Swarm Intelligence-based Sensor Network Deployment Strategy

Hyungmin Park, Ji-Hyeong Han and Jong-Hwan Kim

Abstract— **The wireless sensor network is a decentralized and self-organized system. Each sensor node in the sensor network should be intelligent enough to carry out its task of monitoring the environment. There would be numerous ways for deploying the sensor nodes in the environment. In this paper, swarm intelligence-based sensor network deployment strategy is proposed. To make a reference point for each sensor node, fuzzy integral is utilized as a multi-criteria decision making process. Three criteria, such as sensor value, crowdedness and confidence, are used for partial evaluation and the degree of consideration for each criterion is represented by fuzzy measure. Global evaluation by fuzzy integral determines the best position for each sensor node independently. To show the effectiveness of the proposed strategy, it is compared with the SPSO07-based deployment strategy through computer simulations in a simulation environment. The results show that the proposed strategy covers much wider area with sensor nodes than the SPSO07-based one.**

I. Introduction

SENSOR networks including wireless sensor networks consist of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions, such as temperatures, sound, vibration, pressure, motion, pollutants, etc. [1], [2]. Various sensor networks are now being used in many industrial and civilian applications including environment and inhabitant monitoring. The sensor networks are constructed with multiple independent sensors. The sensor is usually equipped with a small-sized microcontroller, a radio transceiver or other wireless communication device and an energy source. Therefore, each sensor device or sensor node can be treated as a *mobile* sensor like a mobile robot.

The sensor network, which is formed with multiple mobile sensor nodes, should have both exploring and exploiting capabilities. The exploration capability means that the mobile sensor nodes should cover the entire monitoring environment. The exploiting capability means that the mobile sensor nodes should get together around a certain location where concentrated monitoring is required. In order to maximize the both capabilities of sensor networks, a novel deployment strategy should be provided. Assume that there is a sensor network which is designed to monitor environment pollution. If most of the sensor nodes are located in the clean area, it would be said that the exploiting capability of the network is low. On the other hand, if the sensor nodes flock together in a small polluted area, pollution value of the area can be exactly obtained, but the capability of monitoring of entire area would be very low. In this case, the exploration capability would be low.

Authors are with the Department of Electrical Engineering, KAIST, 373-1 Guseong-dong, Yusevgong-gu, Daejeon, 305-701, Republic of Korea (email: {hmpark, jhhan, johkim}@rit.kaist.ac.kr).

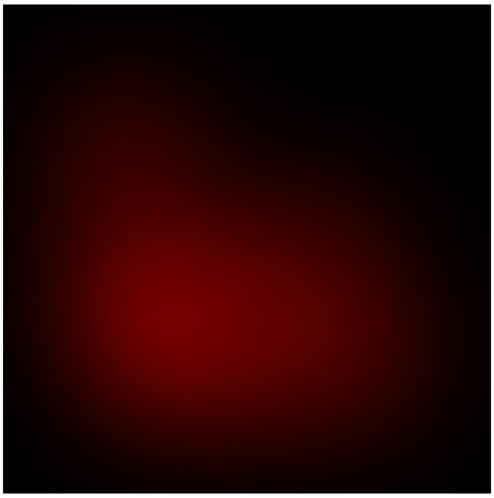


Fig. 1. The simulation environment in which the higher intensity of the red color indicates higher level of pollution.

If the configuration of mobile sensor nodes is controlled by a central controller, the monitoring center controls the sensor network to collect some data at a location, directed by a human operator. However, the sensor network is usually designed as a decentralized configuration such that the monitoring center only collects data from the sensor nodes. It means the center does not control the location of each sensor node. Therefore, each sensor node should have the ability of deciding its location according to the condition of the monitoring environment, the locations of other sensors, and other necessary criteria such as sensor value, crowdedness, confidence, etc.

Although the sensor network problem deals with the covering ability of sensor nodes in an environment to be monitored, it is similar to the exploration-exploitation balancing problem in evolutionary computation [3]-[5]. In evolutionary algorithms, there would be no problem even if probing points, such as artificial genes in GA and particles in PSO, converge into a single global optimum point. But in this monitoring problem, sensor nodes should be located with some distance to each other [6]-[8].

This paper proposes a novel deployment strategy based on swarm intelligence for sensor nodes in a decentralized wireless sensor network. To make a reference point for each sensor node, fuzzy integral is utilized as a multi-criteria decision making process. Three criteria, such as sensor value, crowdedness and confidence, are used for partial evaluation.

978-1-4244-8126-2/10/$26.00 ©2010 IEEE

The degree of consideration for each criterion is represented by fuzzy measure. Global evaluation by fuzzy integral of partial evaluations with respect to fuzzy measures determines the best position for each sensor node independently. To show the effectiveness of the proposed strategy, a test problem is provided in a computer simulation environment and the proposed strategy is compared with the SPSO07-based deployment strategy. The simulation results show that the proposed strategy covers much wider area with sensor nodes than the SPSO07-based one.

The rest of this paper is organized as follows. Section II introduces the test problem and Section III describes the converging and mapping problems of classic strategies. Section IV presents the main concept of the proposed strategy and Section V presents the simulation results. Finally, concluding remarks follow in Section VI.

II. Test Problem Descriptions, Limitations, and Configurations

A. *Goal of the test problem*

The test problem in this paper is for environment monitoring with the following objectives:

- Deploy the sensor nodes widely to cover the whole monitoring region.
- More sensor nodes should get together at the polluted area.

The objectives of the problem are similar to the optimization problem in multi-dimensional space. The main difference between them is that the sensors should cover the polluted area to collect and report data to the monitoring center, while in the optimization problem the probing point should be a global solution. Fig. 1 shows the simulation environment. In the figure, the size of monitoring region is 500 pixels by 500 pixels in which sensor nodes are deployed. Each pixel is mapped into a single location and the intensity of the color red in each pixel indicates the pollution rate of which range is between zero and 255.

Considering practical applications, the sensor network is assumed to be decentralized with mobile sensor nodes without any central controller. The sensor nodes should be intelligent enough to collect data and decide where to move. Wireless commutation device has a range limit for intercommunication between the sensor nodes.

III. Classic Algorithms and Considerations

A. *Particle Filters*

Particle filters, also known as sequential Monte Carlo methods, are sophisticated model estimation techniques based on simulation. In the field of robotics, they are used to solve the localization problems [12]. However, if the particle filters are used to solve the monitoring problem, there is a critical problem. This means that if mobile sensor nodes utilize the particle filters to decide their next location, they are to eventually converge to around the target point – the most polluted location. Thus, the sensors would flock together in a very small area near the source of pollution. This problem may be solved by using Markov-chain Monte Carlo (MCMC) sampling method [12]. However, the solution has another problem such as mapping problem. To describe the motion of mobile sensor nodes, one particle of the particle filters should be arranged to be matched to one sensor node. The probe particles are recreated every iteration using sampling technique. It means the sensor nodes and the probe particles in the particle filters should be remapped all the time. It is a quite time consuming task for mapping between the particles and the sensor nodes. With N independent sensor nodes, the computation complexity of mapping process between them would be $O(N^2)$. In addition, the mapping algorithm requires additional searching or optimization process.

B. *Particle Swarm Optimization*

The particle swarm optimization (PSO) is an evolutionary searching or optimization algorithm based on swarm intelligence [6]. Each particle in the particle swarm represents a probing point in the multi-dimensional search space. The searching power of PSO is based on the communication between the particles. The particles notify their *observed values*, denoting their fitness, to the other particles. Then, each particle updates its location according to the received data. This behavior looks very useful to solve the given problem. The framework of PSO is fully decentralized. Also, the particle is conserved from the beginning to the end such that one-to-one mapping between the particles and the mobile sensor nodes is possible. Note that standard-PSO 2007 [9], denoted by SPSO07, is the most up-to-date version of the PSO algorithm and considered as a *standard* among the PSO and its variant algorithms. Compare to the first version of PSO, the algorithm gives the standardized swarm size and coefficients for the algorithm itself.

The particle swarm in the PSO algorithm is described as a team of communicating particles. At each time step, each particle chooses a few informants at random, selects the best one from this set, and takes into account the information given by the chosen particle. In the algorithmic viewpoint, SPSO07 provides *random-link* between the particles, while the classic version utilizes the entire swarm as a single link. This provides more exploration characteristics to the particle swarm. Each particle takes *link-best*. If it finds no particle better than itself, then the *reasoning* is that the particle itself is the best such that the particle just takes its current velocity and its previous best position into account. Mathematically, the algorithm can be described for each particle with two cognitive/confidence coefficients w and c as follows:

$$\begin{aligned}\mathbf{v}(t+1) = w\mathbf{v}(t) + R(c)\cdot(\mathbf{p}(t)-\mathbf{x}(t)) + \\ R(c)\cdot(\mathbf{g}(t)-\mathbf{x}(t))\end{aligned} \tag{1}$$

$$\mathbf{x}(t+1) = \mathbf{x}(t) + \mathbf{v}(t+1) \tag{2}$$

where $\mathbf{v}(t)$ is the velocity at time t, $\mathbf{x}(t)$ is the position at time t, $\mathbf{p}(t)$ is the best previous position of the particle, $\mathbf{g}(t)$ is the best position among the best previous positions of the informants of the particle, and $R(c)$ is a number coming from a random distribution, which depends on c.

Note that when the particle has no informant better than itself, it implies $\mathbf{p}(t) = \mathbf{g}(t)$ and then (1) become as follows:

$$\mathbf{v}(t) = w\mathbf{v}(t) + R(c)(\mathbf{p}(t) - \mathbf{x}(t)) . \quad (3)$$

The approach using Standard-PSO 2007 also has a similar problem as in the case of using particle filters. Mobile sensor nodes may not be properly deployed in the monitoring environment, as the PSO is a basically search algorithm. All of the nodes would converge into a single point – the source of pollution. Consequently, the sensor network cannot cover the polluted area properly. This phenomenon is shown in the simulation result section.

IV. Swarm Intelligence-Based Deployment Strategy

The *Swarm Intelligence* (SI) describes the collective behavior of decentralized and self-organized systems. The sensor network with mobile sensor nodes has the same characteristics such that it is possible to describe the motion of each mobile sensor node with swarm intelligence. Each sensor node is modeled as an *intelligent particle*. In other words, the particle controls itself, which means there are neither external central controllers nor control units which control the mobile sensors' movement. The particles can memorize the received data and their previous locations. The particles are able to communicate each other. Each sensor is able to transmit its *current position* and *sensor values* to the other sensors with range limit. At each time step, the sensors collect environment data from the environment and the other sensors. Using the collected data, the sensors calculate and move to their next position.

The overall structural framework of the proposed strategy is similar to the PSO. Therefore, the sensor nodes flock together into the most polluted location if only a single criterion of following the sensor node which has the highest sensor value is used. To solve this problem, additional decision making criteria, such as sensor value, crowdedness and confidence, are adopted and *fuzzy measure* is employed to represent the degree of consideration to each criterion. Global evaluation of each particle is calculated by *fuzzy integral* [10], [11]. The following briefly describes fuzzy measure and fuzzy integral.

A. Decision making criteria

In this paper, three criteria are used by the sensor nodes to determine their movement. The sensor value is used to make the sensor nodes flock to the polluted area. Crowdedness criterion is utilized to make the sensor nodes prefer less crowded area. It makes the sensors scattered in the monitoring region for exploration. Confidence measure makes recent data have more effect compared to the old ones.

1) Sensor value

The sensor value of each sensor node is represented as a single normalized value between 0 and 1. The sensor node which is located in the most polluted location has the normalized sensor value of 1. The normalized value is calculated by using the statistics based technique. Assume that the distribution $\mathbf{D}$ is composed of the collected sensor values of the sensor node itself and those of the neighboring sensors. The cumulative density function, in short CDF, of a stochastic distribution $\mathbf{D}$ is defined as follows:

$$\mathrm{CDF}(\mathbf{x} \mid \mathbf{D}) = \int \mathrm{PDF}(\mathbf{x} \mid \mathbf{D}) d\mathbf{x} \quad (4)$$

where $\mathrm{PDF}(\mathbf{x} \mid \mathbf{D})$ is the probability density function of the distribution $\mathbf{D}$. Note that the distribution $\mathbf{D}$ is presumed by the collected sample data. Then the normalized sensor value v for an observed value $\mathbf{x}$ is defined as follows:

$$v = \mathrm{CDF}(\mathbf{x} \mid \mathbf{D}) \quad (5)$$

where the value of v is bounded between 0 and 1, according to the definition of CDF and PDF.

2) Crowdedness measure

If the mobile sensor nodes are crowded, the location should be less preferred. Therefore, the most crowded location takes the value of zero. The average squared-distance from the other sensor nodes is used as the crowdedness value and it is normalized with CDF based normalization technique that is applied to the sensor values.

3) Confidence measure

If the data are collected at the current iteration, the confidence value for the location is 1. At each iteration, the confidence value is reduced by half. That is, after 3 iterations, the confidence value of the data becomes 0.25. The confidence value below than 0.015625 (i.e. after 6 iterations) is treated as 0. In other words, after 6 iterations, the data is discarded.

B. Fuzzy measure and fuzzy integral

There are three criteria – sensor value, crowdedness and confidence – for deciding a reference point for each sensor node. Therefore, it is a multiple criteria decision making (MCDM) problem. Also, the sensor value and the crowdedness criterion have some relationship. The area with higher sensor value is more crowded because there are more sensor nodes for gathering information in this area. However, sensor nodes prefer less crowded area for exploration. Therefore, the criteria have the opposite relation. Since fuzzy measure and fuzzy integral are used in MCDM problem and fuzzy measure can represent some relation between criteria, they are appropriate to decide a reference point for each sensor node. Fuzzy measure and fuzzy integral are briefly described in the following.

1) Fuzzy measure

Consider a finite space $X = \{x_1, x_2, \cdots, x_n\}$ and its power set which is denoted by $P(X)$. The fuzzy measure on $P(X)$ is defined as follows.

Definition 1: A fuzzy measure *g* defined on $(X, P(X))$ is a set function $g: P(X) \to [0,1]$ satisfying the following axioms:

- Boundary condition

$$g(\phi) = 0\,,\ g(X) = 1. \tag{6}$$

- Monotonic increase

$$\forall A, B \subseteq P(X) \\ \text{if } A \subseteq B \text{ then } g(A) \le g(B). \tag{7}$$

Fuzzy measures are classified as belief measure, plausibility measure, probability measure, etc., which are described in the following.

- Belief measure

Belief measure is a set function, $Bel: P(X) \to [0,1]$, satisfying the following additional axiom:

$$Bel(A_1 \cup A_2 \cup \ldots \cup A_n) \ge \sum_i Bel(A_i) - \sum_{i>j} Bel(A_i \cap A_j) \\ + \cdots + (-1)^{n+1} Bel(A_1 \cap A_2 \cap \cdots \cap A_n). \tag{8}$$

Since $Bel(A \cup A^c) = 1$ and $Bel(A \cap A^c) = 0$, $Bel(A) + Bel(A^c) \le 1$. In other words, the sum of belief measure is less than or equal to 1.

- Plausibility measure

Plausibility measure, denoted by *Pl* is a set function, $Pl: P(X) \to [0,1]$, satisfying the following additional axiom:

$$Pl(A_1 \cap A_2 \cap \ldots \cap A_n) \ge \sum_i Pl(A_i) - \sum_{i>j} Pl(A_i \cup A_j) \\ + \cdots + (-1)^{n+1} Pl(A_1 \cup A_2 \cup \cdots \cup A_n). \tag{9}$$

Since $Pl(A \cup A^c) = 1$ and $Pl(A \cap A^c) = 0$, $Pl(A) + Pl(A^c) \ge 1$. It means that the sum of all the plausibility measure is greater than or equal to 1.

- Probability measure

Probability measure can be also defined as a special case of either belief measure or plausibility measure, which satisfies an additional axiom on additive property.

Note that belief and plausibility measures are mutually dual and can be derived from one another as follows:

$$Pl(A) = 1 - Bel(A^c). \tag{10}$$

Note that belief measure indicates one's confidence of making a decision with certainty; on the other hand, plausibility measure represents one's confidence considering all the plausible cases in making a decision. Therefore, $Bel(A)$ is always less than or equal to $Pl(A)$.

By introducing λ, the degree of interaction between A_i and A_j, a general representation of fuzzy measure, called λ-fuzzy measure, $g: P(X) \to [0,1]$ is defined, which additionally satisfies the following axiom [6]:

$$g(A_i \cup A_j) = g(A_i) + g(A_j) + \lambda g(A_i) g(A_j) \\ \forall A_{i,j} \in P(X), A_i \cap A_j = \phi \tag{11}$$

where $i, j = 1, \cdots, n$ and λ>-1.

Depending on the value of λ the λ-fuzzy measure is considered as belief measure if λ>0, plausibility measure if λ<0, or probability measure if λ=0.

2) Fuzzy integral

For global evaluation of each node over criteria with respect to the degree of consideration for each criterion, either *Sugeno fuzzy integral* or *Choquet fuzzy integral* can be used, which are defined in the following.

Definition 2: Let $h: X \to [0,1]$, where *X* can be any set. The *Sugeno fuzzy integral* of evaluated value, *h* over a subset of $X \in P(X)$ with respect to a fuzzy measure *g* is defined as

$$\int_X h \circ g = \max_i \min[h(x_i), g(E_i)] \tag{12}$$

where $h(x_1) \le \cdots \le h(x_n)$ and $E_i = \{x_i, x_{i+1}, \cdots, x_n\}$ for $x_i \in X$ and $i = 1, \cdots, n$.

Definition 3: Let $h: X \to [0,1]$, where *X* can be any set. The *Choquet fuzzy integral* of evaluated value, *h* over a subset of $X \in P(X)$ with respect to a fuzzy measure *g* is defined as

$$\int_X h \circ g = \sum_{i=1}^{n} \{(h(x_i) - h(x_{i-1})) g(E_i)\} \tag{13}$$

where $h(x_1) \le \cdots \le h(x_n)$ and $E_i = \{x_i, x_{i+1}, \cdots, x_n\}$ for $x_i \in X$, $i = 1, \cdots, n$, and $h(x_0) = 0$.

C. Multi-criteria decision making and update rules

Each mobile sensor node decides its next location according to the following time-domain difference equations:

$$\mathbf{a}(t) = (1 - w) \cdot (\mathbf{v} + C\mathbf{R} \circ (\mathbf{r} - \mathbf{x})) \tag{14}$$

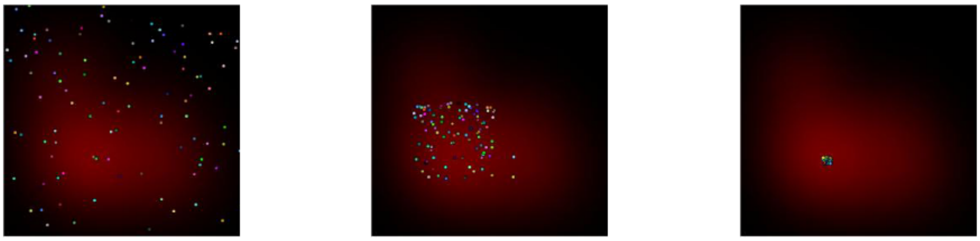
$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \mathbf{a}\Delta t \tag{15}$$

$$\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \mathbf{v}\Delta t \tag{16}$$

where *w* is the slow down factor defined between 0 and 1, vector **x** is the current position, **r** is the reference position, *C* is a constant, **R** is a random vector for scattering, and Δt is the time step and treated as 1 in the rest of this paper for simplicity.

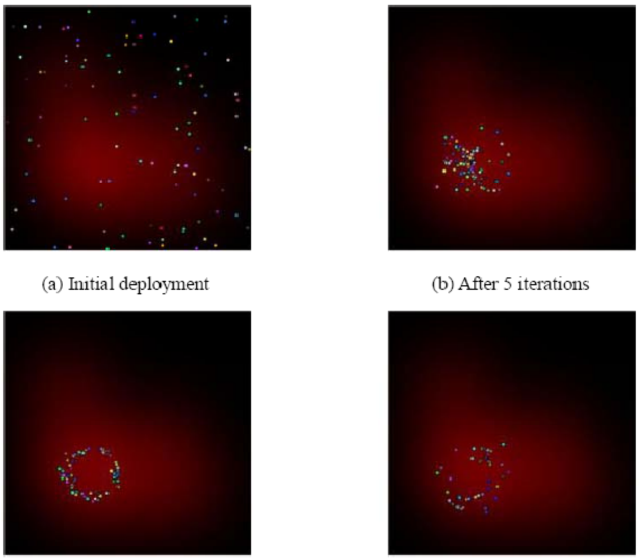
Note that (14), (15), and (16) are inspired by the PSO algorithm such that the velocity vector **v**(*t*) and the acceleration vector **a**(*t*) are considered as virtual vectors, not the physical ones. At the moment of decision making, all of the values except **r** are known or determined.

In PSO, the reference vector **r** is defined as the weighted sum of the location vectors of the global-best particle and each particle's personal best location. In this paper, the proposed strategy utilizes *fuzzy integral as a multi-criteria decision making process* to decide the reference vector **r,** which is the main difference between the proposed strategy and the PSO-based strategy.



(a) Initial deployment (b) After 5 iterations (c) After 20 iterations

Fig. 2.Simulation result using Standard PSO 2007.



(a) Initial deployment (b) After 5 iterations

(c) After 20 iterations (d) After 40 iterations

Fig. 3.Simulation result using the proposed strategy.

Each mobile sensor node advertizes the sensor value to the nearby sensors. For instance, each merchant (in this case, a sensor) in the marketplace shouts what he/she is selling (e.g. the location of the mobile sensor itself) and how much the price is (e.g. sensor value). Similarly, each sensor node advertizes their location and sensor values to its neighbors within communication range. In addition, the sensor nodes reuse the data which are already used in the past decision making process with lower confidence. Finally, using the three criteria, such as sensor value, crowdedness and confidence, each sensor node decides the best data source by fuzzy integral. The reference vector **r** is calculated by fuzzy integral as follows:

$$\mathbf{r} = \arg\max_{\mathbf{x}} \int h \circ \{f_s(\mathbf{x}), f_c(\mathbf{x}), f_t(\mathbf{x})\} \qquad (17)$$

where h is the partial evaluation over each criteria, f_s , f_c , f_t are the fuzzy measures of sensor value, crowdedness, confidence at location $\mathbf{x}$, respectively.

Once each sensor node gets its own reference vector **r**, the mobile sensors update their acceleration **a**, velocity **v**, position **x** according to the update rules of (14), (15), and (16).

V. Computer Simulation Results

A. *Simulation settings*

The test problem is described in Section II. A. The *pollution rate* is described by the intensity of the color *red* in the image. The image size is 500 pixels by 500 pixels. The location of each sensor node is defined as a tuple of two integers in the image, while each sensor node keeps its location as a tuple of two floating-point numbers. The gauss operator is used to map the virtual (floating-point) location into real (integer) location. The location of each sensor node is clipped into the image such that the x- y coordinate values are bounded between zero and 500 pixels.

To show the characteristics of the proposed strategy, the SPSO07 (Standard PSO version 2007, quoted as PSO in the rest of this section) based strategy was used as a control group. With both cases, the number of mobile sensor nodes was 100. In the PSO, particles in the swarm are mapped onto the mobile sensor nodes. The *standard* configuration values were used such that w = 0.721, C = 1.193, and link probability was fixed to 0.25. For simple implementation, the range-limitation was not applied to the PSO-based simulation. With the proposed strategy, the following configuration was used: slow down factor w was 0.721, amplifying factor C was 1.193 and range-limitation was applied to the proposed strategy with the limit of 200 pixels. Additional distortion function for the difference vector $(\mathbf{r}-\mathbf{x})$ was used to strengthen the exploration capability.

The setting of the fuzzy integral which was used in this simulation was provided as follows: *Choquet integral* and λ=0.889 (plausibility measure) were used, weight values for sensor value, crowdedness and confidence were 4, 1 and 2, respectively.

The simulation applet was programmed with Python 2.6-win32 with wxPython 2.8.10, scipy, and numpy module. The test image was created with Adobe® Photoshop using cloud filter. The image was modified to have only red values, where green and blue values were fixed to 0.

B. Results

The simulation results using PSO are shown in Fig. 2(a), 2(b), and 2(c). At the beginning, the sensor nodes, shown as small dots, spread randomly in the monitoring region. But, after 5 iterations, they started to converge and finally after 20 iterations they were almost flocking into a single point; where the most polluted location of the environment. The simulation for PSO was resulted as was expected: the nodes were finally flocking into a global-best position. The exploitation was extremely good, but exploration, which means the covering ability of the strategy, was not good.

The simulation results for the proposed strategy are shown in Fig. 3(a), 3(b), 3(c), and 3(d). Sensor nodes are displayed as dots as the same before. The sensor nodes were randomly spread at the first iteration. After 5 iterations, the sensors were flocking, but not completely converged into a single point like PSO after 20 iterations. Even after 40 iterations, the coverage area of sensor nodes' became wider and the mean of sensors' locations was the most polluted one.

From the result, it can be concluded that the proposed strategy worked well such that the sensor nodes could cover the polluted area without converging into a single point, i.e. the most polluted location. Nevertheless, there were no sensors located in the medium-polluted area, the left upper half and right bottom half. It would be a parametric problem, as the partial evaluated value h, slow downing factor w, and amplifying factor C were not completely tuned. It could be solved if the parameters are properly tuned.

VI. Conclusions

This paper proposed a novel strategy for deploying mobile sensor nodes in a sensor network for environment monitoring. The strategy was based on the swarm intelligence and the key concept of the strategy was inspired by the particle swarm optimization algorithm. To decide the movement of each sensor node, it employed three criteria such as sensor value, crowdedness and confidence measure. To calculate the partial evaluation over the sensor value and crowdedness, statistics-based normalization method was utilized. The preference for the criteria was represented by fuzzy measure as a degree of consideration. Each of the candidate locations was globally evaluated by using fuzzy integral of its partial evaluation over the criteria with respect to the degrees of consideration for the criteria. Compare to the simulation result of PSO-based strategy, the proposed strategy drove the sensors to be spread around the source of pollution to monitor much wider area.

There are two further works for this research. First one is the optimization problem with the configuration parameters to make the sensor nodes cover wider area. The other is related to the evaluation of simulation results. In the figures, human can distinguish the difference of the simulation results of the two algorithms. However, there are no *mathematical or numerical evaluation metrics* for them such that further metric-related research is needed.

Acknowledgement

This research was supported (National Robotics Research Center for Robot Intelligence Technology, KAIST) by Ministry of Knowledge Economy under Human Resources Development Program for Convergence Robot Specialists.

References

[1] Kay Römer and Friedemann Mattern, "The Design Space of Wireless Sensor Networks," IEEE Wireless Communications, pp 54-61, Dec 2004.

[2] Salem Hadim and Mader Mohamed, "Middleware: Middleware Challenges and Approaches for Wireless Sensor Networks," IEEE Distributed Systems Online, Vol. 7, No. 3, March 2006.

[3] Z. H. Zhan, J. Zhang, Y. Li and H. S.-H Chung, "Adaptive Particle Swarm Optimization," IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, Vol. 39, Dec. 2009.

[4] Maurice Clerc and James Kennedy, "The Particle Swarm – Explosion, Stability, and Convergence in a Multidimensional Complex Space," IEEE Transactions on Evolutionary Computation, Vol. 6, No. 1, Feb. 2002.

[5] T. M. Blackwell, "Particle Swarm and Population Diversity," Soft Computing – A Fusion of Foundations, Methodologies and Applications, Springer Berlin, Vol. 9, No. 11, Nov. 2005.

[6] J. Kennedy and R. C. Eberhart, "Particle Swarm Optimization," in Proc. IEEE International Conference on Neural Network, Perth, Australia, 1995, Vol. 4, pp. 1942-1948.

[7] D. N. Wilke, S. Kok, and A. A. Groenwold, "Comparison of linear and classical velocity update rules in particle swarm optimization: notes on diversity," International Journal for Numerical Methods in Engineering, Vol. 70, No. 8, pp. 962–984, 2007.

[8] Maurice Clerc, "Particle Swarm Optimization", ISTE (International Scientific and Technical Encyclopedia), 2006.

[9] Maurice Clerc, "A method to improve Standard PSO," Technical report, Particle Swarm Central.

[10] M. Sugeno, "Theory of Fuzzy Integrals and Its Applications," Doctoral Thesis, Tokyo Institute of Technology, 1974.

[11] E. Takahagi, "On Identification Method of λ-fuzzy Measures using Weight and λ," Journal of Japan Society for Fuzzy Theory and Systems, Vol. 12, No. 5, pp. 665-676, 2000.

[12] Christoper M. Biship, "Pattern Recognition and Machine Learning," Springer Science + Business Media, LLC, 2006

[13] Sheldom M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists 3rd edition," Elsevier Academic Press, 2004.

[14] M.-J. Jung, H.-S. Kim, H.-S. Shim and J.-H. Kim, "Fuzzy Rule Extraction for Shooting Action Controller of Soccer Robot," in Proceeding of IEEE International Fuzzy System Conference, Seoul, Korea, Aug. 1999.

[15] P. Sirdhar, A. M. Madni and M. Jamshidi, "Multi-Criteria Decision Making and Behavior Assignment in Sensor Networks," in Proceeding of 1st Annual IEEE Systems Conference, Honolulu, Hawaii, USA, Apr. 2007.