3D Localization in Large-Scale Wireless Sensor Networks: A Micro-Differential Evolution Approach

Hojjat Salehinejad, *Student Member, IEEE*, Robert Zadeh, *Student Member, IEEE*,
Ramiro Liscano, *Senior Member, IEEE,* and Shahryar Rahnamayan, *Senior Member, IEEE*
Department of Electrical, Computer, and Software Engineering, University of Ontario Institute of Technology,
2000 Simcoe Street North, Oshawa, ON L1H 7K4, Canada
Email: {hojjat.salehinejad, robert.zadeh, ramiro.liscano, shahryar.rahnamayan}@uoit.ca

***Abstract*—Most of the recent proposed approaches for sensor(mote) localization are focused on 2-D environments with limited functionalities. This is mostly due to the nature of problem which is non-linear, large-scale, and has limited hardware resources. The micro-evolutionary algorithms (MEAs) utilize a small-size population to solve optimization problems. Therefore, such algorithms require much less processing time and memory than standard evolutionary algorithms (EA), suitable for implementation on embedded systems. In this paper, a novel protocol for localization of motes in 3-D environments is proposed, simulated, and discussed. The localization problem is modeled as an optimization problem. The proposed model is based on a realistic approach to the localization problem, where possible errors and noises in the localization procedure such as signal strength detection are addressed. To present a suitable approach to solve the proposed optimization model, a comparative study on performance of the micro-differential evolution (MDE) algorithms is performed and the results are discussed.**

***Index Terms*—Localization, Micro-Differential Evolution (MDE), Wireless Sensor Networks (WSNs).**

I. Introduction

Complexity of problems dealing with wireless sensor networks (WSNs) are increasing tremendously. Some of the challenges facing WSNs are spectrum assignment [1] and power management [2]. One of the main challenges is finding the exact location of the mote (sensor), called localization problem. Even though a simple and fast idea to overcome this problem is equipping each mote with geographical positioning system (GPS) modules; however such power hungry systems not only require line-of-sight connection with satellites but also are expensive in comparison with the mote cost itself, particularly for large-scales deployments [3].

Evolutionary algorithms (EAs), such as differential evolution (DE) algorithm, are state-of-the-art methods for solving real-world problems [4]. The term micro-algorithm refers to population-based algorithms with a small population size [5]. The micro-algorithms generally suffer from lack of diversity in the population, which limits the exploration of algorithm for feasible solution. To overcome this problem for micro-differential evolution (MDE) algorithms, methods such as micro-differential evolution with scalar random mutation factor (MDESM) and micro-differential evolution with vectorized random mutation factor (MDEVM) are proposed [5]. The sensor localization problem is formulated as a NP-hard optimization problem [6]. In order to solve such problems, traditional analytical optimization techniques such as quadratic programming and interior-point methods require enormous computational efforts [6], [7]. Therefore, high performance optimization methods with moderate usage of computational resources are of interest. Soft computing algorithms such as genetic algorithm (GA) [9], particle swarm optimization (PSO) [6], and artificial neural networks (ANNs) [10] have been employed for localization in WSNs. In this paper, the localization problem is modeled as an optimization problem and a comparative study on performance of the MDE algorithms family is conducted.

The next section reviews the DE algorithm. In Section III, the proposed approach is discussed and then evaluated in Section IV. Finally, the paper is concluded in Section V.

II. Differential Evolution

The DE algorithm works based on the scaled difference between two individuals of a population's set, where the scaling factor is called the mutation factor [12]. In DE algorithm, the population $\mathbf{P} = \{\mathbf{X}_1, ..., \mathbf{X}_{NP}\}$ consists of NP vectors in generation g, where $\mathbf{X}_i$ is a D-dimensional vector defined as $\mathbf{X}_i = (x_{i,1}, ..., x_{i,D})$. A simple DE algorithm consists of the following steps:

Mutation: This step selects three vectors randomly from the population such that $i_1 \neq i_2 \neq i_3 \neq i$, where $i \in \{1, ..., NP\}$ and $NP \geq 4$ for each vector $\mathbf{X}_i$. The mutant vector is calculated as

$$\mathbf{V}_i = \mathbf{X}_{i_1} + F(\mathbf{X}_{i_2} - \mathbf{X}_{i_3}), \tag{1}$$

where the factor $F \in (0, 2]$ is a real constant number, which controls the amplification of the added differential variation of $(\mathbf{X}_{i_2} - \mathbf{X}_{i_3})$. The exploration of DE increases by selecting higher values for F.

Crossover: The crossover operation increases diversity of the population by shuffling the mutant and parent vector as follows:

$$U_{i,d} = \begin{cases} V_{i,d}, & rand_d(0,1) \leq Cr \; or \; d_{rand} = d \\ x_{i,d}, & otherwise \end{cases}, \tag{2}$$

where $d = 1, ..., D$, $Cr \in [0, 1]$ is the crossover rate

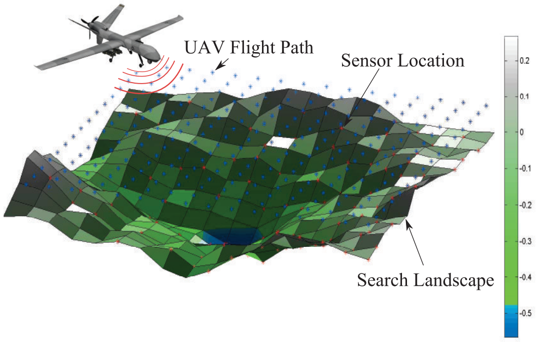


Fig. 1. Motes deployment on a sample landscape and corresponding localization with assistance of a UAV, flighting on a random path and broadcasting a detection signal on the landscape.

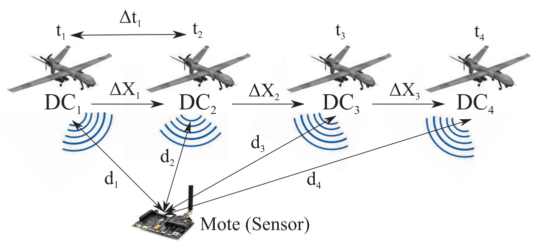


Fig. 2. Sample signal coverage model for four detected coordinates (DC) assisting mote localization.

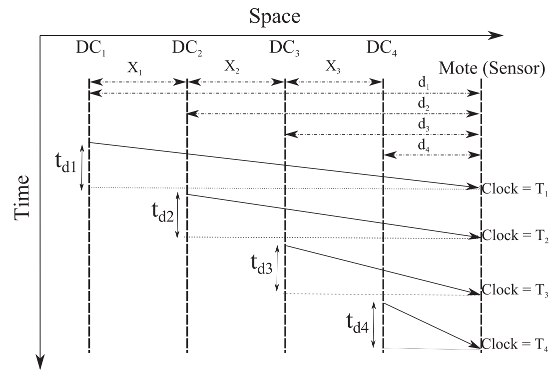


Fig. 3. Timing-diagram of packets arrival from four detection coordinates (DCs) to the mote.

parameter, and $rand(a, b)$ generates a random number in the interval $[a, b]$ with a uniform distribution. Therefore, the trial vector $\mathbf{U}_i$ $\forall i \in \{1, ..., NP\}$ can be generated:

$$\mathbf{U}_i = (U_{i,1}, ..., U_{i,D}). \tag{3}$$

Selection: The $\mathbf{U}_i$ and $\mathbf{X}_i$ vectors are evaluated and compared with respect to their fitness values; the one with better fitness is selected for the next generation.

A. Micro-Differential Evolution

The reduced population size in the MDE algorithm, i.e. $N_P \leq 5$, decreases the computational time dramatically. However, the population size reduction increases the risk of stagnation as well as premature convergence in finding the global optimum solution. The stagnation is not the same as premature convergence. During stagnation, the population remains non-converged but divert and the optimization process does not progress. A large population size offers a more diversified pool of individuals whose recombination offers higher likelihood to locate the global solution [5]. Therefore, reducing the population size while raising the diversity of the population is a key point to achieve a faster convergence speed while maintaining a low risk of premature convergence or stagnation.

B. Micro-Differential Evolution with Vectorized Random Mutation Factor

The mutation factor F plays a major role to deliver diversity into population. The mutation factor F in the DE algorithm is typically a constant mutation factor (CMF), generally set to $F = 0.5$ [5]. In order to deliver diversity to the population, two ideas are proposed recently [5]. The first one is utilizing a random mutation vector for all individuals in each generation in a scalar manner, i. e. the MDESM method. In this case, the mutation factor is considered as

$$F_i = rand(0.1, 1.5), \ \forall i \in \{1, ..., N_P\}. \tag{4}$$

The second idea is utilizing a random vector (not scalar) F for each individual in the population is proposed in [5], the MDEVM algorithm. In this algorithm, for each individual i in the population vector, $F_i = rand(0.1, 1.5)$ selects a value randomly from the interval $[0.1, 1.5]$. Therefore, the mutation factor can be defined for each individual i as

$$\mathbf{F}_i = \{F_{i,1}, ..., F_{i,D}\}, \ \forall i \in \{1, ..., N_P\}, \tag{5}$$

where $F_{i,j} = rand(0.1, 1.5)$, $\forall j \in \{1, ..., D\}$, [5].

III. Proposed Mote Localization Scheme

In this section, the proposed protocol for mote localization in WSNs is presented.

A. Localization Protocol

In real-world applications of WSNs, such as temperature detection in jungles, the motes are deployed randomly on the landscape using air-crafts such as UAVs. In most cases, the motes are not time-synchronized with each other or with their outer environment. In order to localize motes after deployment, we have proposed a protocol as presented in Figures 2 to 3. In the proposed topology, after deployments of sensors on the landscape, the UAV flights on the landscape and broadcasts a signal every Δt_i seconds, after flying a distance ΔX_i as denoted in Figure 2. The message package sent from the UAV is consisted of the packet identification number P_i, time sent t_i, UAV current position $DC_i = (X_{DC_i}, Y_{DC_i}, Z_{DC_i})$, UAV flied distance from last position DC_{i-1} to current position DC_i, i.e. ΔX_{i-1}, and UAV average speed from last sent package at time t_{i-1} to t_i, i.e. ΔV_{i-1}. Each sensor on the landscape S receive a number of packages N_S. Since we are considering the asynchronous mode, which is a more challenging mode than the synchronized mode, the first received packet is used to set the detection clock of sensor to T_1 as denoted in the timing diagram illustrated in Figure 3. As the time passes and arrives to T_2 on the sensor detection clock, that is the time second packet is received, the sensor calculates the time t_{d2} spent on packet signal arrival from UAV at position DC_2 to sensor by using the timing relation as

$$T_2 = T_1 + \Delta t_1 + t_{d2}, \tag{6}$$

where Δt_1 is the time UAV spends to fly from DC_1 to DC_2. By having t_{d2} in mind, the distance from detected position of UAV DC_2 to the sensor location (X_s, Y_s, Z_s) is estimated as $d_2 = V_{EM} \times t_{d2}$ where V_{EM} is the electromagnetic signal speed set to $2.99 \times 10^8 m/sec$, [11]. The same procedure is repeated for the other N_S received packets in the general form

$$t_{di} = T_i - (T_{i-1} + \Delta t_{i-1}), \tag{7}$$

where the distance between each DC_i and the sensor location (X_s, Y_s, Z_s) is computed as $d_i = V_{EM} \times t_{d2}$. The above procedure is conducted for each sensor separated from the rest of sensors on the landscape in a standalone manner. By estimating the distances $d_i \; \forall i \in \{1, ..., N_S\}$ from all received detection coordinates DC_is to the sensor location (X_s, Y_s, Z_s), the sensor localization problem can be modeled.

B. Localization modeling

By considering the $DC_i \; \forall i \in \{1, ..., N_S\}$ coordinates as well as the estimated distances $d_i \; \forall i \in \{1, ..., N_S\}$ to the UAV by each mote, the location of mote, i. e. (X_s, Y_s, Z_s), can be estimated by intersection of $N_S \geq 3$ spheres for a 3-D environment, where the interconnection point is the estimated mote location. However, as it is demonstrated in Figure 4 for a typical 2-D localization problem, in real-world scenarios many errors may occur during the localization procedure such as signal coverage model which is not ideally a sphere, communication channel estimation and modeling, signal path loss and fading, computational errors for instance in speed of UAV or speed of electromagnetic waves, and etc. Therefore, solving the ideal model achieved by interconnecting of spheres is not a realistic approach due to the existence of an additive estimation error between the computed interconnection of spheres and the exact location of mote. To overcome this problem, the additive estimation error is considered in our modeling. In this approach, on the average an equal error for each estimated distance d_i between detection coordinated and the sensor location is assumed.

By having N_S detection coordinates, the sphere equation for spheres $i \in \{1, ..., N_S\}$ is defined as

$$\begin{cases} (X_s - X_{DC_1})^2 + (Y_s - Y_{DC_1})^2 + (Z_s - Z_{DC_1})^2 = d_1^2 \\ \quad \vdots \qquad\qquad \vdots \qquad\qquad \vdots \qquad\qquad \vdots \\ (X_s - X_{DC_{N_S}})^2 + (Y_s - Y_{DC_{N_S}})^2 + (Z_s - Z_{DC_{N_S}})^2 = \\ \qquad\qquad\qquad\qquad\qquad\qquad d_{N_S}^2, \end{cases} \tag{8}$$

where by summing up the equations we will have

$$\sum_{i=1}^{N_S}((X_s - X_{DC_i})^2 + (Y_s - Y_{DC_i})^2 + (Z_s - Z_{DC_i})^2) = \sum_{i=1}^{N_S} d_i^2. \tag{9}$$

By considering the estimated distances as $\tilde{d}_i \forall i \in \{1, ..., N_S\}$ and e_i as the distance estimation error, the exact distance is defined as

$$d_i = \tilde{d}_i + e_i \tag{10}$$

where its square is

$$d_i^2 = (\tilde{d}_i + e_i)^2 \tag{11}$$

By adding up the Eq. (11) for $i \in \{1, ..., N_S\}$ we have

$$\sum_{i=1}^{N_S} d_i^2 = \sum_{i=1}^{N_S} (\tilde{d}_i + e_i)^2. \tag{12}$$

Then, by substituting Eq. (12) into Eq. (9) we have

$$\underbrace{\sum_{i=1}^{N_S}((X_s - X_{DC_i})^2 + (Y_s - Y_{DC_i})^2 + (Z_s - Z_{DC_i})^2) =}_{A} \sum_{i=1}^{N_S} (\tilde{d}_i + e_i)^2. \tag{13}$$

By expanding the right side of Eq. (13) and replacing the left side with A, we have

$$A = \sum_{i=1}^{N_S} \tilde{d}_i^2 + 2 \sum_{i=1}^{N_S} \tilde{d}_i e_i + \sum_{i=1}^{N_S} e_i^2 \tag{14}$$

where by considering equal errors for all estimations as $e = e_i \; \forall i \in \{1, ..., N_S\}$, we can simplify Eq. (14) and arrive to a quadratic equation as

$$N_S e^2 + (2 \sum_{i=1}^{N_S} \tilde{d}_i) e + (\sum_{i=1}^{N_S} \tilde{d}_i^2 - A) = 0. \tag{15}$$

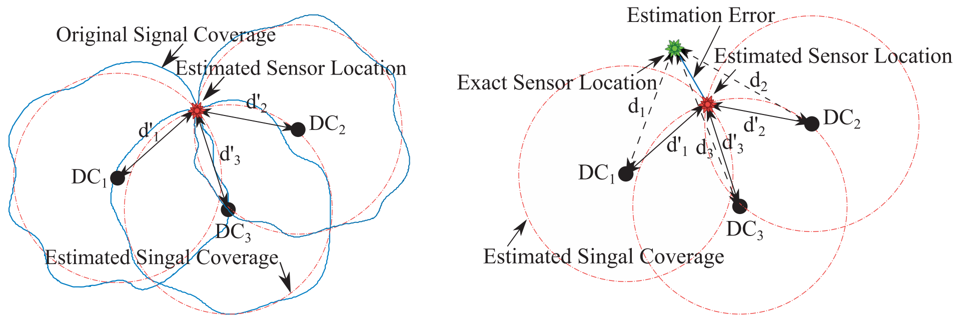
By solving the Eq. (15) for e, the possible solutions are

$$e = \frac{-2 \sum_{i=1}^{N_S} \tilde{d}_i \pm \sqrt{(2 \sum_{i=1}^{N_S} \tilde{d}_i)^2 - 4 N_s (\sum_{i=1}^{N_S} \tilde{d}_i^2 - A)}}{2 N_S} \tag{16}$$

where the negative part of the square root is not applicable. Since we are willing to minimize the estimation error, e, the sensor localization problem can be defined as an optimization problem such as

$$min(e) = min \; \frac{-\sum_{i=1}^{N_S} \tilde{d}_i + \sqrt{(\sum_{i=1}^{N_S} \tilde{d}_i)^2 - N_s (\sum_{i=1}^{N_S} \tilde{d}_i^2 - A)}}{N_S} \tag{17}$$

where the sensor exact location coordinates, i. e. (X_s, Y_s, Z_s), are the variables of the optimization problem.



(a) Original signal coverage, estimated signal coverage, and estimated sensor location.

(b) Exact sensor location, estimated sensor location, and estimation error.

Fig. 4. Sample signal coverage model for three detected coordinates (DC) in a 2-D environment.

Algorithm 1 Micro-Differential Evolution with Scalar Mutation (MDESM) for Sensor Localization

Procedure MDESM
$g = 0$ ***//Initial Population Generation***
for $i = 1 \to N_P$ **do**
  for $d = 1 \to D$ **do**
    $\mathbf{X}_{i,d} = x_d^{min} + rand(0,1) \times (x_d^{max} - x_d^{min})$
  end for
  $\mathbf{P}_i^g = \mathbf{X}_i$
end for***//End of Initial Population Generation***
while $(|BFV - VTR| > EVTR \ \& \ NFC < NFC_{Max})$ **do**
  for $i = 1 \to N_P$ **do**
   //Mutation
    Select three random population vectors from $\mathbf{P}^g$ *where* $(i_1 \neq i_2 \neq i_3 \neq i)$
    $F = rand(0.1, 1.5)$
    for $d = 1 \to D$ **do**
      $\mathbf{V}_{i,d} = \mathbf{X}_{i_1,d} + F(\mathbf{X}_{i_2,d} - \mathbf{X}_{i_3,d})$
    end for
   //End of Mutation
   //Crossover
    for $d = 1 \to D$ **do**
      if $rand(0,1) < C_r$ or $d_{rand} = d$ **then**
        $U_{i,d} = V_{i,d}$
      else
        $U_{i,d} = x_{i,d}$
      end if
    end for
   //End of Crossover
   //Selection
    if $f(\mathbf{U}_i) \leq f(\mathbf{X}_i)$ **then**
      $\mathbf{X}'_i = \mathbf{U}_i$
    else
      $\mathbf{X}'_i = \mathbf{X}_i$
    end if
   //End of Selection
  end for
  $\mathbf{X}_i = \mathbf{X}'_i, \forall i \in \{1, ..., N_P\}$
  $g = g + 1$
  $\mathbf{P}^g = \{\mathbf{X}_1, ..., \mathbf{X}_{N_P}\}$
end while

C. Solving Localization Model

By modeling the sensor localization problems as the optimization problem in Eq. (17), each sensor can utilize the MDESM algorithm (or MDEVM algorithm for more complex problems) [5] to find its own (X_s, Y_s, Z_s) location. The pseudocodes of the MDESM algorithm is presented in Algorithm 1. The termination criterion is met when the difference between best fitness value (BFV) and fitness value to reach (VTR) is less than fitness error-value-to-reach ($EVTR$), or the searching procedure exceeds the maximum number of function calls NFC_{Max}.

IV. Simulation Results

In section, performance of the proposed protocol is evaluated by presenting a comparative study on performance of using the MDE family algorithms.

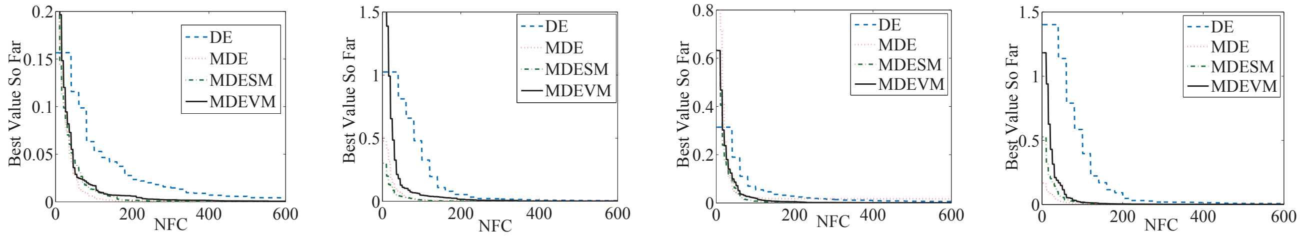
A. Parameters Setting

To simulate the sensor localization problem, 50 sensors are deployed randomly on the landscape illustrated in Figure 1 and the UAV has broadcasted 256 DC messages on the landscape, following a random flight path. The parameter setting of algorithms for all simulations are set as in Table I. The reported values are averaged for N_{Run} independent runs per sensor localization per algorithm to minimize the effect of the stochastic nature of the algorithms on the results [5].

B. Performance Evaluation

In order to study the effect of number of received packets by each sensor in sensor localization performance, the simulations are performed for $N_S \in \{4, 15\}$ received number of packets. The average of performance results for 50 deployed sensors are reported in terms of number of functions calls ($NFCs$), N_{Run} number of runs, success rate (SR), and best achieved accuracy in exact location finding ($Best$) in Table II. As the results demonstrate, performance of the MDESM algorithm is the best among the other algorithms. The MDESM methods has $SR = \%99.77$ with $NFC = 429.46$ and has achieved the error of $Best = 3.06E{-}10$ for $N_S = 4$. This is while for the $N_S = 15$ has the same performance for all the approaches. This means that increasing the number of received samples does not help the system for to achieve better performance.

In order to study the convergence of algorithms toward the solution (exact location of sensor), algorithms performance for best value achieved so far versus NFC for 4 randomly selected sensors are presented in Figure 5.



(a) S_1 (b) S_{10} (c) S_{20} (d) S_{40}

Fig. 5. Performance comparison among the DE, MDE, MDESM, and MDEVM algorithms for $N_S = 4$ and population size $N_P = 20$ for the DE algorithm and $N_P = 5$ for the other algorithms. Plots are scaled for better presentation.

TABLE I
PARAMETER SETTING FOR ALL CONDUCTED EXPERIMENTS, UNLESS STATED

Name	Description	Value
D	Problems Dimension	3
N_P	Population Size (DE algorithm)	20
N_P	Population Size (MDE algorithms)	5
Cr	Crossover Probability Constant	0.9
NFC_{Max}	Maximum Number of Function Calls	$1e3D$
$EVTR$	Objective Function Error Value to Reach	1E-8
N_{Run}	Number of Runs	30

TABLE II
PERFORMANCE RESULTS. THE BEST RESULTS ARE BOLDFACED.

$Algorithm$	N_S	Measure		
		NFC	SR	$Best$
DE	4	2965.00	9.53	2.80E-09
	15	2964.00	10.27	6.60E-09
MDE	4	1315.80	62.53	4.74E-10
	15	1244.20	65.13	4.75E-10
MDESM	4	**429.46**	**99.77**	**3.06E-10**
	15	452.54	99.73	3.60E-10
MDEVM	4	2207.80	51.13	5.92E-10
	15	2256.20	46.40	7.70E-10

For the sensor localization problem in WSNs, the simulation results demonstrate that the MDESM method performs much better than the other approaches. This is due to the dimensionality of problem which is three, where the MDESM can provide enough diversity for the search procedure. This is while the MDE diversity is limited for this problem and the MDEVM and DE algorithms provide more than enough diversity.

V. Conclusion and Further Works

In this paper a new protocol for sensor localization in WSNs is proposed. In this approach, an unmanned aerial vehicle (UAV) flies on the landscape where the sensors are deployed on, and it consequently broadcasts packets while flying. Each sensor on the landscape individually receives a number of packets and form its localization problem as an optimization model by replacing received data in the model. Then, the micro-differential evolution with scalar random mutation factor (MDESM) algorithm is utilized to solve the problem.

The proposed protocol can be further developed for indoor environments. In addition, since the MDESM algorithm works based on a small population size, it can be developed for parallel processing by assigning each individual of the population to a nearby mote.

References

[1] H. Salehinejad, F. Pouladi, and S. Talebi, "A metaheuristic approach to spectrum assignment for opportunistic spectrum access," in Proc. *17th IEEE International Conference on Telecommunications*, pp. 234-238, 2010.

[2] D. Puccinelli and M. Haenggi "Wireless sensor networks: applications and challenges of ubiquitous sensing," *IEEE Circuits and Systems Magazine*, vol.5, no.3, pp.19-31, 2005.

[3] L. Cheng, and et. al., "A survey of localization in wireless sensor network," *International Journal of Distributed Sensor Networks*, 2012.

[4] F. Koroupi, S. Talebi, and H. Salehinejad, "Cognitive radio networks spectrum allocation: An ACS perspective," *Scientia Iranica*, vol. 19, Issue 3, pp. 767-773, 2012.

[5] H. Salehinejad, S. Rahnamayan, H. R. Tizhoosh, and S. Y. Chen, "Micro-Differential Evolution with Vectorized Random Mutation Factor," in *Proc. IEEE Congress on Evolutionary Computation*, 2014.

[6] R. V. Kulkarni, and G. K. Venayagamoorthy, "Particle Swarm Optimization in Wireless-Sensor Networks: A Brief Survey, *IEEE Transactions on Systems, Man, and CyberneticsPart C: Applications and Reviews*, vol. 41, no. 2, 2011.

[7] F. Koroupi, H. Salehinejad, S. Talebi, "Spectrum Assignment in Cognitive Radio Networks Using Fuzzy Logic Empowered Ants," *Iranian Journal of Fuzzy Systems*, vol. 10, no. 6, pp. 1-19, 2013.

[8] R.V. Kulkarni, A. Forster, and G.K. Venayagamoorthy, "Computational Intelligence in Wireless Sensor Networks: A Survey," *Communications Surveys & Tutorials, IEEE* , vol.13, no.1, pp.68,96, 2011.

[9] G. Yang, Z. Yi, and N. Tianquan, "An improved genetic algorithm for wireless sensor networks localization," in *Proc. IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications*, 2010, pp. 5-9.

[10] S. Li and F. Qin,"A dynamic neural network approach for solving nonlinear inequalities defined on a graph and its application to distributed, routing-free, range-free localization of WSNs," *Neurocomputing*, vol. 117, pp. 72-80, 2013.

[11] J. R. Reitz, F. J. Milford, and R. W. Christy, "Foundations of Electromagnetic Theory", (4th Edition), *Addison-Wesley Publishing Company*, USA, 2008.

[12] H. Salehinejad, S. Rahnamayan, and H. R. Tizhoosh, "Type-II Opposition-based Differential Evolutions," in *Proc. IEEE Congress on Evolutionary Computation*, 2014.