A probabilistic fuzzy approach for location estimation in Wireless sensor networks." presented at IEEE International Conference on Fuzzy systems, Spain, 2010.*

ISSN: 2288-6710 IJAIASD

1. Introduction

Recent advances in hardware miniaturization and communication technologies have led to the emergence of Wireless Sensor Networks (WSN). A WSN consists of small, inexpensive, battery-powered, and wirelessly connected sensors which are usually employed for distributed sensing of physical events. In most cases, sensors are statically put over vast areas, but these sensors can be mobile as well. They can detect different objects, collect information and transmit messages among each other. WSNs can be used in a wide range of applications, such as environmental monitoring, surveillance, automated health care, intelligent building management, traffic control and object tracking [1-3].

Despite many advantages, these sensors are usually small in size and have many physical limitations. For instance, in many applications it is crucial to know a node's location. Clearly, the most accurate and reliable way to obtain this information is to equip each node with a GPS receiver. Nevertheless, this method does not seem to be feasible in many cases, which is due to the fact that most sensor nodes are battery operated and cannot be recharged because of deployment in harsh and remote environments [4].

To solve this constraint, researchers have developed many localization methods in which instead of installing GPS on every node, only a few nodes are equipped with GPS hardware. These particular sensor nodes are called anchor nodes and they know their exact positions. Each of the other normal sensors will then calculate their locations based on the information they receive through talking to each other. Two main methodologies of distributed localization are range-based and range-free localization. Range-based localization approaches [5-8] are hardware intensive methods that localize a sensor node using techniques such as time of arrival (TOA), time difference of arrival (TDOA), and angle of arrival (AOA).

Although range-free approaches [9-12] normally produce less accurate location results than the range-based ones, they are more economical and provide simpler estimates. In a range-free proximity-based localization algorithm, introduced by Bulusu *et al.*, the anchor nodes broadcast their positions within the network and each sensor node computes its position as a centroid of the positions of all the connected anchor nodes to itself [5]. This is a rather simple and economical method, but can erroneous. Later in 2005, Kim and Kwon [10] proposed an improved version of the same localization algorithm. In their method, anchor nodes are weighed based on their proximity to sensor nodes, and each sensor node computes its position as a weighed centroid of the positions of all connected anchor nodes to itself. This method has proven to have fewer errors than the previous approach, nevertheless its performance is highly related to the design and choice of the weights which is heuristic-based. Sukhyun Yun *et al.*, [13] presented some intelligent localization approaches to improve accuracy of position estimation.

None of the above-mentioned methods address the issue of uncertainty handling which is fundamental in any wireless sensor network. In the real world, WSNs operate in very complex environments with highly limited resources. Nonetheless, to the best of author's knowledge, there are a few papers where the influence of Signal-to-Noise Ratio on the localization accuracy has been taken into account. Among the few related works, Wan-Young Chung *et al.*, [14] in 2007 introduced a RSSI accuracy refinement algorithm to reduce noise in signal strength mainly caused by the effect of reflecting and attenuating objects in the indoor environment. Later in 2008, Songbin Zhou *et al.*, [15] proposed a new node location estimation scheme based on Support Vector Regression (SVR). This method utilizes the generalization capability of SVR to achieve better location estimation in the noisy conditions. Also, Pawel Kulakowski *et al.*, [16] in 2010 presented an AOA localization algorithm which considers available environment noise. In this approach, sensors sample the beacons multiple times and take average values to filter out the noise.

Kadkhoda *et al.*, [17]modeled the problem of location estimation in WSN for the first time by a Probabilistic Fuzzy Logic System (PFLS). The concept of Probabilistic Fuzzy Logic was first introduced in 2001 by Meghdadi and Akbarzadeh [18] as a way of representing and/or modeling existing uncertainty in many real world systems. Their approach was based on combining both the two factors of probability of truth and degree of truth within a unique framework. Later in 2005, Liu and Li presented a formal mathematical framework for PFLS to handle this hybrid paradigm of uncertainty [19].

Similarly, this research focuses on the issues of noise and uncertainty in WSNs. Using PFLS as the main model, it employs GA to present an optimized version of PFLS to enhance localization accuracy. It also examines simulation behavior of the two models (PFLS and optimized PFLS) and makes a comparison between these two and other localization estimation methods such as Fuzzy Logic Systems (FLS) and Neural Networks (NN).

This remainder of the paper is organized as follows. Section II looks at several basic concepts and preliminaries used in the article. Section III introduces the proposed strategy and explains the intelligent localization algorithm that has been developed and used in the research. Section IV, explores the environmental setup and conditions required for simulation. Having experimented with four localization methods of PFLS, FLS, optimized PFLS and NN, Section V, examines and analyzes our findings of all the simulations run in a dynamic environment. The paper is then concluded in the last Section.

2. Preliminaries

Fuzzy modeling techniques are well established and are extensively used for modeling complex and nonlinear deterministic systems [20]. A FLS is an inference system that imitates the human thinking and consists of a fuzzifier, some fuzzy IF–THEN rules, a fuzzy inference engine and a defuzzifier.

A simple conventional fuzzy if-then rule with multiple inputs and single output can be shown in the following form:

$$Rule\ i : If\ x_1\ is\ A_{1,i}\ and\ x_2\ is\ A_{2,i}\ and \ldots.\ x_n\ is\ A_{n,i}\ then\ \tau\ is\ B_i \qquad (1)$$

Where $A_{j,i}(j = 1, \ldots, n)$ and B_i $(i = 1, \ldots, I)$ corresponds to the fuzzy sets in the antecedent and consequent part. I is the number of rules and n is the number of input variables.

Fuzzy modeling techniques are not suitable in their conventional form for probabilistic modeling of randomized and stochastic systems. This has made it inevitable for the need of a probabilistic fuzzy modeling approach as many of the real world complex systems may exhibit randomness in their behavior as well [21].

An overview of the PFLS formulation as described by Liu and Li [19] is included here: “Similar to the ordinary FLS, a PFLS includes fuzzification, fuzzy rules, a fuzzy inference engine and defuzzification stage. The significant difference of PFLS to FLS is that the fuzzification and defuzzification procedure are based on probabilistic fuzzy sets instead of ordinary fuzzy sets. The concept of a probabilistic fuzzy set is defined as follows:

Definition 1*:* For an input variable $\in X$, and its fuzzy membership grade $u \in [0,1]$, the probabilistic fuzzy set $\tilde{A}$ can be expressed by a probability space(U_x, ξ, P) , where U_x is the set of all possible events $\{ u \in [0,1]\}$, ξ is a σ-field, and the probability P is defined on ξ. For all element event E_i’s in U_x

$$P(E_i) \geq 0\ , P(\sum E_i)\text{=}\sum P(E_i), \quad P(U_x) = 1 \qquad (2)$$

where E_i corresponds to an event that u= $u_i \subseteq [0,1]$, $u_i=(1,...,S)$ is a certain value of fuzzy membership grade, and $P(E_i)$ is the probability for the event E_i. S is the number of the element events in the set $\{u \in [0,1]\}$. The probabilistic fuzzy set can be expressed as the union of finite subprobability space as follows:

$$\tilde{A} \ \equiv \ \cup_{x \in X} \ (U_x, \xi, P) \tag{3}$$

The inference engine of PFLS is similar to that of the ordinary FLS which gives a mapping from input fuzzy set to output fuzzy set. The inference process is based on the operation methods of PFLS including union, intersection and complete operation. The ith rule of the PFLS is expressed as

$$Rule\ i\colon\ If\ \ x_1\ \ is\ \ \tilde{A}_{1,i}\ \ and\ \ x_2\ is\ \ \tilde{A}_{2,i}\ and\ ....\ x_n\ \ is\ \ \tilde{A}_{n,i}\ then\ \ \tau\ \ is\ \ \tilde{B}_i. \tag{4}$$

where $\tilde{A}_{j,i}(j = 1, \dots, \bar{J})$, $(\bar{J} = n)$ and $\tilde{B}_i$ $(i = 1, \dots, I)$ are the probabilistic fuzzy set of antecedent and consequent parts, $\bar{J}$ is the number of input variables and I is the number of fuzzy rules".

3. The Proposed Strategy

Having randomly distributed sensor nodes in a simulated environment, this research aims to estimate exact location of sensor nodes using their weights which is computed from Received Signal Strength (RSS). Although this method is not precise, it can be a cue, as obviously if an anchor node sends out high powered signal, the anchor node is probably close to a given sensor node and it should have a higher weight. Conversely, if an anchor node sends out low powered signal, it is likely to be far from the given sensor node and should have a low weight. To overcome the uncertainty of the RSS and the nonlinearity between the RSS and the distance, we use FLS or PFLS to model the relationship between the weight of an anchor node and its RSS. Figure 1 show RSS transmitted between anchors and sensor nodes in an environment.

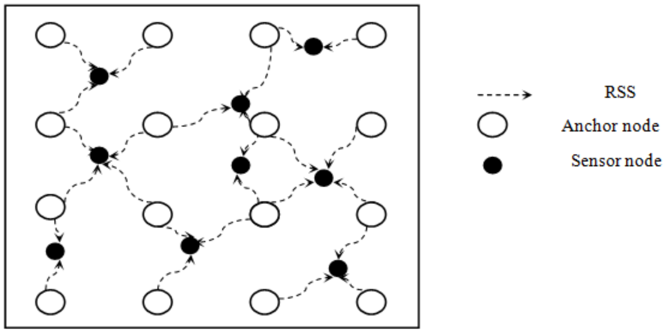

Figure 1. RSS Transmitted between Anchors and Sensor Nodes

A summary of the localization algorithm can be seen below:

1. The adjacent anchor nodes $C_i, i \in N_j$'s are discovered using connectivity, where N_j is the index of anchor j's neighbors.
2. The IDs and positions of anchor nodes C_i and their RSSs are computed.

3. The weight w_i of each anchor node is calculated by either PFLS or Optimized PFLS as explained in the next section.
4. The position of a sensor node are computed by

$$(X_{est}, Y_{est}) = \left(\frac{\sum_{i=1}^{k} w_i X_i}{\sum_{i=1}^{k} w_i}, \frac{\sum_{i=1}^{k} w_i Y_i}{\sum_{i=1}^{k} w_i}\right) \tag{5}$$

Where (X_i, Y_i) are the positions of the anchor nodes, k is the number of adjacent anchor nodes, and w_i is the weight of each anchor node C_i. This algorithm is the same as the approach used by Sukhyun *et al.*, [13] except that PFLS or Optimized PFLS are used to compute the weights of anchor nodes. The proposed strategy is depicted in Figure 2.

3.1. Localization with FLS

The proposed fuzzy If Then rule is in the following form:

$$Rule\ i\text{: } If\ x\ is\ A_i\ then\ \ \tau\ \ is\ \ \ B_i \tag{6}$$

The input variable x is the *RSS* from anchor node and takes a value in the interval [0,$RSS_{\max}$], where $RSS_{\max}$ is the maximum *RSS* value. The output variable τ is the edge weight of each anchor node for a given sensor node and takes a value in the interval [0,$W_{\max}$], where $W_{\max}$ is the maximum weight. The input and output spaces consist of five membership functions: VL, L, M, H, and VH, as shown in Figure 3 and Figure 4. Table 1 presents fuzzy rule base used in this research.

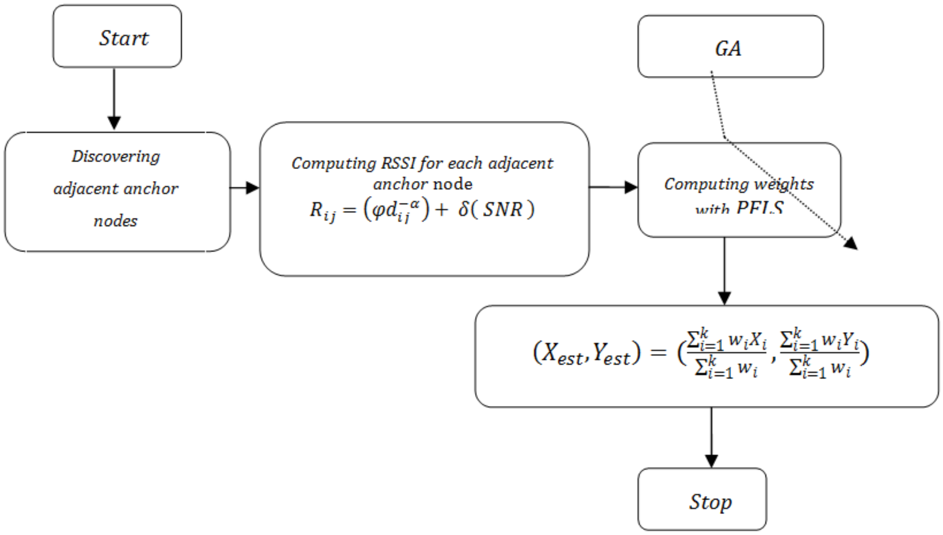
Start
Discovering adjacent anchor nodes
Computing RSSI for each adjacent anchor node
$R_{ij} = (\varphi d_{ij}^{-\alpha}) + \delta(SNR)$
GA
Computing weights with PFLS
$(X_{est}, Y_{est}) = \left(\frac{\sum_{i=1}^{k} w_i X_i}{\sum_{i=1}^{k} w_i}, \frac{\sum_{i=1}^{k} w_i Y_i}{\sum_{i=1}^{k} w_i}\right)$
Stop

Figure 2. The Proposed Algorithm for Location Estimation in WSN

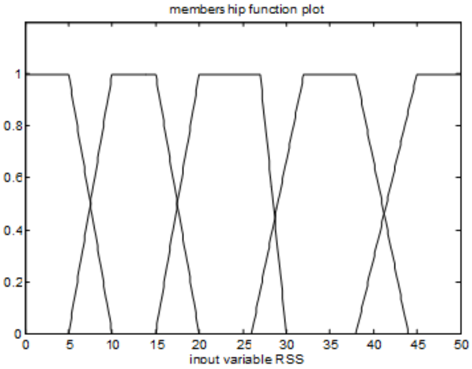

Figure 3. Fuzzy Membership Function of Input

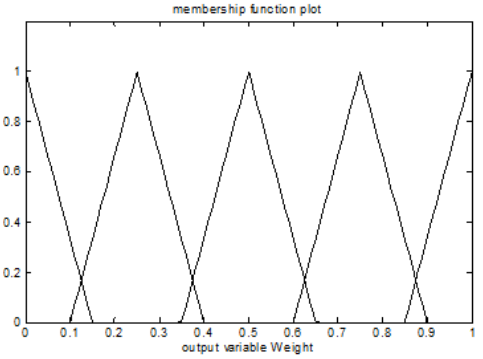

Figure 4. Fuzzy Membership Function of output

Table 1. Fuzzy Rule Base for Edge Weight

Rule	**If *RSS* is**	**Then *weight* is**
Rule 1	Very Low	Very Low
Rule 2	Low	Low
Rule 3	Medium	Medium
Rule 4	High	High
Rule 5	Very high	Very high

3.2. Localization with PFLS

Probabilistic fuzzy logic is a new approach for incorporating probability in fuzzy logic in order to better represent non_deterministic real world systems [21]. In this research, a special case of PFLS is used for the simulation. With Taking into account probability only in the consequent part of the rule, a PFLS rule has the following format:

$$\begin{aligned} &Rule\ i\colon\ If\ x\ is\ A_i\ then\ \hat{y} = y_{i1}\ with\ p_1 = \Pr(y_{i1} | A_i) \\ &\qquad and\ \hat{y} = y_{i2}\ with\ p_2 = \Pr(y_{i2} | A_i) \\ &\qquad and\ \ldots.. \\ &\qquad and\ \hat{y} = y_{iM}\ with\ p_M = \Pr(y_{iM} | A_i) \quad , \quad \textstyle\sum_{j=1}^{M} p_j = 1 \qquad (7) \end{aligned}$$

Where x is the input variable, A_i and y_{ij} $(= 1..M)$ are linguistic variables in the antecedent and consequent parts respectively. $\hat{y}$ is the stochastic output variable that takes one of the values y_{i1}, $y_{i2}, \ldots,$ y_{iM} proportional to the conditional probabilities $\Pr(y_{i1} | A_i)$, $\Pr(y_{i2} | A_i), \ldots,$ $\Pr(y_{iM} | A_i)$.

In order to model the existing uncertainty in the localization, in defining the optimal rule set, each conventional rule is substituted with a probabilistic fuzzy rule. For example this is how a PFLS rule is defined in this research:

$$\begin{array}{llll} Rule\ i\text{: } If\ RSS\ \ is\ low\ \ then & weight = very\ low & with\ p_1 = \%10 \\ & and\ \ weight = \ low & with\ p_2 = \%80 \\ & and\ \ weight = \ medium & with\ p_2 = \%10 \end{array}$$

The consequent part of the probabilistic rule can be thus expressed in a compact form using the output probabilities vector P. Hence, the above rule can be presented as follows:

$$Rule\ i\text{: } If\ RSS\ \ is\ low\ \ then\ \ P = [\ 0.1, 0.8, 0.1, 0.0, 0.0]$$

Finally, the sample probabilistic fuzzy rule set used in the simulation presented here:

$$\begin{array}{lll} Rule\ 1\text{: } If\ RSS\ \ is\ very\ low & then & P = [\ 0.9, 0.1, 0.0, 0.0, 0.0] \\ Rule\ 2\text{: } If\ RSS\ \ is\ low & then & P = [\ 0.1, 0.8, 0.1, 0.0\ ,0.0] \\ Rule\ 3\text{: } If\ RSS\ \ is\ medium & then & P = [\ 0.0\ , 0.1, 0.8, 0.1, 0.0] \\ Rule\ 4\text{: } If\ RSS\ \ is\ high & then & P = [\ 0.0\ , 0.0\ ,0.1, 0.8, 0.1] \\ Rule\ 5\text{: } If\ RSS\ \ is\ very\ high & then & P = [\ 0.0, 0.0, 0.0\ ,0.1, 0.9] \end{array}$$

Probabilistic fuzzy systems are regarded as an extension of their conventional counterparts. If the statistical parameters are selected such that the degree of randomness tends to zero, then probabilistic fuzzy system is converted to conventional form. In other words, conventional fuzzy logic is a special case of the probabilistic fuzzy logic with zero degree of randomness [18].

3.3. Localization with Optimized PFLS

Unlike PFLS system, in which probability vector P is constant, in Optimized PFLS, this vector is no longer fixed. We knew that the overall performance of the PFLS system could be improved, only if this vector could be dynamically changed, taking into account various conditions of a real environment. Hence, in this research, Genetic algorithms (GA), has been used to determine the value of the previously-constant probability vector. GA is a particular class of evolutionary algorithms that can be applied to find approximate or exact solutions for optimization and search problems. In this case, it has been used to optimize probability vector P which will lead to more accuracy. GA optimization parameters used in this study are listed in Table 3.

Table 3. Parameters of Genetic Algorithm

Parameter	Value
Generation	100
Population	10
Crossover Rate	0.6
Mutation Rate	0.2

4. Simulation Setup

The conditions used for simulation in this paper are identical to the ones applied by Yun *et al.*, [13] in 2008. In this simulation, a 100*100 m^2 region with 121 anchor nodes is used. As shown in Figure 5, anchor nodes are placed regularly within 10m distance from each other, and 60 sensor nodes are randomly placed across the area. A sensor node can receive signals from the adjacent anchor node if it stands at a distance smaller than the transmission range which is 8.94m in this case. Within this particular setup, each anchor node knows its exact position, either through GPS or by other means such as pre-configuration. The radio propagation is perfectly spherical and the transmission range of all radios is identical.

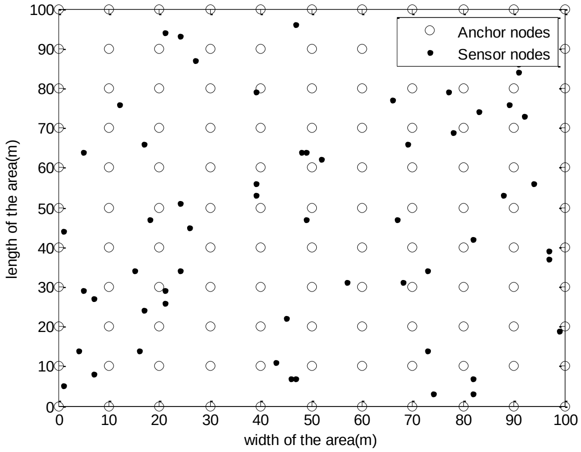

Figure 5. Distribution of Nodes in the Simulated Area

While anchor nodes send out beacon signals, each sensor node listens for a fixed time period and collects the RSS information in the beacon signals received from adjacent anchor nodes in order to locate itself. For the simulation, the following RSS model is used which also takes into account noise:

$$R_{ij} = \left(\varphi d_{ij}^{-\alpha}\right) + \delta(SNR) \tag{8}$$

where R_{ij} is the RSS value between the ith adjacent anchor node and the jth sensor node, φ is a constant which takes into account carrier frequency and transmitted power, d_{ij} is the distance between the ith adjacent anchor and the jth sensor node and α is the attenuation exponent. Here, we use φ =50 and α = 1 and δ (additive white Gaussian noise) to simulate more realistic environment. SNR (signal_to_noise ratio) as a parameter of δ is the ratio of signal power to noise power and it is used to generate various noise levels in the simulations.

To evaluate the proposed approaches, the two following performance criteria are considered: Location Error (LE) and Average Location Error (ALE). Location Error is the distance between estimated location (X_{est}, Y_{est}) and actual position (X, Y) of sensor node. Also, ALE is defined as the average error over all sensor nodes.

$$LE=\sqrt{(X_{est} - X)^2 + (Y_{est} - Y)^{\,2}} \tag{9}$$

$$ALE = \frac{\sum \sqrt{(X_{est}-X)^2+(Y_{est}-Y)^{\,2}}}{N} \quad , \qquad N\text{= The number of sensor nodes.} \tag{10}$$

5. Simulation Results

Having implemented four localization methods in a dynamic environment, this section compares the behavior of these approaches: localization by FLS, PFLS, NN and optimized PFLS.

To examine location estimation results of NN, we applied the method presented by Sukhyun *et al.*, [13] which calculates the location of a sensor node using NN and based on the RSSs received from the anchor nodes. In this approach, all sensor nodes are divided to four sections depending on the number of their adjacent anchor nodes. If RSSs of the anchor nodes are represented by R and k stands for the number of adjacent anchor nodes, the relative position of the sensor node in the training region is estimated as:

$$(X_{est}, Y_{est}) = NN_k(R)$$

For computing the actual location, it is sufficient to add the output of NN to the location of first adjacent anchor node. NN_k is a neural network with k inputs and three layers. We use 100 hidden nodes for the simulation and train them by back propagation algorithm.

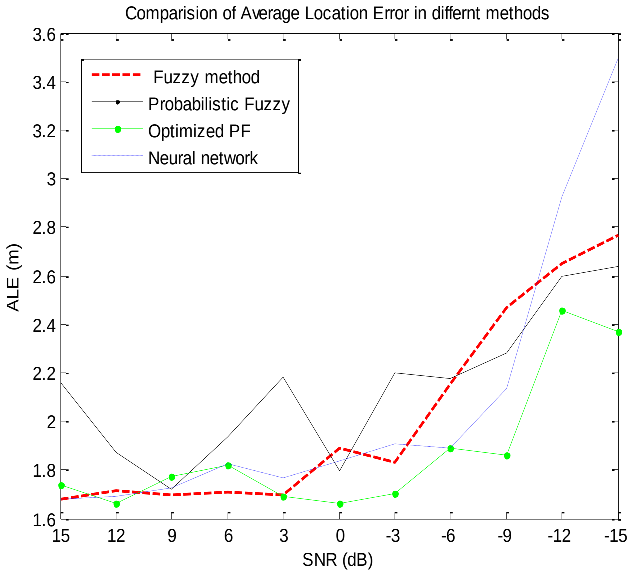

Figure 6. Average Location Error in Four Methods under varying Noise Level

In order to compare all the four aforementioned methods, 10 total simulations were run with variant amount of noise level. In the beginning of the experiment, the level of noise was almost zero. In each simulation, the noise level was increased and subsequently ALE was computed accordingly. Results of all these experiments are depicted in Figure 6. From these findings, it is not difficult to conclude that initially when the system has less noise, NN model is the best and has the least location error. With the gradual increase in noise level, NN loses its localization accuracy very quickly. Contrary to that, the other three methods of FLS, PFLS and optimized PFLS become more reliable.

Focusing on the two methods of FLS and PFLS, the results reveal that at the beginning, FLS shows a better performance in comparison with PFLS. However, as the environment gets noisier, ALE amount of FLS gets higher than PFLS'. In other words, PFLS seems to produce more accurate results in very noisy conditions. Through all these simulations, the behavior of optimized PFLS method is better than PFLS.

Table 4. Comparison of Simulation Results of Methods

Method	Avg. Error(m)		Min Error(m)		Max Error(m)	
FLS	2.2311	%22.3	1.6793	%16.7	2.7645	%27.6
PFLS	2.1423	%21.4	1.7176	%17.1	2.6396	%26.3
Optimized PFLS	1.8748	%18.7	1.6608	%16.6	2.4584	%24.5
Neural network	2.7513	%27	1.6813	%16	3.4994	%34

Following the previous experiment, the average, minimum and maximum location error of all simulation runs have been calculated for each method. The result is shown in Table 4. As it can be seen with the average error of 1.87 meter in 10 meters (%18.7), the maximum error of 2.45 meter (%24.5), and the minimum error of 1.66 meter (%16.6), optimized PFLS seems to be the best method among these four.

As mentioned before, FLS can be much more reliable than PFLS method if the environment is either noise-free or contains a low amount of noise. However, as the environmental noise increases, the PFLS proves to be a better option. Now, the question is how one should determine the critical point after which PFLS gives us better results. And another question should be answered is whether the number of sensors have effect on occurring critical point. In order to find out the answer to these questions, the same experiment is repeated for various number of sensors. The results are presented in Figure 7. As can be seen, there is no relation between the number of sensors and the critical point. Interestingly, this critical point seems to have occurred when the signal_to_noise ratio in the environment reaches a constant number (-15 is here). We also found out that the pace at which noise is increased has a direct impact on the critical point; meaning if the noise level is increased faster, the critical point would happen sooner.

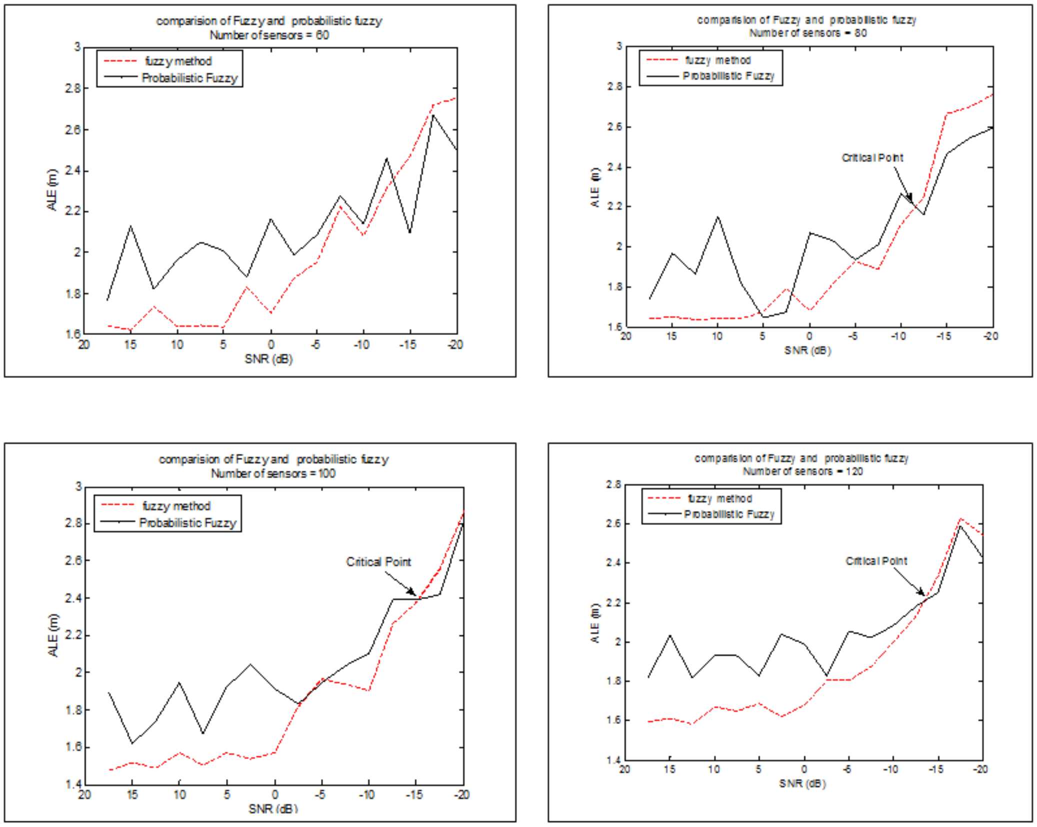

Figure 7. Comparison of Average Location Error in Fuzzy andProbabilistic Fuzzy

6. Conclusion

Sensor location estimation in Wireless Sensor Networks is a non-deterministic problem in real conditions which can be modeled by PFLS. PFLS can be a general framework for combining fuzzy logic with probability theory and hence a better handling of cases where both sources exist. In this research, uncertainty in the localization problem is modeled by PFLS. GA, too, have been used to enhance PFLS' performance. The paper simulates and examines localization with PFLS and GA_optimized PFLS in a dynamic environment and compares the behavior of these methods with others such as FLS and NN. The findings reveal that NN and FLS approaches yield a better performance in a static environment; nevertheless under noisy conditions, PFLS and optimized PFLS approaches seems to be much more effective than the other ones. According to our simulation results, the optimized PFLS approach results in better efficiency in a dynamic environment, particularly when the amount of noise in the system increases considerably.

References

[1] M. Saleem, G. A. Di Caro and M. Farooq, "Swarm intelligence based routing protocol for wireless sensor networks: Survey and future directions", Information Sciences, vol. 181, no. 20, **(2011)** October, pp. 4597-4624.

[2] C. K. Ting and C. C. Liao, "A memetic algorithm for extending wireless sensor network lifetime", Information Sciences, vol. 180, no. 24, **(2010)**, pp. 4818-4833.

[3] M. Y. Naderi, H. R. Rabiee, M. Khansari and M. Salehi, “Error control for multimedia communications in wireless sensor networks: A comparative performance analysis”, Ad Hoc Networks, vol. 10, no. 6, August **(2012)**, pp. 1028-1042.

[4] M. K. Jha, A. K. Pandey, D. Pal and A. Mohan, “An energy-efficient multi-layer MAC (ML-MAC) protocol for wireless sensor networks”, AEU-International Journal of Electronics and Communications, vol. 65, no. 3, **(2011)**, pp. 209-216.

[5] N. Bulusu, J. Heidemann and D. Estrin, “GPS-less low-cost outdoor localization for very small devices”, Personal Communications, IEEE, vol. 7, no. 5, **(2000)** October, pp. 28-34.

[6] N. Patwari and A. O. Hero III, "Using proximity and quantized RSS for sensor localization in wireless networks", Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications, USA, **(2003)**.

[7] M. McGuire, K. N. Plataniotis and A. N. Venetsanopoulos, “Location of mobile terminals using time measurements and survey point”, IEEE Transactions on Vehicular Technology, vol. 52, no. 4, **(2003)**, pp. 999-1011.

[8] W. Jian-guo, W. Zhong-sheng, Z. Ling and S. Fei, "An improved range-based localization algorithm in wireless sensor network", 4th International Conference on Biomedical Engineering and Informatics (BMEI), China, vol. 4, **(2011)**, pp. 2157-2161.

[9] S. Yun, J. Lee, W. Chung and E. Kim, "Centroid localization method in wireless sensor networks using TSK fuzzy modeling", International symposium on advanced intelligent systems, Sokcho, Korea, **(2008)**.

[10] S. Y. Kim and O. H. Kwon, “Location estimation based on edge weights in wireless sensor networks”, Journal of Korea Information and Communication Society, vol. 30, **(2005)**, pp. 877-978.

[11] D. Niculescu and B. Nath, “DV based positioning in ad hoc networks”, Telecommunication Systems, vol. 22, no. 1, **(2003)**, pp. 267-280.

[12] V. Kaseva, T. D. Hamalainen and M. Hannikainen, "Range-free algorithm for energy-efficient indoor localization in Wireless Sensor Networks", Conference on Design and Architectures for Signal and Image Processing (DASIP), Germany, **(2011)**.

[13] S. Yun, J. Lee, W. Chung, E. Kim and S. Kim, “A soft computing approach to localization in wireless sensor networks”, Expert Systems with Applications, vol. 36, no. 4, **(2009)**, pp. 7552-7561.

[14] W. Y. Chung, B. G. Lee and C. S. Yang, “3D virtual viewer on mobile device for wireless sensor network-based RSSI indoor tracking system”, Sensors and Actuators B: Chemical, vol. 140, no. 1, **(2009)**, pp. 35-42.

[15] S. Zhou, X. Zhang and G. Liu, "Node location estimation scheme in wireless sensor networks based on support vector regression", 7th World Congress on Intelligent Control and Automation, China, **(2008)**.

[16] P. Kułakowski, J. Vales-Alonso, E. Egea-López, W. Ludwin and J. García-Haro, “Angle-of-arrival localization based on antenna arrays for wireless sensor networks”, Computers & Electrical Engineering, vol. 36, no. 6, **(2010)**, pp. 1181-1186.

[17] M. Kadkhoda, M.-R. A. Totounchi, M. H. Yaghmaee and Z. Davarzani, "A probabilistic fuzzy approach for sensor location estimation in wireless sensor networks", 2010 IEEE International Conference on Fuzzy Systems, Barcelona, Spain, **(2010)**.

[18] A. H. Meghdadi and M. R. Akbarzadeh-T, "Probabilistic fuzzy logic and probabilistic fuzzy systems", The 10th IEEE International Conference on Fuzzy Systems, Australia, **(2001)**.

[19] Z. Liu and H. X. Li, “A probabilistic fuzzy logic system for modeling and control”, IEEE Transactions on Fuzzy Systems, vol. 13, no. 6, **(2005)**, pp. 848-859.

[20] L. X. Wang and C. Wei, “Approximation accuracy of some neuro-fuzzy approaches”, IEEE Transactions on Fuzzy Systems, vol. 8, no. 4, **(2000)**, pp. 470-478.

[21] A. H. Meghdadi and T. M. Akbarzadeh, "Uncertainty modeling through probabilistic fuzzy systems", Fourth International Symposiu on Uncertainty Modeling and Analysis, Maryland, **(2003)**.

Authors

Mahnaz Kadkhoda received her BS degree in computer engineering from the Ferdowsi University of Mashhad in 1995. She continued her education in Tehran University, obtained an MSc in computer science in 2001. Since then she has been working with the Computer Engineering Department of Birjand University as a lecturer. She is currently a PhD candidate in Computer Engineering Dapartment, Ferdowsi University of Mashhad. Her research interests include Computational Intelligence (Fuzzy Logic, Neural Networks, Genetic Algorithms), intelligent systems, Wireless sensor networks.

Mohammad-R. Akbarzadeh-Totonchi received the B.S. and M.S. degrees from the University of New Mexico, Albuquerque, in 1989 and 1992, respectively, and the Ph.D. degree in evolutionary and fuzzy control of complex systems from the University of New Mexico, in 1998. He is currently a Professor of Electrical Engineering and Computer Engineering at the Ferdowsi University of Mashhad, Mashhad, Iran. From 2006 to 2007, he completed a one-year Visiting Scholar Position at the Berkeley Initiative on Soft Computing, University of California, Berkeley, followed by a visit to the Department of Aerospace and Aeronautics, Purdue University, West Lafayette, IN. From 1996 to 2002, he was affiliated with the NASA Center for Autonomous Control Engineering, University of New Mexico. His research interests include evolutionary algorithms, fuzzy logic and control, soft computing, multiagent systems, complex systems, robotics, and biomedical engineering systems. He has published more than 200 peer-reviewed articles in these and related research fields.

Prof. Akbarzadeh-Totonchi is the Founding President of the Intelligent Systems Scientific Society of Iran. Also, he is the Founding Director of the Center of Excellence on Soft Computing and Intelligent Information Processing of Iran . He is a life member of Eta Kappa Nu (the Electrical Engineering Honor Society), Kappa Mu Epsilon (the Mathematics Honor Society), and the Golden Key National Honor Society. He has received several awards, including the Service Award from the Mathematics Honor Society in 1989, the Outstanding Graduate Student Award in 1998, the Outstanding Faculty Award in 2002, the Outstanding Faculty Award in Support of Student Scientific Activities in 2004, the prestigious Islamic Development Bank Scholarship for High Technology in 2006, and the Outstanding Faculty Award in 2008.