

ENHANCED PARTICLE SWARM OPTIMIZATION FOR EFFECTIVE RELAY NODES DEPLOYMENT IN WIRELESS SENSOR NETWORKS

Bader Alshaqqawi¹, Sardar Anisul Haque²,
Mohammed Alreshoodi³ and Ibrahim Alsukayti⁴

¹Department of Mathematics, Collage of Science,
Qassim University, Buraydah, Saudi Arabia

²Department of Mathematics and Computer Science,
Alcorn State University, Lorman, Mississippi, USA

³Department of Applied Science, Unizah Community College,
Qassim University, Unizah, Saudi Arabia

⁴Department of Computer Science, Collage of Computer,
Qassim University, Buraydah, Saudi Arabia

ABSTRACT

One of the critical design problems in Wireless Sensor Networks (WSNs) is the Relay Node Placement (RNP) problem. Inefficient deployment of RNs would have adverse effects on the overall performance and energy efficiency of WSNs. The RNP problem is a typical example of an NP-hard optimization problem which can be addressed using metaheuristics with multi-objective formulation. In this paper, we aimed to provide an efficient optimization approach considering the unconstrained deployment of energy-harvesting RNs into a pre-established stationary WSN. The optimization was carried out for three different objectives: energy consumption, network coverage, and deployment cost. This was approached using a novel optimization approach based on the integration of the Particle Swarm Optimization (PSO) algorithm and a greedy technique. In the optimization process, the greedy algorithm is an essential component to provide effective guidance during PSO convergence. It supports the PSO algorithm with the required information to efficiently alleviate the complexity of the PSO search space and locate RNs in the spots of critical significance. The evaluation of the proposed greedy-based PSO algorithm was carried out with different WSN scenarios of varying complexity levels. A comparison was established with two PSO variants: the classical PSO and a PSO hybridized with the pattern search optimizer. The experimental results demonstrated the significance of the greedy algorithm in enhancing the optimization process for all the considered PSO variants. The results also showed how the solution quality and time efficiency were considerably improved by the proposed optimization approach. Such improvements were achieved using a simple integration technique without adding to the complexity of the system and introducing additional optimization stages. This was more evident in the RNP scenarios of considerably large search spaces, even with highly complex and challenging setups.

KEYWORDS

Wireless Sensor Networks, Relay Node Placement, Swarm Intelligence, Particle Swarm Optimization, Greedy Algorithm

1. INTRODUCTION

Over the recent years, Wireless Sensor Networks (WSNs) have attracted great interest at both the academic and industrial levels. A variety of WSN applications have emerged in different domains

including remote monitoring, logistics, health care, surveillance, utilities, and military. The deployment of WSNs for these applications involves implementing a number of sensor nodes over varying types of terrains and environments. Sensor nodes are typically small-sized devices of limited computing, sensing, memory, and communication resources. They also have limitations in terms of energy supply being typically powered using batteries. This would limit the lifetime of the nodes in addition to posing other challenges regarding the costly and time-wasting process of battery replacement.

Therefore, one of the common critical considerations for WSNs is energy efficiency. The most energy-consuming operation for sensor nodes is wireless radio communications. Accordingly, communication technologies supporting low-power low-rate wireless transmission/reception have become popular options for improving energy consumption and network lifetime. However, such an approach could come at the cost of limiting network coverage in practice. It is critical to address these WSN limitations to stop them hindering real-world WSN deployments. One strategic solution is the deployment of special nodes, Relay Nodes (RNs), in the network in addition to sensor nodes. RNs are typically mains-powered or energy-harvesting devices with good energy capacity and computing capabilities. Being costly components, it is of great significance to perform well-planned RNs placement for better management of the deployment and maintenance costs.

The challenging task in this context is how to identify the optimal locations of the RNs considering multiple objectives towards improving WSN performance. This is known as the Relay Node Placement (RNP) problem. A number of architectural networking considerations need to be taken into account when addressing the RNP problem. The placement of RNs can be achieved in a single-tiered network structure with multi-hop topology or a two-tiered network structure with one-hop star topology. RNs can also be placed to ensure sensor nodes connectivity and bi-connectivity using two connectivity models, the connected and survivable models, respectively. It is also possible to carry out anywhere RN placement (unconstrained deployment) or restrict placement areas of the RNs (constrained deployment).

The RNP problem is a Non-Deterministic Polynomial-time (NP)-hard optimization problem. Addressing such a problem using approximate techniques instead of exact techniques is more effective. This is evident considering time complexity as the problem dimension increases. Therefore, metaheuristics can be adopted to efficiently address the RNP problem. A number of metaheuristics exist in the literature and Swarm Intelligence (SI) is one of the common approaches in this context due to its effectiveness. It provides an effective methodology to solve multi-objective problems without prior knowledge of the problem domain. The SI family has different members among of which is the Particle Swarm Optimization (PSO) algorithm. The RNP problem can be addressed with multi-objective formulation using PSO to optimize certain conflicting objectives. These could include network connectivity, coverage, energy efficiency, overall performance, and cost effectiveness.

The RNP problem is addressed in this paper for two-tiered RNs placement with connected and unconstrained deployment strategies. We consider the placement of energy-harvesting RNs in a pre-established WSN having a set of pre-positioned stationary sensor nodes. The RNP problem is formulated as a multi-objective optimization problem and addressed using a novel approach integrating a PSO algorithm with a greedy technique. The focus is on optimizing certain critical WSN objectives, namely network coverage, energy consumption, and deployment cost. The integration of the greedy algorithm provides an effective approach to limit the search space for RNS placement and help PSO to accelerate convergence time. From the evaluation results, the greedy algorithm provided significant information to enhance the optimization process for different PSO variants. In addition, the solution quality and time efficiency were considerably

improved by the proposed optimization approach without adding to the complexity of the system and introducing additional optimization stages. This was more evident in the RNP scenarios of considerably large search spaces, irrespective of their complexities.

In the following section, related research efforts are presented and discussed. In Section 3, the main networking and optimization assumptions considered in this work are presented. Section 4, discusses the mathematical model. The proposed greedy-based PSO approach is described in Section 5. In Section 6, the experimental setup and the evaluation results are presented. The conclusion is provided in Section 7.

2. RELATED WORK

A number of survey papers in the literature discussed varying multi-objective problems in the context of WSNs [1, 2, 3, 4, 5]. The RNP problem is one of such problems which has been widely considered and addressed using different evolutionary algorithms [6, 7, 8, 9]. However, PSO has been a widely adopted optimization algorithm for the RNP problem in addition to other WSN optimization problems such as routing [10, 11], security [12, 13], clustering [14, 15, 16], and mobility [17].

In [18], the RNP problem was addressed using a PSO algorithm incorporating an advanced scheme of particle encoding. It also considered the minimization of the linear combination of two objective functions factoring in the sensor-to-relay nodes distance and hop count for energy efficiency. In [19], a PSO-based model was proposed to address constraint placement of relay nodes in addition to energy-efficient routing for smart grid applications. It enables capturing the restricted area of placement and encoding them into particle-space. A real-to-particle space transformation function was used to transform the areas into a contiguous domain of boundary to fill the particle space of PSO. In [20], the focus was on addressing the RNP problem using a PSO algorithm towards establishing fault tolerance in WSNs with multi-path connectivity. PSO was adopted for finding the minimum number of relay nodes to achieve the required k-connectivity. It was based on establishing a connected network with minimum k-vertex disjoint paths among the different nodes. PSO takes a list of degree deficient nodes and places relay nodes with the objective of maximizing connectivity and minimizing sum of weights for all the nodes in the list. In [21], PSO was applied with a clustering-based approach to address the RNP problem considering energy balancing for large-scale WSNs. Initially, the network is partitioned into multiple clusters using K-mean clustering and Balance clustering algorithms. PSO is then applied to optimize the placement of a relay node towards energy balancing for each cluster. Similarly, PSO was adopted in [22] to address the RNP problem in a clustered WSN with multi-hop topology for tunnel monitoring applications. The focus was on minimizing the distances between relay nodes to optimize energy consumption. In [23], PSO was adopted to find the optimal location to place a sink node with respect to pre-deployed RNs in clustered WSN setups with multi-hop topologies.

There has also been a number of attempts to develop more effective PSO-based approaches by combining PSO with other algorithms and techniques. In [24], PSO was combined with binary search for addressing the RNP problem in addition to optimizing the number of relay nodes in a single run. The binary search mechanism was used to initially fix the upper bound of the search space and then find for the optimal number of RNs. For every potential output, PSO was applied on the selected number of RNs to find their optimal locations. The best solution is determined according to a fitness function considering a single network performance metric. In [25], PSO was enhanced to discretize the RNP problem and integrate local search. It starts by discretizing the continuous RNP problem with a non-linear fitness function before applying discrete PSO with local search. The local search was used to improve the optimization process at the global level.

In a different research work [26], PSO was integrated with the Leave-one-out (LOO) algorithm in a general and customizable RNP optimization framework. The focus was on optimizing network connectivity in multi-hop WSNs considering both constrained and unconstrained strategies. It is based on a multi-stage procedure addressing the RNP problem into three different stages in which different algorithms can be flexibly adopted. LOO was used to select a subset of initially distributed relay nodes and PSO was then applied for further unconstrained placement optimization. Moreover, a PSO approach integrating multiple PSO variants of different optimization features was considered in [27] to address Quality of Service (QoS)-oriented deployment of WSNs. A heterogeneous multi-swarm PSO algorithm incorporating three different PS optimizers was proposed. These are the PSO with inertia weight, PSO with constriction factor and dynamic probabilistic PSO. Each is applied on a different sub-swarm after dividing the population into three sub-swarms. By allowing these sub-swarms to communicate and collaborate, the sub-swarms are able to coevolve. Information from the different optimizers is used in changing the convergence direction of the population.

Other research works focused on addressing optimal placement of relay nodes for achieving network restoration. This was approached in [28] by modeling the RNP problem as a Steiner Minimum Tree (SMT) problem where Steiner points are located and a number of random spanning trees are generated. Then, a PSO variant modifying PSO to address discrete optimization problems with multi-dimensional discrete search space is applied to find the least cost tree. It is based on having a swarm particle jump without specifying its velocity and attracts others once getting good fitness in a certain area. Similarly, the proposed approach in [29] extends PSO to consider variable-dimensional search spaces instead of the fixed number of dimensions considered for the search space in the classical PSO. The focus was on addressing the RNP problem with candidate solutions of varying lengths. Thus it was formulated as a SMT problem considering bounded edge length and the extended-PSO is applied for determining the anchor Steiner points.

PSO was also adopted for the RNP problem in other similar domains such as in wireless body sensor networks [30]. A more dynamic PSO-based approach was also considered in [31] to find the best locations of mobile relay nodes over time in robotic-based MANETs. It enables moving relay nodes to optimal locations to maximize network performance in addition to restore network connectivity in case of dis-connectivity.

On the other hand, RNP optimization in WSNs has been addressed by the research community considering different optimization objectives. Some optimization solutions were developed with a single objective such as maximizing network lifetime [18, 19, 22, 32], enhancing network performance [24, 25, 30], and improving connectivity [26]. However, multi-objective optimization for the RNP problem has received more attention in the literature. In [4, 27], the focus was on maximizing network coverage while minimizing energy consumption. Another example is [31] in which network connectivity was optimized in combination with the overall network performance. RNP Optimization towards effective cost management and maximized network connectivity was also addressed in [33, 34]. Several optimization objectives were also considered such that a limited number of RNs is deployed for improving network lifetime while achieving higher network coverage [35, 36] or maintaining full network connectivity [37, 38]. Maximizing network coverage and connectivity was also targeted in conjunction with minimizing deployment cost [39, 40]. In other proposals, the focus was on improving coverage, lifetime, and connectivity of the network [41, 42].

Table 1. Basic Details of the Experimental Instances.

Ref.	Approach	Objectives	Deployment Strategy	App.	Large-Scale Deployment
[18]	Incorporating an advanced scheme of particle encoding	Network lifetime	Unrestricted	Smart Grid	×
[19]	Real-to-particle space transformation	Network lifetime	Restricted	-	×
[20]	Establishing a connected network with minimum k-vertex disjoint paths	Connectivity & Cost	Unrestricted	-	×
[21]	PSO was integrated with K-mean clustering and Balance clustering algorithms	Energy balancing	Unrestricted	-	√
[22]	Clustering-based approach	Network lifetime	Unrestricted, Multihop	Tunnel monit.	×
[23]	Sink node placement with respect to pre-deployed RNs in clustered WSN	Network lifetime	Unrestricted, Multihop	-	×
[24]	PSO was combined with binary search	Network performance	Unrestricted	-	×
[25]	PSO was enhanced to discretize the RNP problem and integrate local search	Network performance	Unrestricted	-	×
[26]	PSO was integrated with the Leave-one-out (LOO) algorithm	Connectivity	Unrestricted/Restricted, Multihop	-	×
[27]	A heterogeneous multi-swarm PSO algorithm	Network coverage & lifetime	Unrestricted, QoS support	-	×
[28]	A PSO variant for discrete optimization with multi-dimensional discrete search space	Connectivity	Unrestricted	-	×
[29]	Extending PSO to consider variable-dimensional search spaces	Connectivity & Cost	Unrestricted	-	×
Proposed Work	Incorporating a greedy algorithm to guide the PSO algorithm during convergence	Network coverage, lifetime, & cost	Unrestricted	-	√

On the other hand, several studies were conducted to compare the applicability of PSO with other meta-heuristic algorithms for a number of WSN problems [43-46]. These included the Genetic Algorithm (GA) which is an important meta-heuristic technique to solve optimization problems. In [43], the authors found that PSO performs better than GA in forming clusters for wireless sensor networks considering both the computation time for convergence and the quality of the solution on various network topology. It should be noted that both RNP problem and forming clusters in WSN are similar considering the objectives. We found similar conclusions for RNP

problems in comparison with GA and PSO. In this paper, we study the performance of PSO in solving very large RNP problems. To the best of our knowledge, this work is the first attempt that incorporates a greedy algorithm to guide the PSO algorithm during convergence. Such an integration approach has been commonly adopted in different research works for different optimization algorithms. For example, the authors in [47] hybridized a greedy algorithm with a metaheuristic based algorithm called ant colony optimization in finding a near optimal routes for the sensors in multihop based WSN. In [48], the authors propose a location-free greedy algorithm to make routing decision in WSN. In our proposed approach, the greedy algorithm is an essential contributor to the PSO optimization process. First, it provides an estimated number of RNs for PSO so that it can apply fixed-size encoding for a swarm in PSO. The actual number of RNs that will be included for the solution in each swarm is controlled by a set of predefined rules. These rules are flexible and can be controlled by algorithmic parameters so that the PSO can get rid of local optima. Using the output of the greedy algorithm, the search space complexity of the PSO can be effectively limited and the locations for energy-efficient placement of multiple RNs can be efficiently identified. Table 1 provides an overview of the reviewed literature and a comparison with the proposed research work in this paper.

3. ASSUMPTIONS

In this paper, a hierarchical network architecture is considered with a two-tiered WSN structure. It is assumed that a set of stationary battery-powered sensor nodes implemented as small-sized sensor devices in one tier. Each sensor node has short-range low-power low-rate wireless radio which is put to sleep during idle periods. It is also assumed that the sensor nodes are pre-placed in the field with random deployment considering a 2D Euclidean area without any obstacle. A remote monitoring application in which each sensor node periodically collects and transmits sensor data is considered. The communications are made with the RNs at the upper tier of the architecture using unified-sized data packets. Network paths of single hops are used to transmit the packets over the wireless radios using constant transmission power. Once received at the RNs, data is forwarded to the Internet infrastructure via nearby base stations. An ideal network situation of no external interference, collision, nor retransmission is assumed.

At each sensor node, the operation of sensing the application data incur very less power consumption than that of wireless radio communication. Therefore, we only consider the power consumed in sensor data transmission while neglecting sensing power. It is important to note that sensor nodes only transmit sensor data and have no engagement in the operation of data reception. In the case of RNs, they receive and transmit sensor data while having no sensing modules to perform the sensing operation.

Another important assumption is the implementation of the same wireless radio modules with identical transmission and reception ranges for all the sensor and relay nodes. Each base station is assumed to have a sufficient range to interconnect all the placed RNs in the deployment field. The Euclidean distance is used to examine sensor-relay nodes connectivity. Once having the Euclidean distance between two nodes shorter than their communication ranges, it is then assumed that the connectivity is established between them.

The RNs placement is addressed to construct a network topology of single-hop connectivity considering no topological constraints. Adopting single-hop topology for power-constrained sensor devices would be a feasible approach to maintain energy efficiency. On the other hand, placing RNs in varying positions can result in imbalanced distribution of the number of the sensor nodes connected to each RN. As a consequence, traffic load and transmission power consumption will be varied among the RNs. However, we put no emphasis on such

considerations as the RNs are mains-powered or energy-harvesting devices with sufficient computing resources.

4. MATHEMATICAL MODEL

Table 2 list all the notations used throughout this section. Input to the RNP problem is a list of sensor nodes, each of which is described by a point. The list of points is drawn from a 2-D space. We will consider this as the search space associated with the given RNP problem. Note that the unit of our search space is arbitrary. We limit this search space by computing both maximum and minimum values in both X and Y directions. The solution to RNP problem is also a set of points, each of which describes the location of a RN. It should be noted that both input and output data are supposed to be continuous variables. This makes a RNP problem computationally hard as the optimal points can have arbitrary precision during the optimization process. To overcome this situation, we build a mathematical model of our problem which deals with only integers while describing the locations of both sensor nodes and RNs. As we are looking for a near-optimal solution, this conversion does make sense [43].

Let d be a small value such that if a sensor or relay node's location is moved d units in X or Y direction, it causes negligible effect on our near-optimal solution, and the distance between any two sensor nodes is more than d . Let (x_{\min}, y_{\min}) , (x_{\max}, y_{\min}) , (x_{\max}, y_{\max}) , and (x_{\min}, y_{\max}) be the four corners of the smallest rectangle of our search space such that all input points (locations of sensor nodes) are inside of it and the distance from any sensor node to any side of the rectangle is more than d . For simplicity, let the length of each side of the rectangle be a multiple of d . We divide the rectangle by $((x_{\max} - x_{\min})/d - 1)$ straight lines that are parallel to Y and $((y_{\max} - y_{\min})/d - 1)$ straight lines that are parallel to X. These two groups of straight lines will divide the rectangle in such a way that it will have $((x_{\max} - x_{\min})/d) * ((y_{\max} - y_{\min})/d)$ number of squares, each of the squares has area d^2 units. This makes the discretization of any point in the search space straight forward, which can be described as follows: let (x_i, y_i) be a point in our search space, then this point is located in the square marked by a pair of two integers $(\text{floor}(x_i/d), \text{floor}(y_i/d))$.

The above procedure redefines not only the sensor nodes' positions but also the search space itself. Our new search space becomes a grid having $\text{ceil}((x_{\max} - x_{\min})/d)$ and $\text{ceil}((y_{\max} - y_{\min})/d)$ squares in X and Y directions, respectively. In this new search space, our computation involves only integers considering the locations of sensor nodes and RNs.

The output to our integrated approach will be a set of pairs of integers. Each pair defines a particular square in the search space, into which we can place a RN. The actual position of the RN should not affect our near-optimal solution according to the definition of d .

Let $m = \text{ceil}((x_{\max} - x_{\min})/d)$ and $n = \text{ceil}((y_{\max} - y_{\min})/d)$. Now the sensor nodes can be described by S , a zero-one matrix of order m and n , where:

$$S[i][j] = \begin{cases} 1, & \text{if there is a sensor node in } (i, j) \text{ square,} \\ 0, & \text{Otherwise.} \end{cases}$$

s is the total number of sensor nodes such that $\sum_{0 \leq i < m, 0 \leq j < n} S[i][j] = s$.

In addition, Q is a list of s unique pairs of integers, where each pair (i, j) describes a square where a particular sensor is located or $S[i][j] = 1$.

Let w be a value such that a sensor node's signal can travel at best. Let $c = \text{floor}(w/d)$. It means that a sensor can communicate with a RN if and only if the distance between the RN and the

sensor node is at most c square units. In other words, if the nearest RN of a sensor node is c squares away from it in both X or Y directions, the sensor node cannot communicate with the nearest RN.

Table 2. List of Notations.

Symbol	Quantity
S	Sensor Nodes Matrix
s	Total Number of Sensor Nodes
Q	List of sensor nodes
d	The length of square
w	Physical Communication Range
c	Discretized Communication Range
R	RNs Matrix
r	Total Number of RNs
P	A list of pairs of integers that describes a solution to RNPP
max	Upper bound on the number of RNs
R_{greedy}	RN Matrix from greedy algorithm
r_{greedy}	Total RNs by greedy algorithm
$r_{firstKind}$	Number of RNs of first kind
$r_{secondKind}$	Number of RNs of second kind
c_{greedy}	Discretized Communication Range used in greedy algorithm
I	Influence Matrix
$CouterCircle$	Artificial Communication Range for first kind of RN in GA
$m_{outerBox},$ $n_{outerBox}$	Extension of search space to have flexible number of RNs of second kind

R is a zero-one matrix of order m and n to store the positions of RNs, as follows:

$$R[i][j] = \begin{cases} 1, & \text{if there is a RN in } (i, j) \text{ square,} \\ 0, & \text{Otherwise.} \end{cases}$$

Let r be the total number of RNs, which we expect to find by our solution. Then, $\sum_{0 \leq i < m, 0 \leq j < n} R[i][j] = r$.

The output of our solution is P , a list of r unique pairs of integers, where each pair (i, j) describes a square in our search space in the 2-D space and $R[i][j] = 1$. We consider P as a near-optimal solution to RNP problem.

An optimal solution to a RNP problem is to find out the locations for a set of RNs, such that (i) the number of RNs in the set is minimum, (ii) each of the sensor nodes can reach at least one RN, and (iii) the energy consumed by each sensor node is minimized. We will call each of these expected properties as *Cost*, *Coverage* and *Energy*, respectively. We expressed each of these properties by a numerical value as follows:

- (i) *Cost*: the cost associated with the procurement, installment, and maintenance of RNs that are suggested in a candidate solution. Let, max be an integer that we assume to

be an upper bound for the expected number of RNs. So, for any candidate solution, $P_{candidate}$, let $r_{candidate}$ be the number of RNs to solve a RNP problem, where $r_{candidate} \leq max$. We compute cost as the ratio between $r_{candidate}$ and max as follows:

$$f_1 = \frac{r_{candidate}}{max} \quad (1)$$

This ratio is then multiplied by 100 to normalize it with other metrics. This normalized value is further multiplied with a weight factor $w1$.

- (i) *Coverage*: the fraction of sensor nodes that can communicate with at least one RN. This value is further multiplied with a weight factor $w2$. Let s' be the number of sensor nodes that can communicate with at least any RN for a given solution. Then, the network coverage rate is computed as follows:

$$f_2 = \frac{s'}{s} \quad (2)$$

- (ii) *Energy*: Let the maximum distance possible for a sensor node to reach the nearest RN is limited by c squares. Let the sum of distances in square units by the sensors, which has the nearest RN in c square, be *Distance*. We compute energy consumption rate as the ratio between *Distance* and $s*c$ as follows:

$$f_3 = \frac{Distance}{s*c} \quad (3)$$

This ratio is then multiplied by 100 to normalize it. This normalized value is further multiplied with a weight factor $w3$.

The objective value for $P_{candidate}$ is the weighted sum of (1), (2), and (3) as follows:

$$F = w_1 f_1 + w_2 (-f_2) + w_3 f_3 \quad (4)$$

Algorithm 1 OBJECTIVE

Input: $P_{candidate}$, $w1$, $w2$, $w3$, $r_{candidate}$

Output: a real value that is the evaluation of the given candidate solution $P_{candidate}$

Begin

1: total_cost = cost * $r_{candidate}$

2: norm_cost = $w1 * 100 * (total_cost / (max * cost))$

3: covered_sensor = 0

4: total_distance = 0

5: total_dist_covered = 0

6: **for** each pair (i, j) in Q **do**

7: (a, b) = closest($P_{candidate}$, (i, j)) //closest() finds RN at (a,b) from $P_{candidate}$ which is closest to (j,j)

8: total_distance += distance (a, b) to (i, j)

9: **if** distance (a, b) to (i, j) < c **then**

10: ++covered_sensor

11: total_dist_covered += distance (a, b) to (i, j)

12: **end if**

13: **end for**

14: norm_energy = $w3 * 100 * (total_distance / (s * c))$

15: norm_coverage = $w2 * 100 * (-covered_sensor / s)$

16: **return** (norm_energy + norm_cost + norm_coverage)

End

Accordingly, the optimization problem is formulated as a minimization problem. The actual values for the weights $w1$, $w2$ and $w3$ depend on user requirements. For example, we keep $w1 < w2$, if we emphasize more on network coverage than deployment cost. Algorithm 1 describes the above procedure.

5. THE PROPOSED SOLUTION

Our greedy-based PSO solution has two main steps. In the first step, we apply a greedy algorithm to find out some squares on the search space in which we observe signals from a maximum number of sensor nodes. In other words, the greedy algorithm tries to place RNs to reduce the energy consumption by sensor nodes. As the greedy algorithm is used to accelerate the convergence in the PSO, we are not looking for any complete solution from it. Once the greedy algorithm returns the list of squares, in which RNs are in need, we instrument and execute the PSO with the help of that list of squares. The following subsections provide detailed descriptions of these two main steps: the greedy algorithm and greedy-based PSO. Figure 1 provides an architectural overview of the proposed PSO-based solution.

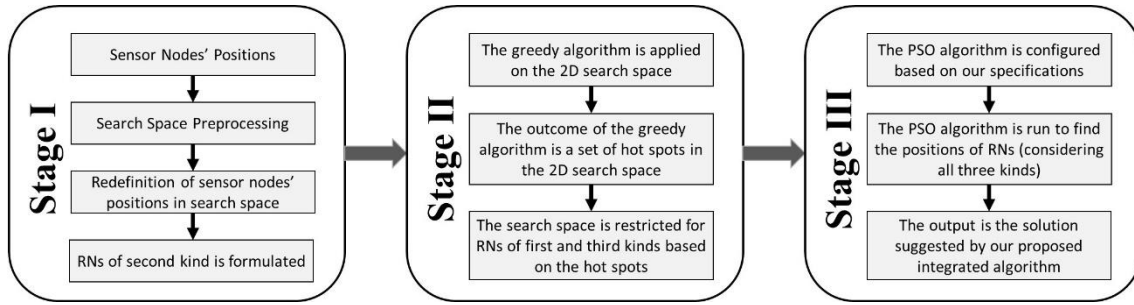


Figure 1. An Architectural Overview of the Proposed PSO-based Solution

5.1. The Greedy Algorithm

We need to list the following notations in order to explain the working principle of our greedy algorithm. Let R_{greedy} be a zero-one matrix of order m and n , where:

$$R_{greedy}[i][j] = \begin{cases} 1, & \text{if GREEDY puts a RN in square } (i, j), \\ 0, & \text{Otherwise.} \end{cases}$$

Let r_{greedy} be the total number of RNs that our greedy algorithm suggests to be placed across the search space. It is clear that $\sum_{0 \leq i < m, 0 \leq j < n} R_{greedy}[i][j] = r_{greedy}$.

In the greedy algorithm, we assume that each sensor node can connect to a RN if the distance between them is equal or smaller than c_{greedy} squares. c_{greedy} is an artificial value such that $c_{greedy} > c$. It is expected that r_{greedy} is inversely proportional to the value of c_{greedy} .

Our greedy algorithm is explained with the ComputingInfluence and GREEDY algorithm as presented in Algorithm 2 and 3, respectively. The input to ComputingInfluence is a set of sensor nodes, G . The output of this algorithm is I , an integer matrix of order m and n . $I[i][j]$ is the number of sensor nodes from G whose signal can be reached at square (i, j) assuming that the signal from each sensor node can be propagated c_{greedy} squares in all directions.

The GREEDY algorithm is an iterative algorithm. The input to this algorithm is Q , which is the set of all sensor nodes. In each iteration, it computes I using ComputingInfluence. It then finds the maximum value in I . Let $I[i][j]$ contains maximum value, it then updates with $R[i][j] = 1$ and deletes the sensor nodes which could be served by placing a RN at (i, j) before starting the next iteration (if any).

5.2. The Greedy-Based PSO Algorithm

The PSO algorithm is executed once the greedy algorithm has completed its execution. Two very important pieces of information are obtained from the greedy algorithm:

- (i) the approximate number of RNs that can be used to serve all sensor nodes
- (ii) the squares in the search space where RNs are more likely to be placed

These two outputs from the greedy algorithm will be used to effectively configure the PSO to solve the RNP problem.

Algorithm 3 GREEDY

Input: Set Q

Output: R_{greedy}

Begin

```

1: Set  $Q_I = Q$ 
2: Create a zero matrix  $R_{greedy}$  of order  $m$  and  $n$ 
3: While  $Q_I \neq \text{null}$  do
4:    $I = \text{ComputingInfluence}(Q_I)$ 
5:    $I(i,j) = \max(I)$ 
6:    $R_{greedy}[i][j] = 1$ 
7:   for each sensor  $g$  in  $Q_I$ ; where  $g$  is at  $(x, y)$  do
8:     if distance from  $(x, y)$  to  $(i, j) \leq c_{greedy}$  then
9:       delete  $g$  from  $Q_I$ 
10:    end if
11:  end for
12: end while
13: return  $R_{greedy}$ 

```

End

Algorithm 2 ComputingInfluence

Input: Set G

Output: Matrix I of order m and n

Begin

```

1: Create a zero matrix  $I$  of order  $m$  and  $n$ 
2: for each sensor  $g$  in  $G$ ; where  $g$  is at  $(i, j)$  do
3:   for each square  $(x, y)$  in  $I$  do
4:     if distance from  $(i, j)$  to  $(x, y) \leq c_{greedy}$  then
5:        $I[x][y] = I[x][y] + 1$ 
6:     end if
7:   end for
8: end for
9: return  $I$ 

```

End

From the PSO algorithm, three kinds of RNs are expected:

- i. $r_{\text{firstKind}}$: First Kind RNs are the nodes that are being tried to place near to the squares suggested by the greedy algorithm. To describe this kind, we need to introduce another artificial communication range $c_{\text{outerCircle}}$, where $c_{greedy} > c_{\text{outerCircle}} > c$. In total, PSO will try to place r_{greedy} number of RNs of this kind. The range of each of the RN's coordinate

values is by both adding and subtracting c_{greedy} with the corresponding coordinate positions from greedy algorithm. The main purpose of these RNs is to increase the coverage metric. During the convergence phase of PSO, a RN will be considered for fitness value if it is, by random choice, not more than $c_{outerCircle}$ away from the position suggested by the greedy algorithm. This technique gives us the flexibility for our PSO on the number of RNs. We use $r_{firstKind}$, to represent the total number of RNs of first kind.

- ii. $r_{secondKind}$: Second Kind RNs are the nodes that are being tried to place anywhere in the search space. To describe this kind of RNs, we need to introduce $m_{outerBox}$, $n_{outerBox}$, where $m_{outerBox} > m$ and $n_{outerBox} > n$. The PSO will try to place a number of RNs of second kind. The range of coordinates of each RN is defined as $(0, m_{outerBox})$ and $(0, n_{outerBox})$. The main purpose of these RNs is to get rid of local optima that might be caused by those of first kind. The number of RNs of second kind will depend on the complexity of RNP problem. If the solution from the PSO algorithm is not satisfactory considering energy consumption and network coverage, it can be increased. During the convergence phase of the PSO, a RN will be considered for fitness value if it is, by random choice, placed in the legal boundary of the search space. This technique gives us the flexibility for our PSO on the number of RNs. We use $r_{secondKind}$, to represent the total number of RNs of second kind.
- iii. $r_{thirdKind}$: Third Kind RNs are exactly like first kind with one exception. For each square, that is suggested by the greedy algorithm, we can try to put any number of relay nodes of third kind. For some of those squares, we might not try for any relay node of third kind. On the other hand, for some of those squares, we might try for more than one relay node. The number of relay nodes in each of those squares, proportionally depends on the value computed for the corresponding square from ComputingInfluence algorithm. The main purpose of these relay nodes is to improve the energy value in the RNP problem.

In our PSO code, the number of decision variables is equal to the sum of RNs of first, second and third kind. As we can see these different kinds of RNs can be defined by applying lower and upper bound value constraints. We kept the value for social and self-adjustment weight small (around 1.0) for our experimentations. These two values are used to calculate the velocity of the swarms by considering the swarm's own best solution and the best solution by any swarm.

6. EVALUATION

6.1. Experimental Setups

The evaluation of the proposed greedy-based PSO algorithm was carried out using an experimental WSN dataset representing varying RNP problem scenarios of fairly complex setups. The dataset contains four artificial WSN instances with randomly distributed stationary sensor nodes. The instances had deployment areas of square shapes with different dimensions and numbers of sensor nodes as illustrated in Table 3. We made sure that the proposed approach was tested under a varied set of RNP scenarios, from fairly small- to large-scale. We deemed it important to include the sufficient number of sensor nodes to cover the entire deployment area in each scenario. For all the sensor nodes and RNs in all the scenarios, w and d were set to 40 and 4m, respectively [44, 45].

The implementation and evaluation of the proposed algorithm was accomplished using MATLAB R2018b on a PC having an Intel Core i5, 2.5 GHz CPU, 4 GB RAM, and Mac OS. Similar PSO configurations were set for all the experiments. However, the initial population varies for all the scenarios and increases as the instance size increases. In each experiment, the greedy algorithm provides the PSO process with the number of RNs. For the proposed greedy-

based PSO, the number of RNs that the solution will try to place is the sum of $r_{firstKind}$ and $r_{secondKind}$. Each $r_{firstKind}$ has a domain of defined by the corresponding RN from greedy algorithm and $C_{outerCircle}$ whereas $r_{secondKind}$'s domain is defined as $(0, m_{outerBox})$ and $(0, n_{outerBox})$. Any non-zero positive value can be assigned to the weight factors, W_1 , W_2 and W_3 . For the current implementation, the network coverage was deemed the most important objective while energy consumption was considered more important than the deployment cost. Therefore, the weights were set to have $W_2 > W_3 > W_1$.

Considering the randomness effect in such optimization techniques, 20 independent runs were conducted for each instance in all the experiments. The average of the collected results was taken for each experiment. It is important to note that a confidence level of 95% was considered.

Table 3. Basic Details of the Experimental Instances.

Scenario	Area Size (m2)	s
INS1	300×300	150
INS2	600×600	300
INS3	800×800	400
INS4	1000×1000	500

Performance evaluation of the proposed greedy-based PSO algorithm was assessed considering two different aspects. Firstly, it was compared with the performance of the classical PSO algorithm. A further comparison was also established with an enhanced PSO version integrating the Pattern Search (PS) optimizer. We add "hybrid" in a PSO algorithm to refer the integration of pattern search algorithm with it. This gave more insights into the performance of the proposed approach and allowed for a further and deeper investigation. As in many similar studies [16, 17, 40], the main considered comparison criteria were solution quality and time efficiency. The solution quality was measured by the values of the main objectives: network coverage, energy consumption, and deployment cost. For the time efficiency, a machine-independent measure, which was the total number of generations required by the algorithm before code termination, was considered. The second assessment aspect was studying the impact of network area size and number of sensor nodes on the performance of the proposed approach. This is important to understand to what extent the greedy-based PSO algorithm would be affected by the different deployment setups.

6.2. Results and Discussion

For the small-scale scenario INS1, all the considered variants of PSO achieved similar performance in terms of network coverage and energy consumption with relatively high solution quality as presented in Tables 4 and 5. The classical PSO was able to cover more than 94% of the sensor nodes and the Greedy-based PSO increases the coverage rate to more than 96%. However, the classical PSO required relatively less number of generations to maintain such high performance as indicated by the results in Figure 2.

In general, the performance of all the variants degraded as the network size increased in INS1-INS3. However, this was more adverse in the case of the classical PSO as the coverage rate decreased by more than 22% compared to less than 10% for the greedy-based PSO and hybridized greedy-based PSO. In terms of energy consumption, less degradation was experienced by greedy-based and hybridized PSO while the network became wider and denser. On the other

hand, all the variants achieved quite a similar performance in INS3 and INS4 despite the increase in the size of INS4.

The classical PSO performed inefficiently in the large-scale scenarios (INS2-INS4) providing a coverage rate of less than 80% and energy consumption rate of more than 74%. The greedy-based PSO improved such results by increasing the network coverage rate and reducing the energy consumption rate. For example, in Scenario INS4 with a very large setup, the rates were improved by more than 17% and 6%, respectively. In regards to time efficiency, the proposed algorithm tended to converge faster than the classical PSO in the large-scale setups as the results indicated in Figure 2. For example, it required about 13% less iterations for converging to a better solution in scenario INS4.

Another comparison for the number of RNs was established between the classical PSO and the greedy-based PSO as presented in Figure 3. It can be seen that the greedy-based PSO did not add much improvement to the number of RNs in the small-scale scenario of INS1. That is, the classical PSO required only one additional RN to provide the same performance of the greedy-based PSO in that case. However, the number of RNs was reduced by the greedy-based PSO as the network scaled up. It was able to provide a cost-effective deployment solution with high solution quality using less number of RNs by more than 15%.

Table 4. Network Coverage Results.

Scenario	Classical-PSO		PS-Hybridized PSO		Greedy-Based PSO		PS-Hybridized Greedy-PSO	
	Mean	STD	Mean	STD	Mean	STD	Mean	STD
INS1	94.0	±3.0	97.0	±2.1	96.7	±0.9	97.5	±1.2
INS2	79.4	±3.5	95.8	±1.4	91.2	±1.5	95.0	±1.6
INS3	73.7	±3.2	93.1	±1.7	88.2	±1.4	92.3	±1.3
INS4	73.4	±3.6	94.5	±1.7	87.3	±1.4	92.8	±1.6

Table 5. Energy Consumption Results.

Scenario	Classical-PSO		PS-Hybridized PSO		Greedy-Based PSO		PS-Hybridized Greedy-PSO	
	Mean	STD	Mean	STD	Mean	STD	Mean	STD
INS1	63.7	±3.7	57.8	±1.5	61.4	±1.6	57.3	±1.1
INS2	74.4	±1.7	62.1	±3.0	69.5	±1.3	64.6	±3.0
INS3	76.9	±1.8	67.8	±4.5	72.4	±0.6	71.1	±2.6
INS4	77.0	±1.8	63.4	±4.7	72.2	±0.9	69.1	±5.0

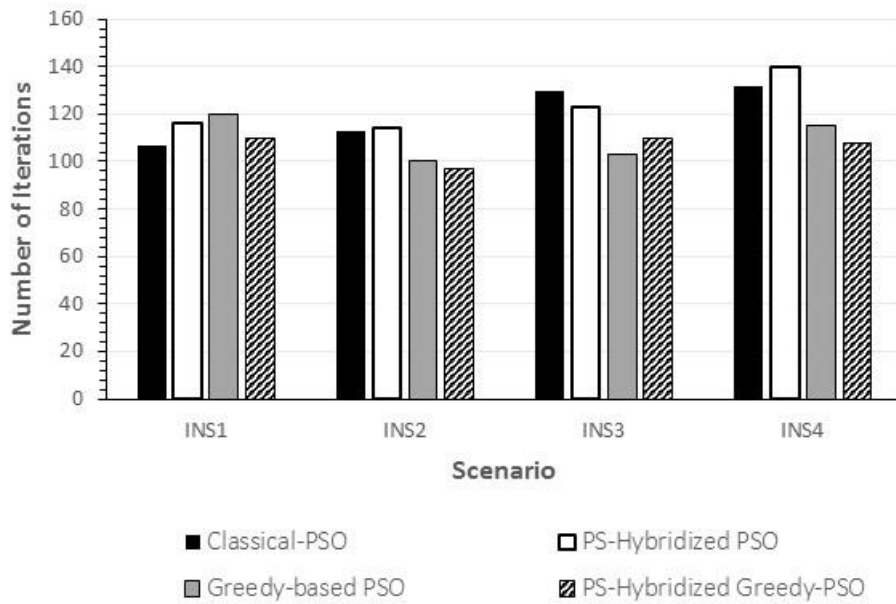


Figure 2. Number of Generation Comparison

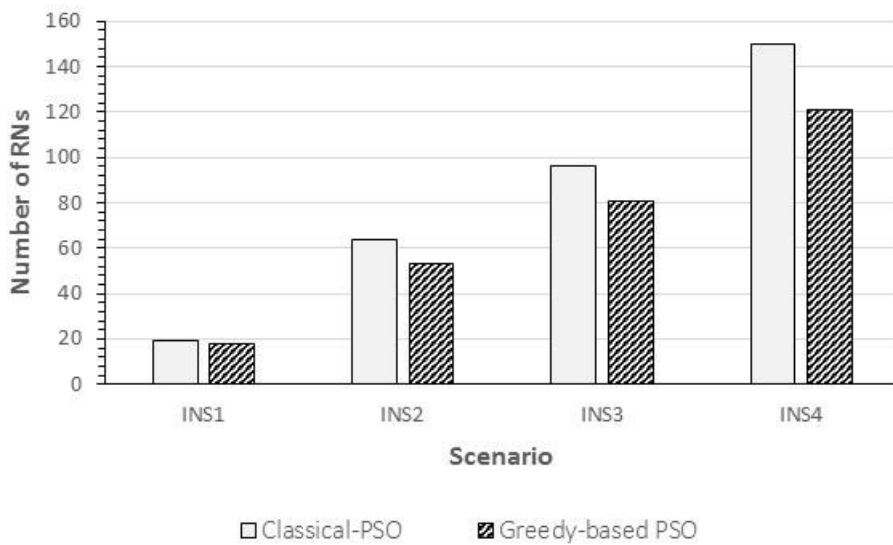


Figure 3. Number of RNs Comparison

In a further investigation, we compared the improvement provided by the classical PSO and the greedy-based PSO with other PSO variants hybridizing PSO with a further optimization technique. This was done by applying the pattern search optimizer on the PSO output in a second optimization stage for the classical PSO and the greedy-based PSO separately. As the results indicated in Tables 4 and 5, the hybridized greedy-based PSO improved network coverage and energy consumption of the greedy-based PSO by about 4% and 5% on an average, respectively. Higher increase in the network coverage and energy consumption rates by 19% and 14% on an average, respectively, was experienced once the pattern search optimization incorporated into the classical PSO. By comparing these performance gaps introduced by the hybrid approach in both cases, it is evident that the greedy-based PSO did not leave much room for significant improvement by additional optimizers. It was able to provide highly optimized results using a

simple integration technique without adding to the complexity of the system and introducing additional optimization stages.

It is also important to notice the high solution quality provided by the classical PSO when hybridized with the pattern search optimizer as indicated by the results in Tables 4 and 5. It noticeably improved the performance of the classical PSO and achieved a high network coverage rate of more than 94% and low energy consumption rate of less than 68% in all the scenarios. However, such an improvement was a result of a significant help of the greedy algorithm providing the best number of RNs to effectively start the optimization process. In Table 6, the performance of the hybridized PSO is presented for different numbers of RNs less than the greedy algorithm's output by 40-10% (represented as x). It can be seen that the performance improved as the number of RNs came closer to the greedy value. For example, when taking only 60% of the number of RNs given by the greedy algorithm ($x=0.6$, RNs=32), the PSO was unable to maintain the high solution quality even with the pattern search optimizer. On the other hand, when the optimization process stopped with 43 RNs ($x=0.8$), the pattern search optimizer concluded by covering about 87% of the sensor nodes with an energy consumption rate of about 74%. However, by knowing the near optimal number of RNs, it concluded by covering about 95% of the sensor nodes with an energy consumption rate of about 62% as presented in Table 4 and 5 for Scenario INS2. In the former case, if we add more RNs by a separate optimization algorithm to have all the sensor nodes covered, it is very unlikely that the resulting energy consumption is so little to make averaged energy consumption comparable with the latter case. This provides a good indication of the significance of our greedy algorithm in enhancing the optimization process with critical information regarding the near-optimal number of RNs at the initial stage.

Table 6. Results of the PS-Hybridized PSO for INS2 with Different Numbers of RNs.

X	RNs	Network Coverage Rate	Energy Consumption Rate	Number of Iterations
0.6	38	82	76.62	112
0.7	43	87.02	73.72	118
0.8	48	92.05	70.74	112
0.9	53	95.8	62.13	114
1	32	74.12	80.21	109

Moreover, we studied the impact of the complexity of the scenario on the performance of the greedy-based PSO compared to the classical PSO. As presented in Table 7, different scenarios of varying levels of network density (ranging from 500-50 sensor nodes) were considered for the large-scale scenario of INS4. For each case, a different number of RNs proportional to the number of sensor nodes was considered by the algorithms.

The results presented in Table 7 shows that the lower the network density the higher the improvements provided by the greedy-based PSO for network coverage and energy consumption. Comparing the results for the scenario INS4 with 500 and 50 sensor nodes, such improvements increased by more than 52 and 16 percentage points, respectively. Noticeably, the greedy-based PSO performed relatively better as the scenario became more complex whereas the classical PSO failed to maintain satisfactory performance in cases of lower density levels.

Table 7. Impact of the Problem Complexity on the performance of the greedy-based PSO Considering INS4.

No. of Sensor Nodes	RNs	Network Coverage Rate			Energy Consumption Rate			No. of Iterations		
		Classical-PSO	Greedy-PSO	Improvement	Classical-PSO	Greedy-PSO	Improvement	Classical-PSO	Greedy-PSO	Improvement
500	121	74.43	87.28	17.26%	76.99	72.18	6.25%	132	115	12.88%
400	105	70.06	87.51	24.91%	79.03	72.07	8.81%	122	108	11.48%
300	93	68.47	86.73	26.67%	80.42	73	9.23%	134	104	22.39%
200	77	67.35	89.35	32.67%	81.07	70.87	12.58%	112	89	20.54%
100	52	65.1	91.9	41.17%	82.08	67.73	17.48%	105	94	10.48%
50	30	52.3	89	70.17%	88.06	67.76	23.05%	84	72	14.26%

It can also be seen that the performance variation experienced by the classical PSO across the different cases was relatively very high. Considering the scenarios with 500 and 50 sensor nodes, the network coverage rate decreased by almost 30% and the energy consumption rate increased by about 14%. On the other hand, the greedy-based PSO was able to maintain its high performance with very low performance variation across the different scenarios, despite their complexity levels.

In all the experiments, we configured the algorithms to terminate once the relative changes of objective values for a fixed number of iterations do not exceed a predefined small value. We expect that the PSO should find out the positions of the RNs before it terminates. In this experiment, we found that the classical PSO does not demonstrate the ability to explore the search space for finding the suitable places for RNs to cover a significant percentage of the sensor nodes before reaching the terminating condition. Whereas, the greedy-based PSO demonstrates a better performance and the difference in their iteration numbers was not significant. That means that the classical PSO reached the terminating condition without exploring the search space well enough. For example, in the scenario with 50 sensor nodes in the 1000*1000 search space, it was only able to cover about 52% of the sensor nodes by 84 iterations before reaching the terminating condition as can be seen in Table 7. For the greedy-based PSO to cover 89% of the sensor nodes, however, it required 14% less iterations before reaching the terminating condition.

7. CONCLUSION

The RNP problem considering unconstrained RNs deployment into a pre-established stationary WSN was addressed in this paper based on a novel integration of the PSO algorithm and a greedy technique. The proposed integration enables effective initiation of the optimization process with significant information allowing better guidance of the PSO algorithm during its convergence. As the evaluation results demonstrated, such a critical contribution of the greedy strategy played a significant role in enhancing the overall performance of the PSO algorithm even when hybridized

with other optimizers. In addition, the proposed integration provided a simple approach to improve solution quality and time efficiency without the need for introducing additional optimization stages. As the search spaces became larger and the setups got more complex, the proposed greedy-based PSO maintained its performance with relatively high network coverage and low energy consumption in addition to enabling cost-effective deployment. In our future work, the focus will be on addressing other variants of the RNP problem. Other aspects such as k-connectivity establishment, multihop WSN topologies, 3D search spaces, constrained deployment, and heterogeneous setups will be considered.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

ACKNOWLEDGEMENTS

This work was supported by Deanship of Scientific Research, Qassim University, according to the agreement of the funded project No. SRD-ucc-2018-1-14-S-5126. The author thanks the sponsor of this work for their support.

REFERENCES

- [1] Z. Fei, B. Li, S. Yang, C. Xing, H. Chen, and L. Hanzo, "A survey of multi-objective optimization in wireless sensor networks: Metrics, algorithms, and open problems," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 550-586, 2016.
- [2] D.S. Deifand Y. Gadallah, "Classification of wireless sensor networks deployment techniques," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 2, pp. 834-855, 2014.
- [3] M. Iqbal, M. Naeem, A. Anpalagan, N.N. Qadri, and M. Imran, "Multi-objective optimization in sensor networks: Optimization classification, applications and solution approaches," *Comput. Networks.*, vol. 99, pp. 134-161, 2016.
- [4] S. Harizanand P. Kuila, "Evolutionary algorithms for coverage and connectivity problems in wireless sensor networks: a study," In *Design Frameworks for Wireless Networks*, Springer, Singapore, pp. 257-280, 2020.
- [5] V.W. Morais, F.S. Souza, and G.R. Mateus, "Optimization Problems, Models, and Heuristics in Wireless Sensor Networks," 2018.
- [6] J.M. Lanza-Gutiérrez, N. Caballé, J.A. Gómez-Pulido, B. Crawford, and R. Soto, "Toward Robust Multi-Objective Metaheuristic for Solving the Relay Node Placement Problem in Wireless Sensor Networks," *Sensors*, vol. 19, no. 3, pp. 677, 2019.
- [7] J.M. Lanza-Gutiérrez 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.*, vol. 30, pp. 675-687, 2015.
- [8] B.O. Ayinde and H.A. Hashim, "Energy-efficient deployment of relay nodes in wireless sensor networks using evolutionary techniques," *Int. J. Wirel. Inf. Networks.*, vol. 25, no. 2, pp. 157-172, 2018.
- [9] J.M. Lanza-Gutiérrez, N. Caballé, J.A. Gómez-Pulido, B. Crawford, and R. Soto, "Toward a robust multi-objective metaheuristic for solving the relay node placement problem in wireless sensor networks," *Sensors*, vol. 19, no. 3, pp. 677, 2019.
- [10] S. Tabibi, and A. Ghaffari, "Energy-efficient routing mechanism for mobile sink in wireless sensor networks using particle swarm optimization algorithm," *Wirel. Pers. Commun.*, vol. 104, no. 1, pp. 199-216, 2019.
- [11] D.R. Edla, M.C. Kongara, and R. Cheruku, "A PSO based routing with novel fitness function for improving lifetime of WSNs," *Wirel. Pers. Commun.*, vol. 104, no. 1, pp. 73-89, 2019.
- [12] S. Radha, S.A. Nandhini, and R. Hemalatha, "Efficient Anomaly Detection System for Video Surveillance Application in WWSN with Particle Swarm Optimization," In *Computational Intelligence in Wireless Sensor Networks*, Springer, Cham, pp. 153-177, 2017.

- [13] P. Ranjan and H. Om, "Computational Intelligence Based Security in Wireless Sensor Networks: Technologies and Design Challenges," In *Computational Intelligence in Wireless Sensor Networks*, Springer, Cham, pp. 131-151, 2017.
- [14] K. Selvakumar and R.M. Arieth, "Contrast for QOS based clustered energy efficient protocol with PSO and multi-hop gateways in wireless sensor network," *Cluster Comput.*, vol. 22, no. 5, pp. 11883-11890, 2019.
- [15] X. Wang, H. Gu, Y. Liu, and H. Zhang, "A two-stage RPSO-ACS based protocol: A new method for sensor network clustering and routing in mobile computing," *IEEE Access*, vol. 7, pp. 113141-113150, 2019.
- [16] T. Kaur and D. Kumar, "Particle swarm optimization-based unequal and fault tolerant clustering protocol for wireless sensor networks," *IEEE Sensors J.*, vol. 18, no. 11, pp. 4614-4622, 2018.
- [17] J.R. Parvin and C. Vasanthanayaki, "Particle swarm optimization-based energy efficient target tracking in wireless sensor network," *Measurement*, vol. 147, pp. 106882, 2019.
- [18] H. Banka & P.K. Jana, "PSO-based multiple-sink placement algorithm for protracting the lifetime of wireless sensor networks," *Proc. of the second international conference on computer and communication technologies*, Springer, New Delhi, pp. 605-616, 2016.
- [19] T.N. Mathaba, "Optimal Sink-Node Placement and Routing for an Energy Efficient Two-Tier Wireless Sensor Network," In *2018 International Conference on Advances in Big Data, Computing and Data Communication Systems, IEEE*, pp. 1-6, Aug. 2018.
- [20] D.R. Dandekar and P.R. Deshmukh, "Relay node placement for multi-path connectivity in heterogeneous wireless sensor networks," *Procedia Technol.*, vol. 4, pp. 732-736, 2012.
- [21] D.R. Dandekar and P.R. Deshmukh, "Energy balancing multiple sink optimal deployment in multi-hop wireless sensor networks," In *2013 3rd IEEE International Advance Computing Conference*, pp. 408-412, Feb. 2013.
- [22] B. Yu, W. Yuanping, Z. Liang, H. Yuan, and Z. Aijuan, "Relay node deployment for wireless sensor networks based on PSO," In *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, pp. 2393-2398, Oct. 2015.
- [23] M.N. Rahman & M.A. Matin, "Efficient algorithm for prolonging network lifetime of wireless sensor networks," *Tsinghua Sci. Technol.*, vol. 16, no. 6, pp. 561-568, 2011.
- [24] C.N. Nyirinda and S.G. Nyirongo, "Binary Search Based PSO for Master Node Enumeration and Placement in a Smart Water Metering Network," In *International Conference on e-Infrastructure and e-Services for Developing Countries*, Springer, Cham, pp. 104-118, Dec. 2019.
- [25] H. Safa, W. El-Hajj, and H. Zoubian, "Particle swarm optimization based approach to solve the multiple sink placement problem in WSNs," In *2012 IEEE International Conference on Communications*, pp. 5445-5450, Jun. 2012.
- [26] R. Magán-Carrión, R.A. Rodríguez-Gómez, J. Camacho, and P. García-Teodoro, "Optimal relay placement in multi-hop wireless networks," *Ad Hoc Networks*, vol. 46, pp. 23-36, 2016.
- [27] Q. Ni, H. Du, Q. Pan, C. Cao, and Y. Zhai, "An improved dynamic deployment method for wireless sensor network based on multi-swarm particle swarm optimization," *Nat. Comput.*, vol. 16, no. 1, pp. 5-13, 2017.
- [28] R. Sharma and V. Ranga, "Federating Disjoint Segments in Wireless Sensor Networks Using Jumping Particle Swarm Optimization," In *International Conference on Emerging Research in Computing, Information, Communication and Applications*, Springer, Singapore, pp. 559-568, Jul. 2016.
- [29] Y. Xu, Y. Xiao, and Q. Sun, "A Swarm-Based Meta-Heuristic for Relay Nodes Placement in Wireless Sensor Networks," *Int. J. Innov. Comput. Inf. Control.*, vol. 15, no. 2, pp. 551-567, 2019.
- [30] T.Y. Wu and C.H. Lin, "Low-SAR path discovery by particle swarm optimization algorithm in wireless body area networks," *IEEE Sensors J.*, vol. 15, no. 2, pp. 928-936, 2014.
- [31] R. Magán-Carrión, J. Camacho, P. García-Teodoro, E.F. Flushing, and G.A. Di Caro, "Drns: Dynamical Relay Node Placement Solution," In *International Conference on Practical Applications of Agents and Multi-Agent Systems*, Springer, Cham, pp. 273-276, Jun. 2016.
- [32] N.T. Tam, H.T.T. Binh, T.H. Hung, and D.A. Dung, "Prolong the Network Lifetime of Wireless Underground Sensor Networks by Optimal Relay Node Placement," In *International Conference on the Applications of Evolutionary Computation (Part of EvoStar)*, Springer, Cham pp. 439-453, Apr. 2019.

- [33] S. Sapre and S. Mini, "Optimized relay nodes positioning to achieve full connectivity in wireless sensor networks," *Wirel. Pers. Commun.*, vol. 99, no. 4, pp. 1521-1540, 2018.
- [34] Y. Li, C.S. Chen, K. Chi, and J. Zhang, "Two-tiered relay node placement for WSN-based home health monitoring system," *Peer-to-Peer Netw. Appl.*, vol. 12, no. 3, pp. 589-603, 2019.
- [35] D. Djenouria and M. Bagaa, "Energy harvesting aware relay node addition for power-efficient coverage in wireless sensor networks," In *2015 IEEE International Conference on Communications (ICC)*, IEEE, pp. 86-91, Jun. 2015.
- [36] J.M. Lanza-Gutierrez and J.A. Gomez-Pulido (2017). "A gravitational search algorithm for solving the relay node placement problem in wireless sensor networks," *Int. J. Commun. Syst.*, vol. 30, no. 2, e2957, 2017.
- [37] B.O. Ayinde and H.A. Hashim, "Energy-efficient deployment of relay nodes in wireless sensor networks using evolutionary techniques," *Int. J. Wirel. Inf. Networks.*, vol. 25, no. 2, pp. 157-172, 2018.
- [38] D. Djenouria and M. Bagaa, "Energy-aware constrained relay node deployment for sustainable wireless sensor networks," *IEEE Trans. Sustain. Comput.*, vol. 2, no. 1, pp. 30-42, 2017.
- [39] M.A. Benatia, M.H. Sahnoun, D. Baudry, A. Louis, A. El-Hami, and B. Mazari, "Multi-objective WSN deployment using genetic algorithms under cost, coverage, and connectivity constraints," *Wirel. Pers. Commun.*, vol. 94, no. 4, pp. 2739-2768, 2017.
- [40] K. Nitesh and P.K. Jana, "Relay node placement with assured coverage and connectivity: a Jarvis March approach," *Wirel. Pers. Commun.*, vol. 98, no. 1, pp. 1361-1381, 2018.
- [41] J.M. Lanza-Gutiérrez, J.A.G. Pulido, and M.A. Vega-Rodríguez, "A New Realistic Approach for the Relay Node Placement Problem in Wireless Sensor Networks by Means of Evolutionary Computation," *Ad Hoc Sens. Wirel. Networks.*, vol. 26, no. 1-4, pp. 193-209, 2015.
- [42] H. Zhang, Z. Zhang, F. Zhang, L. Li, and Y. Wang, "Optimized design of relay node placement for industrial wireless network," *Int. J. Distrib. Sens. Networks.*, vol. 10, no. 11, pp. 759826, 2014.
- [43] P. Parwekar, S. Rodda, and S. Vani Mounika, "Comparison between Genetic Algorithm and PSO for Wireless Sensor Networks," *Smart Computing and Informatics—Smart Innovation, Systems and Technologies*, S. Satapathy, V. Bhateja, and S. Das, eds., Springer, Singapore, vol 77, 2018.
- [44] Cazacu, Razvan. "Comparison between the performance of GA and PSO in structural optimization problems," *Am. J. Eng. Res.*, vol. 5, no. 11, pp. 268-272, 2016.
- [45] Lambert-Torres, Germano, et al. "Comparison between PSO and GA in system restoration solution," *2009 15th International Conference on Intelligent System Applications to Power Systems*. IEEE, 2009.
- [46] Y. Yoon and Y. Kim, "An Efficient Genetic Algorithm for Maximum Coverage Deployment in Wireless Sensor Networks," *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1473-1483, Oct. 2013, doi: 10.1109/TCYB.2013.2250955.
- [47] M. Tewari and K. S. Vaisla, "Optimized hybrid ant colony and greedy algorithm technique based load balancing for energy conservation in WSN," *International Journal of Computer Applications*, vol. 104, no. 17, pp. 14-18, Oct 2014.
- [48] A. A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura and A. A. F. Loureiro, "A Novel Location-Free Greedy Forward Algorithm for Wireless Sensor Networks," In *IEEE International Conference on Communications*, Beijing, pp. 2096-2101, 2008. doi: 10.1109/ICC.2008.402.

AUTHORS

Bader Alshaqqawi received the B.S. in mathematics from Qassim University, Buraydah, KSA in 2004 and the M.S. degree in applied mathematics from University of Manchester, Manchester, UK, in 2010. He is currently a lecturer in the department of mathematics, collage of science, Qassim University, Buraydah, KSA. His research interests in the area of Applied Mathematics including Numerical Analysis, Optimization, Algorithms for NP-hard problems.



Sardar Anisul Haque achieved his Ph.D. in Computer Science from the University of Western Ontario, Canada in 2014. He is working as an assistant professor in computer science at Alcorn State University. He is a former faculty member of Qassim University, Saudi Arabia and Islamic University of Technology, Bangladesh. His research Interest is in high performance computing, GPU programming, and Algorithms for NP-hard problems.



Mohammed Alreshoodi received the B.S. degree in computer science from Qassim University, Buraydah, KSA, in 2004 and the M.S degree in computer networks from University of Essex, Colchester, UK, in 2011. He also received the Ph.D. degree in in computer networks from University of Essex, Colchester, UK, in 2011. He is currently an Associate Professor in the department of applied science in Unizah community college at Qassim University, KSA. His research interest includes computer networking, wireless networks, WSN, IoT, networks security. He is a member of WSN research group and Cyber Security research group at Collage of Computer, Qassim University, KSA.



Ibrahim S. Alsukayti received the B.S. degree in Computer Science, from Qassim University, Buraydah, KSA, in 2006. He then received the M.S. degree in Computer and Information Networks from University of Essex, Colchester, UK, in 2010 and the Ph.D. in Computer Networks from Lancaster University, Lancaster, UK, in 2014. Currently, he is an Associate Professor with the Computer Science Department, College of Computer, Qassim University, Buraydah, KSA. He is also a team member of two ongoing funded research projects and the director of a research group targeting Internet of Things technologies & applications. His research interests include network routing, Networks, networking protocols, network security, and IoT.



Wireless Sensor