Journal of Theoretical and Applied Information Technology
31st January 2019. Vol.97. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

LAGGED MULTI-OBJECTIVE JUMPING PARTICLE SWARM OPTIMIZATION FOR WIRELESS SENSOR NETWORK DEPLOYMENT

[1]ALI NOORI KAREEM, [2] ONG BI LYNN, [3] RBADLISHAH AHMED, [4] HASNA AHMAD, [5] MOHAMED SHUJAA

[1] School of Computer and Communication Engineering, University Malaysia Perlis (UniMAP), Perlis, Malaysia.

E-mail: [1]3li.noori@gmail.com ,[2]drlynn@unimap.edu.my

ABSTRACT

Studies pertaining to wireless sensor network deployment (WSND) have escalated in recent years due to its exceptional function in planning configurations for sensor networks in order to attain maximum coverage and lifetime in a cost-effective manner. Although the approach of meta-heuristic searching optimization has been commonly applied, it has failed in addressing several issues related to multiple objectives and intricate optimization surface. As such, this work developed a novel multi-objective optimization (MOO) called the lagged multi-objective jumping particle swarm optimization (LMOJPSO) in order to overcome the drawbacks of WSND. It aims at finding the best locations and configuration of sensors in 2D environment in order to prolong the life time of the network with obtaining the best coverage and other performance measures. Three types of Pareto front, which were global, iteration (including lag), and local, had been incorporated for optimization search. Upon application to WSND, the proposed algorithm appears to ascertain network coverage and connectivity. When the outcomes of LMOJPSO were compared with the state-of-the-art NSGA-II method, the proposed algorithm seemed to display superior outputs for (MOO).

Keyword: Multi-Objective, WSND, NSGA-II, Coverage, Connectivity, LMOJPSO, Optimization

1. INTRODUCTION

The wireless sensor network (WSN) refers to sensors that are connected wirelessly to collect data from a source and later transmitted to a predetermined sink node using multi-hop communication. A sensor node has several components; processor, sensor board, battery, and radio, to sense, process, and communicate data within certain radii. Hence, WSN need to operate for a longer time without any defect to maintain its efficiency in collecting and disseminating data. As such, a number of studies have examined the significant functions of WSNs in vast applications [1]. Some investigations that had been carried out are collecting data for structural health to perform rapid structural examination [2], remote monitoring using WSN in mine tunnels [3], and determining toxic organic substances from the environment [4]. Nevertheless, WSNs have several drawbacks, such as low energy storage, limited processing capabilities, and restricted communication ability. Additionally, the inability of replacing or charging the battery at constrained areas is another challenge. Thus, improvisation is sought to establish effective communication mechanisms and balanced consumption of energy for prolonged network lifetime and conservation of energy [5] [6]. Another typical method used to conserve energy in WSNs refers to the topology control technique that modifies the ranges of communication for the sensors [7]. The redundant sensor nodes for random deployment sense similar data and waste much energy. Thus, ranges of sensing and communication are essential parameters for WSN sensor nodes to ensure energy conservation [8]. Employing maximized communication and sensing ranges does not only provide redundant data, but also shortens the lifetime of nodes and network. Therefore, effective optimization of WSNs is highly sought. For instance, the sensor nodes in heterogeneous WSNs offer varied communication and sensing ranges that could be adjusted for optimum outputs [9]. Although reduction of these ranges can effectively save energy, the efficiency of connectivity and coverage may be jeopardized. This makes energy conservation, amount of sensor nodes, and connectivity the primary issues in WSNs. Thus, an approach of multi-objective optimization (MOO) is required to overcome issues related to multiple contradicting objectives in optimization search. With that, this paper proposes a MOO approach based on jumping particle swarm optimization (JPSO) to not only address the abovementioned issues, but also to ensure maximum connectivity, coverage, and topology control. Additionally, the proposed approach embeds mobility rule based three factors; (1) best local, (2) best iteration, and (3) best global solutions. The proposed method can also adapt to the rates of crossover and mutation in a dynamic manner without demanding control from external in order to enhance the aspects of diversity and convergence in the optimization algorithm. The remaining sections that make up this paper are described in the following: Section 2 depicts related past studies, and section 3 background, while Section 4 presents the problem to be addressed and related formulation. Next, Section 5 details the methodology and the evaluation steps, whereas Section 6 reveals the experimental outcomes. Lastly, this study ends with Section 7 that provides the study conclusion and several recommendations for future investigations.

2. RELATED STUDIES

The issues of coverage control and quality that can affect the lifetime of WSNs have led to substantial development in this field. [10] developed an algorithm known as complex alliance strategy with multi-objective optimization of coverage (CASMOC) that enhanced coverage of node. The algorithm depicted a proportional correlation for conversion of energy amongst the working nodes and their neighbors, which was later applied to schedule low-energy mobile nodes. With that, the calculation steps were also offered to estimate the minimum sensor nodes required to reduce energy use and to cover all monitoring areas. Nevertheless, the study omitted coverage for multi-targets with nodes of nonlinear rule, as well as high use of energy to verify identities, complete data encryption and end all tasks in a reliable manner within a disoriented setting. [11] also disregarded coverage for sensing area with the required connectivity when applying meta-heuristic methods, wherein location of relay nodes was explained from the stance of three contradicting aims: network reliability, average energy & cost, and average sensitive area. Meanwhile, [12] maximized network lifetime by including several redundant nodes and taking into account both connectivity and coverage excluding transmission and sensing ranges restriction. As such, an algorithm was proposed to maximize the sensor nodes disjoint sets for maintaining both connectivity and coverage (MDS-MCC) with the issue of incomplete NP. [13] developed a greedy algorithm (GA) for placement of lay nodes. This algorithm placed the minimum amount of relay nodes at certain target spots to connect all sensor nodes (targets) with the relay nodes. Nonetheless, the study only looked into the aspect of connectivity between relay and sensor nodes, while ignoring the connectivity between the relay nodes that were placed. The primary issue in WSNs is deployment of sensor with optimum locations subset for a cost-effective network. In fulfilling the demands of connectivity and coverage,[14] maximized coverage and sought optimum locations for placement of sensors, instead of choosing coverage sensor nodes from a sensor network with random deployment. Connectivity was omitted in this study. Next, [15] applied direction-adjustable directional sensors and Voronoi diagram features to develop a GA that enhanced directional sensor network coverage effectively without global information, while disregarding network lifetime. [16] developed a scheme for localization called Opportunistic Localization by Topology Control (OLTC) for sparse underwater sensor networks (UWSNs). Non-localized sensor nodes in UWSNs determine their locations using reference nodes with spatiotemporal relation. Hence, the study developed a model based on game theory specifically for control of topology, along with restricted communication range and adjustable sensing. Meanwhile, [17] examined the issue of sensor nodes placement in WSN by maximizing coverage and minimizing energy consumption via MOO of TPSMA (MOTPSMA), whereas only TPSMA for maximum coverage. However, this study ignored other parameters that might affect sensor nodes placement. [18] applied multi-objective evolutionary algorithm based on decomposition (MOEA/D) to relocate mobile nodes in WSN to minimize energy usage and to maximize coverage for sensing area, while disregarding network connectivity. It appears that no study has simultaneously investigated the aspects of coverage, connectivity, lifetime, and cost or explored for viable solutions. Thus, this study proposes a MOO-based approach to concurrently optimize network lifetime and connectivity, area coverage, cost, nodes adjustment, transmission range, and conservation of energy within a WSN. From the previous study we can observe many researchers ignore important factor in optimization some of them optimum balancing between the number of sensors and the coverage was not taken into consideration. Other problems to be investigated such as optimizing the algorithm to be scalable to larger areas, k-coverage sensor networks, sensor network with various sensing range. Another author Presented a Multi-Objective Genetic Algorithm to optimize the layout of WSN. Number of sensors has not been considered in the presented approach as a design variable, as well as account for the uncertainty in the position of the sensors due to the mode of deployment. The proposed our optimization attempts to overcome the limitations above, this study also incorporated three types of Pareto front to enable interaction with the solutions derived from prior iterations based on pre-defined lag for elitist elucidations.

3. BACKGROUND

Particle Swarm Optimization (PSO) algorithm refers to a modern computing method that takes after the social traits of birds flock [19]. In PSO, swarm denotes the solutions to optimization issue, wherein every possible solution is termed as a particle. The primary goal of PSO is to determine the best location of particles for a determined function. At the start of PSO, every particle is equipped with random parameters and 'flown' into the search space with many dimensions. The particle applies information regarding its best prior locations (individual and global) during every generation to obtain enhanced solution space, and thus, improvised fitness of function. The prior fitness is replaced with the better individual fitness and updates the potential solution. Originally, the PSO derived from Discrete Particle Swarm Optimization (DPSO), which was initiated by [20] to solve continuous optimization issues. In fact, numerous variations of DPSO have been developed to address multi-combinatorial issues. For instance, [21] initiated an algorithm based on DPSO called Jumping Particle Swarm Optimization (JPSO) to overcome issues related to combinatorial optimization. With absence of velocity, JPSO denotes varied jumps/moves for each particle from a position to another within hyperspace that is discrete [22]. [23] employed JPSO to address issues related to Steiner tree within continuous dimension, which can reduce redundant data in WSN, hence conserve energy. Next, [24] used JPSO to overcome a common NP-hard issue for communication network, which is multicast routing ssue. Thus, this present study proposes a novel algorithm from these stances: (1) incorporation of three Pareto fronts into LMOJPSO to explore continuous space, (2) the first MOO-based algorithm that embeds lag for iteration Pareto, (3) its functionality is effectively adapted to WSND.

4. PROBLEM FORULATION

The following is assumed: flat rectangular region A as the region of interest (ROI), sensors that are heterogeneous with initial energy, a static H sink with indefinite energy, sink located at the center of ROI without loss, and wireless sensor nodes that have the ability to sense all points with Rs range, as well as the ability to communicate with other sensor nodes within range Rc. These sensor nodes monitor ROI and transmit sensed data to the sink in a direct manner or by using multi-hop communication. The primary aim of this study is to determine the best sensor deployment and configuration within the environment. Deployment of the sensors is represented by Cartesian coordinates $\{(x_i, y_i), i = 1,2 \dots N\}$, where N is the sensors quantity. Meanwhile, configuration of the sensors is represented by dual variables $\{(Rc_i, Rs_i), i = 1,2 \dots N\}$, where Rc_i, Rs_i refer to communication and sensing radii, respectively. As such, a set of Pareto front solutions (non-dominated solutions) is the best sensors configuration and deployment, in line with the objective functions presented in Equations 1 until 3.

$$f_1(x) = n \tag{1}$$

Where

The intersection is minimized between the sensed areas, for example, Rs_i is reduced as much as possible to conserve energy.

$$f_2(x) = \sum_{i=1}^{N} \sum_{j=1}^{N} Rs_i \cap Rs_j \tag{2}$$

Minimized intersection region refers to reduction in Rs_i while increment in nodes, and vice versa.

$$f_3(x) = \sum_{i=1}^{N} \frac{Rc_i}{d_i} + SD\left(\frac{Rc_i}{d_i}\right) \tag{3}$$

$$d_i = \sqrt{(x_i - x_{sink})^2 + (y_i - y_{sink})^2}$$

Where d_i stands for distance between sink and node, whereas SD reflects standard deviation. The Rc/d_i ratio was minimized due to the heavy data load in nodes located closer to the sink than those further. SD was reduced to aid in decreasing the components of the sensors. A valid solution is obtained upon meeting two essential criteria: connectivity (every node must possess at least a path towards the sink), and coverage (assurance

Journal of Theoretical and Applied Information Technology
31st January 2019. Vol.97. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

that the nodes assigned cover the entire environment).

Development of the proposed MOO framework to optimize the pre-defined issue of multiple objectives is presented in this section. Besides, the framework adopts the PSO-based algorithm, which is a meta-heuristic algorithm that seeks iteratively for the best Pareto front.

5.1 MOJPSO

In this section, we present lagging multi-objective jumping particle swarm optimization. This algorithm is a multi-objective variant of PSO, which means that the output is not an optimum solution, rather, it is a set of non-dominated solutions are called as Pareto. The algorithm is similar to classical PSO in the aspect of moving solutions within the searching space toward the best solution. However, when we talk about multi-objective optimization, there will be no single optimal solution, instead a Pareto where all other solutions will move toward the pareto. In LMOJPOS, there are three Pareto: global, local, and iteration. The global Pareto is calculated from the whole set of solutions for all iterations (or for the iteration within a previous lag). The local Pareto is calculated from the history of the solution itself, and the iteration Pareto is calculated from the current iteration. The combination of the three Pareto is where the solutions are attracted.

The pseudocode of LMOJPSO is given in table (1), As it is shown the solution moves to one of the three Pareto according to the values given to the constant C_1, C_2. Also, as the Pareto is not one solution, a random selection is done to select one solution from each Pareto in order to have the subject solution moving toward. After the solution is updated, its local Pareto is updated. Then we wait until finishing the entire solutions in the iteration in order to update the global Pareto and the iteration Pareto. The combination between two solution, which when logic of moving one subject solution toward the target solution is given in table-2-. As it is seen, the solution moves toward the target with a random velocity. The total behavior is that C_1, C_2 are control parameters for how many solutions and how much speed every solution will move toward each of the local, global, and iteration Pareto.

5. METHODOLOGY

Table 1: The pseudo code of LMOJPSO

```
Input:
numberParticlesInTheSwarm,
maximumNumberOfIterations,
c1, c2
Output
Pareto
Start
Initialization()
Evaluation()
 globalParetoFront=[]
localParetoFront=[]
iterationParetoFront=[]
ParticlesInTheSwarm
loop =true
iter =1 , sol =1 ,lag
while loop
      r= random
      if 0<r<c1
                  Target=uniformRandomSelect
(globalParetoFront,lag)
      else
                  if c1<r<c2

      Target=uniformRandomSelect
(localParetoFront)
                  else
                              Target=
uniformRandomSelect (iterationParetoFront)
                  end if
      end if
      NewSolution=Combine(particlesInTheS
warm(sol),Target)
      Evaluate(NewSolution)
      Update(localParetoFront)
      sol = sol + 1
      if sol > numberParticlesInTheSwarm
               Update(globalParetoFront)
               Update(iterationParetoFront)
                  iter= iter + 1
                  if iter >
maximumNumberOfIterations
                              return
globalParetoFront
                  else
                              sol = 1
                  end if
      end if
end while
end
```

Table 2: The solution interaction or combination in general

```
Input
 Solution
 Target
 LowerLimit
 UpperLimit
 Output
 NewSolution
 Start
 1.  For index=1 until Solution.Dimension
 2.  NewSolution(index)=Solution(index)
     +Const*rand()*(UpperLimit(index)-
     LowerLimit(index))+ LowerLimit(index)
 3.  MutationChance=Rand;
 4.  If(MutationChance>MutationRate)
 5.  NewSolution(index)=Solution(index)
     +Const*rand()*(UpperLimit(index)-
     LowerLimit(index))+ LowerLimit(index)
 6.  Endfor
 7.  End
 8.  End
```

5.2 LMOJPSO for WSND

The first step in the algorithm is to initialize the parameters of the algorithm where the parameters are the following

- Number of solutions in the generation n
- Lower bounds and higher bounds of x, y, Rs, Rc

  $xMin, yMin, RsMin, RcMin,$
  $xMax, yMax, RsMax, RcMax$
- Selection parameters $(c1, c2)$
- Time lag Ʈ Generate Initial solutions

Each new solution go through serial process, Firstly, we generate a random number of sensors then Generate random coordinates (x, y) for the sensors in the region of interest. Using Voronoi method Rs_i values will be generated for the sensors. i.e., Voronoi method guarantees the coverage of the whole environment. If one of the sensor Rs is bigger than Rs_{max} the solution will be refused and the initialization process will start again. After that If the solution satisfies the coverage constraint, a graph represents the sensors network will be built where two sensors are considered connected (neighbors) if the distance between them is less than Rc_{max}. If the graph is connected, Rc_i values are generated, otherwise the solution is refused. The process of computing Rc_i values starts from the sink node where its neighbors are determined. Each neighbor is assigned Rc value equal to the distance between the sensor and the sink node. The process performed on the sink node is performed on each neighbor until all of the sensors are assigned Rc value.

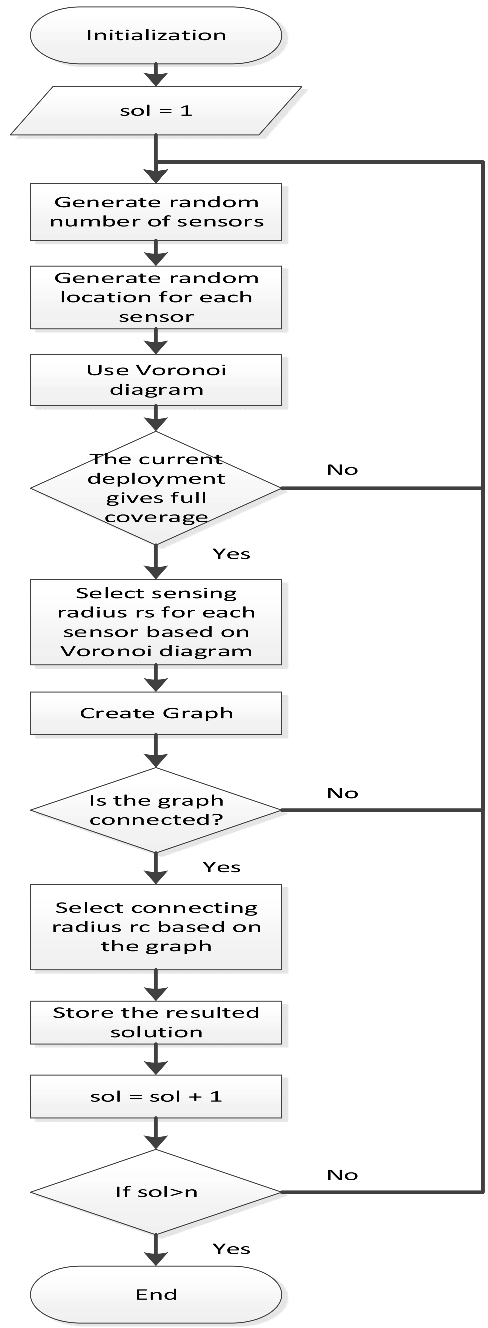


Figure 1: Flow chart for the Initialization procedure

The process of combining between two solutions constitutes from various stages, for each sensor in the current solution, we shift each sensor to the nearest sensor in the target solution. Then the process of selection sensors for the new solution takes place when we make a set contains the

Journal of Theoretical and Applied Information Technology
31st January 2019. Vol.97. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

sensors of the two solutions then find the sensor which achieves the maximum coverage and gets access to the sink node and add this sensor to the new solution. While the environment is not covered, keep adding the sensor, which achieves the best coverage and gets access to the sink or a previously added sensor. This process of combination tries to minimize the number of sensors with guaranteeing full coverage and network connectivity. The pseudocode is given in Figure2.

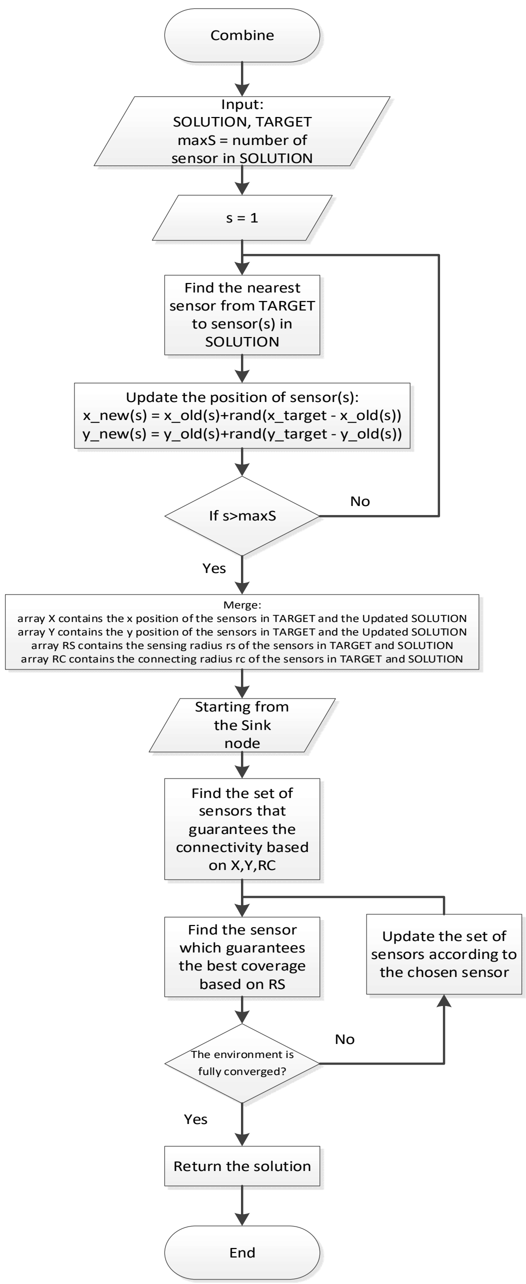


Figure 2:Flow char for the combine procedure

5.3 Evaluation Measures

Some evaluation steps were required to address issues related to MOO. First, the set coverage metric (C-metric) was employed by taking two optimal sets as the input, while the C-metric as the output, as depicted in the following:

$$C(P_{s1}, P_{s2}) = \frac{|\{y \in P_{s2} | \exists x \in P_{s1} : x > y\}|}{|P_{s2}|} \qquad (4)$$

C is equivalent to the non-dominated solutions ratio in P_{s2}, which is subjugated by solutions that are non-dominated in P_{s1}, to the total solutions in P_{s2}. Upon examining PS set, maximizing the $C(x, P_s)$ value is essential with x as the other Pareto set. In the second measure, the hyper-volume metric (HV-metric/S-metric) that is commonly employed in MOO to examine search algorithm performance, is incorporated. This calculates the dominated portion volume of the objective space in relation to point of reference. Higher values signify viable solutions. The HV-metric examines the diversity of the gained solutions and the convergence to the real Pareto front. This is illustrated in the equation that follows: The HV-metric:

$$I_H^*(A) = \int_{(0,0,..0)}^{(1,1,...1)} \alpha_A(z)dz \qquad (5)$$

Where $\alpha_A(z) = \begin{cases} 1 & if\ A \succcurlyeq \{z\} \\ 0 & else \end{cases}$

The final measure refers to the quantity of solutions that are non-dominated to reflect the efficacy of the optimization algorithm in generating viable solutions. This is calculated by determining Ps size, as given below:

$$NDS(N) = |P_s| \qquad (6)$$

Higher NDS values signify sufficient number of solutions. Table 3 presents the generated evaluation measures.

Table 3: Description of evaluation measures

Measure	Class
Set-Coverage	Capacity
NDS	Capacity
Hypervolume	Convergence and Diversity

6. WSN EVALUATION

The performance of the proposed algorithm was compared with NSGA II, which is also an optimization algorithm within the similar arena. Two experimental works were carried out for varied dimensions. As such, 10 experiments were carried out for every scenario.

Table 4: Description of scenario

Scenario	Environment dimension	nmax
1	1200	200
2	1000	100

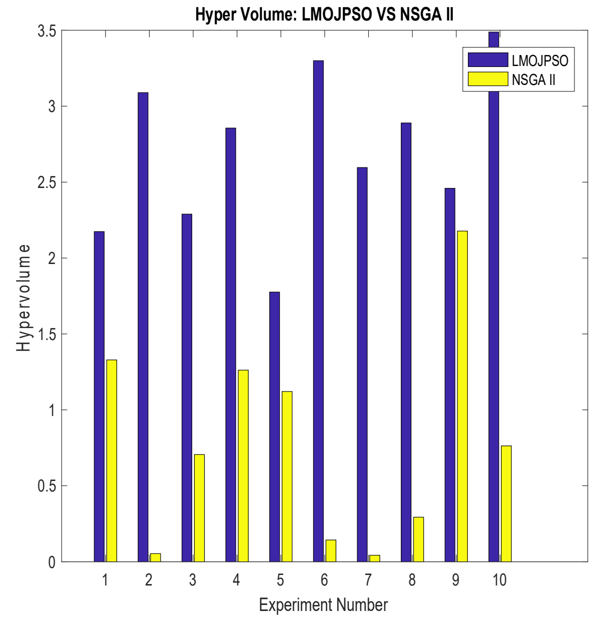
Scenario 10 represents the proposed algorithm parameters, while Table 5 displays the benchmark algorithm parameters.

Table 5: Benchmark algorithm parameters

Number of solutions	200
Number of generation s	100
Crossover probability	0.85
Mutation probability	0.01

For further comparison, the two approaches were tested for WSND in two environments, the first one has a dimension of *1200* $\times$*1200* m^2 and the second one has a dimension of *1000* $\times$*1000* m^2. In each scenario, ten experiments were performed. The measures that were generated, hyper-volume, hyper-volume average, NDS, NDS average, average of set coverage.

For the first scenario:
The first one in figure-3- the hyper volume for the two approaches LMOJPSO and NSGA-II for the environment with dimension of *1200* $\times$*1200* m^2 is shown. Apparently, the hyper-volume of our developed approach LMOJPSO is superior over the benchmark for all experiment. For summarizing the results, the hyper-volume average for each of the two developed approaches is obtained, as it is observed in figure-3-: the average hypervolume of LMOJPSO for the ten experiments is higher than the hyper-volume average of NSGA-II.



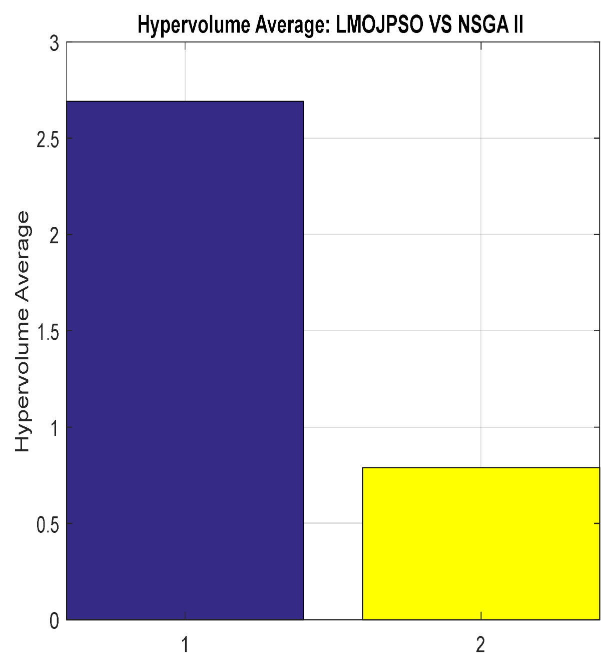
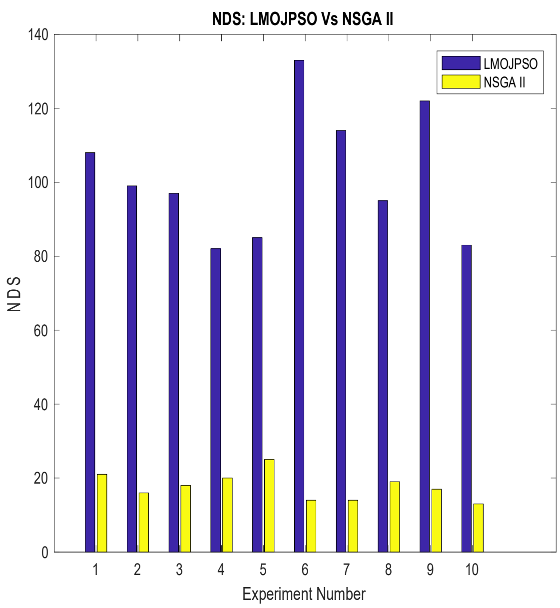


Figure 3: Show hypervolume for the benchmark and the developed algorithm 1200 $\times$*1200* m^2

Journal of Theoretical and Applied Information Technology
31st January 2019. Vol.97. No 2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Another aspect of the performance is NDS which is also generated for each of the ten experiments for the first environment type figure-4- and the average of all NDS of ten experiments in figure-4-. Results show an obvious superiority of LMOJPSO over NSGA-II for the individual experiments and for the overall experiments.



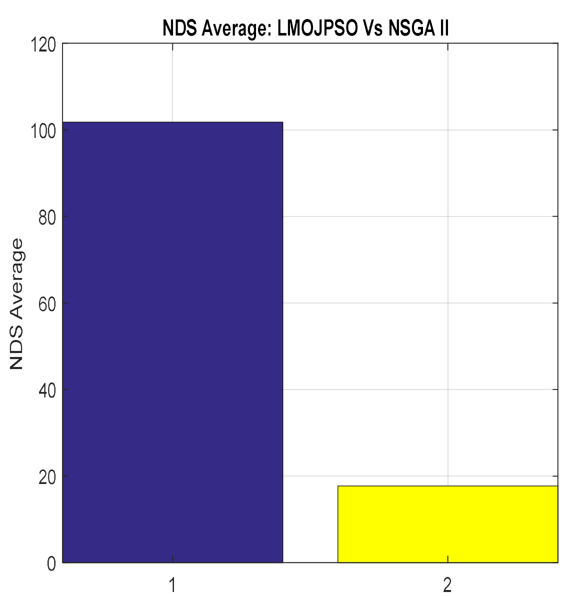


Figure 4: Show NDS values for the benchmark and the developed algorithm $1200 \times 1200\ m^2$

The final measure to generate is the average set coverage for the ten experiments, as it is shown in figure-5-, only 5% of the solutions in LMOJPSO were dominated by NSGA-II.

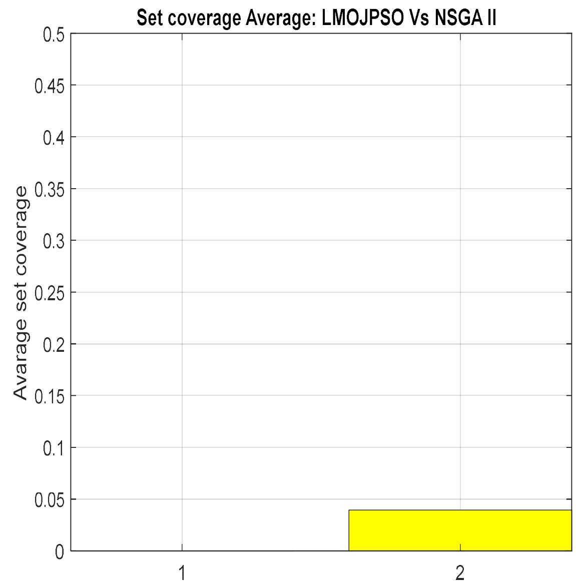
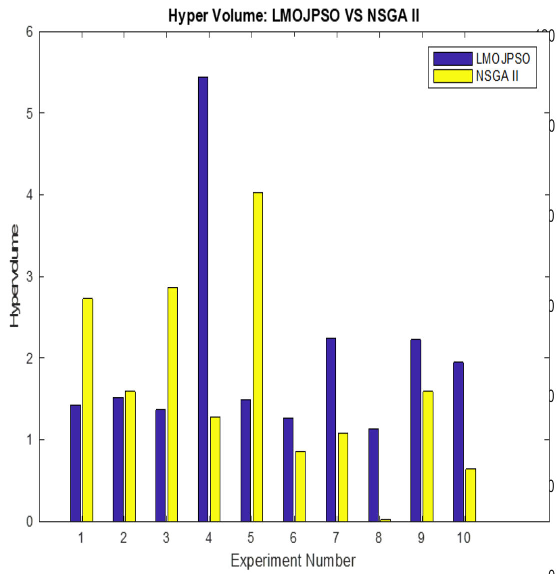


Figure 5: Show set coverage values For the benchmark and the developed algorithm

For the second scenario:

The same sequence of experiments was repeated for second scenario an environment with different size $1000 \times 1000\ m^2$ similar results were obtained: a superiority of LMOJPSO in most experiments and in the overall results for hyper-volume in figures-6-.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195



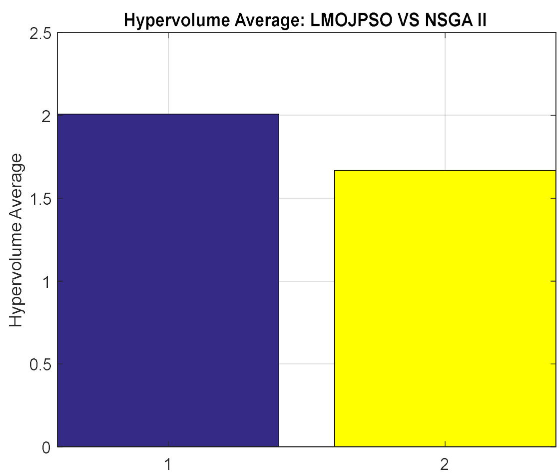
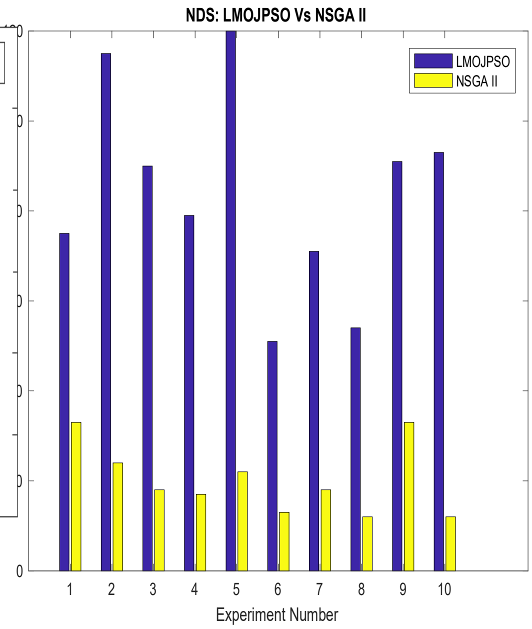


Figure 6: Show Hypervolume for the benchmark and the developed algorithm 1000 ×1000 m^2

Furthermore, NDS for all ten experiments have shown a superiority of LMOJPSO over NSGA-II in figure-7- and the average of NDS is shown in figure with obvious superiority of LMOJPSO over NSGA-II



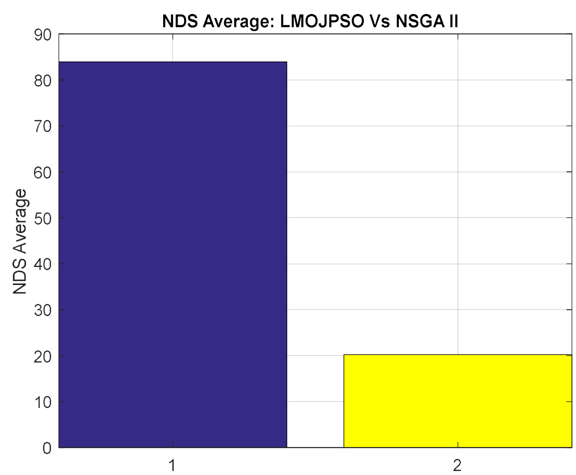


Figure 7: Show NDS values for the benchmark and the developed algorithm 1000 ×1000 m^2

In addition, the set coverage for the two approaches is calculated as an average of the ten experiments and presented in figure-8- to show that only close to 5% of the solutions of LMOJPSO are dominated by NSGA-II.

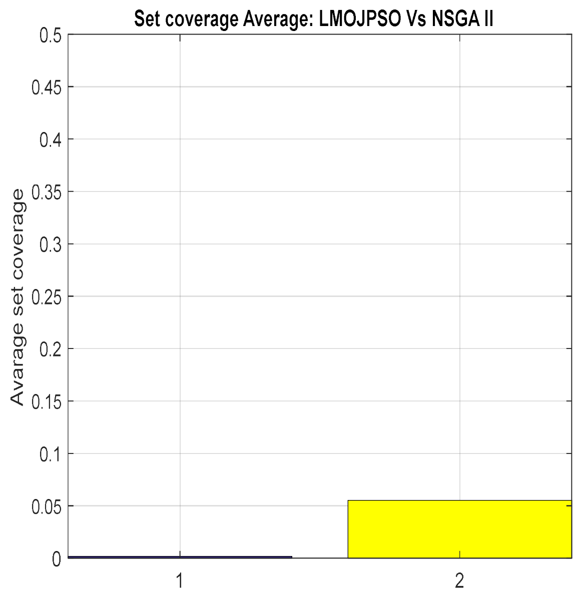


*Figure 8: **S**how set coverage values for the benchmark and the developed algorithm* 1000 $\times$1000 m^2

7. CONCLUSION AND FUTURE WORK

This study proposes a newly-developed MOO framework that incorporated connectivity, coverage, and topology control in WSNs by embedding three Pareto fronts, primarily to explore the generation of viable solutions and to conserve energy. Additionally, the proposed algorithm ascertains network coverage and connectivity. The outcomes of this study signify that LMOJPSO appears to be superior in terms of some measures, apart from being competitive with other algorithms, such as NSGA II. The research conducted in this area is to perform wireless sensor deployment in 2D environment with fixed number of sensors. As such, future studies may look into the use of other accurate frameworks with multiple objectives, which may substantially affect the sensor nodes in terms of lifespan and placement. Besides, the interaction between the solutions could be further enhanced to generate wider exploration within the search space.

REFERENCES

[1] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," *Proc. 1st ACM Int. Work. Wirel. Sens. networks Appl. - WSNA '02*, p. 88, 2002.

[2] D. Mascarenas, E. Flynn, C. Farrar, G. Park, and M. Todd, "A mobile host approach for wireless powering and interrogation of structural health monitoring sensor networks," *IEEE Sens. J.*, vol. 9, no. 12, pp. 1719–1726, 2009.

[3] H. Jiang, J. Wu, L. Chen, S. Chen, and H. Leung, "A Reliable and High-Bandwidth Multihop Wireless Sensor Network for Mine Tunnel Monitoring," *IEEE Sens. J.*, vol. 9, no. 11, pp. 1511–1517, 2009.

[4] F. Tsow *et al.*, "A Wearable and Wireless Sensor System for Real-Time Monitoring of Toxic Environmental Volatile Organic Compounds," *IEEE Sens. J.*, vol. 9, no. 12, pp. 1734–1740, 2009.

[5] W. Dargie, X. Chao, and M. K. Denko, "Modelling the energy cost of a fully operational wireless sensor network," *Telecommun. Syst.*, vol. 44, no. 1–2, pp. 3–15, 2010.

[6] J. C. Lim and C. J. Bleakley, "Trading sensing coverage for an extended network lifetime," *Telecommun. Syst.*, vol. 52, no. 4, pp. 2667–2675, 2013.

[7] M. Younis, I. F. Senturk, K. Akkaya, S. Lee, and F. Senel, "Topology management techniques for tolerating node failures in wireless sensor networks: A survey," *Comput. Networks*, vol. 58, no. 1, pp. 254–283, 2014.

[8] W. K. Lai, C. S. Fan, and C. S. Shieh, "Efficient cluster radius and transmission ranges in corona-based wireless sensor networks," *KSII Trans. Internet Inf. Syst.*, vol. 8, no. 4, pp. 1237–1255, 2014.

[9] X. Chen *et al.*, "ProHet: A probabilistic routing protocol with assured delivery rate in wireless heterogeneous sensor networks," *IEEE Trans. Wirel. Commun.*, vol. 12, no. 4, pp. 1524–1531, 2013.

[10] Z. Sun, Y. Zhang, Y. Nie, W. Wei, J. Lloret, and H. Song, "CASMOC: a novel complex alliance strategy with multi-objective optimization of coverage in wireless sensor networks," *Wirel. Networks*, vol. 23, no. 4, pp. 1201–1222, 2017.

[11] J. M. Lanza-Gutierrez and J. A. Gomez-Pulido, "Assuming multiobjective metaheuristics to solve a three-objective optimisation problem for Relay Node deployment in Wireless Sensor Networks," *Appl. Soft Comput. J.*, vol. 30, pp. 675–687, 2015.

[12] C. P. Chen, S. C. Mukhopadhyay, C. L. Chuang, M. Y. Liu, and J. A. Jiang, “Efficient coverage and connectivity preservation with load balance for wireless sensor networks,” *IEEE Sens. J.*, vol. 15, no. 1, pp. 48–62, 2015.

[13] S. K. Gupta, P. Kuila, and P. K. Jana, “Genetic algorithm approach for k-coverage and m-connected node placement in target based wireless sensor networks,” *Comput. Electr. Eng.*, vol. 56, pp. 544–556, 2016.

[14] J. N. Al-Karaki and A. Gawanmeh, “The Optimal Deployment, Coverage, and Connectivity Problems in Wireless Sensor Networks: Revisited,” *IEEE Access*, vol. 5, pp. 18051–18065, 2017.

[15] T. W. Sung and C. S. Yang, “Voronoi-based coverage improvement approach for wireless directional sensor networks,” *J. Netw. Comput. Appl.*, vol. 39, no. 1, pp. 202–213, 2014.

[16] S. Misra, T. Ojha, and A. Mondal, “Game-Theoretic Topology Control for Opportunistic Localization in Sparse Underwater Sensor Networks,” *IEEE Trans. Mob. Comput.*, vol. 14, no. 5, pp. 990–1003, 2015.

[17] H. Z. Abidin and S. Alam, “Multi-objective Optimization (MOO) approach for sensor node placement in WSN Multi-objective Optimization (MOO) Approach for Sensor Node Placement in WSN,” no. December 2013, 2015.

[18] Qingfu Zhang and Hui Li, “MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition,” *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, 2007.

[19] J. Kennedy and R. Eberhart, “Particle swarm optimization,” *Neural Networks, 1995. Proceedings., IEEE Int. Conf.*, vol. 4, pp. 1942–1948 vol.4, 1995.

[20] J. Kennedy and R. C. Eberhart, “Dc 20212,” *Eng. Technol.*, pp. 4–8, 1997.

[21] M. Sevkli, R. Mamedsaidov, and F. Camci, “A novel discrete particle swarm optimization for p-median problem,” *J. King Saud Univ. - Eng. Sci.*, vol. 26, no. 1, pp. 11–19, 2014.

[22] S. Consoli, J. A. Moreno-Pérez, K. Darby-Dowman, and N. Mladenović, “Discrete particle swarm optimization for the minimum labelling steiner tree problem,” *Nat. Comput.*, vol. 9, no. 1, pp. 29–46, 2010.

[23] Y. Lu, J. Chen, I. Comsa, P. Kuonen, and B. Hirsbrunner, “Construction of data aggregation tree for multi-objectives in wireless sensor networks through jump particle swarm optimization,” *Procedia Comput. Sci.*, vol. 35, no. C, pp. 73–82, 2014.

[24] Y. Marinakis and A. Migdalas, “A particle swarm optimization algorithm for the multicast routing problem,” *Springer Proc. Math. Stat.*, vol. 104, no. October, 2014.