electronics

MDPI

Article

Particle Swarm Optimization for *k*-Coverage and 1-Connectivity in Wireless Sensor Networks

Georgios Siamantas and Dionisis Kandris *

microSENSES Research Laboratory, Department of Electrical and Electronics Engineering, Faculty of Engineering, University of West Attica, Ancient Olive Grove Campus, 250, Thivon Avenue, 12241 Athens, Greece; gsiamantas@uniwa.gr
* Correspondence: dkandris@uniwa.gr

Abstract: Wireless Sensor Networks are used in an ever-increasing range of applications, thanks to their ability to monitor and transmit data related to ambient conditions in almost any area of interest. The optimization of coverage and the assurance of connectivity are fundamental for the efficiency and consistency of Wireless Sensor Networks. Optimal coverage guarantees that all points in the field of interest are monitored, while the assurance of the connectivity of the network nodes assures that the gathered data are reliably transferred among the nodes and the base station. In this research article, a novel algorithm based on Particle Swarm Optimization is proposed to ensure coverage and connectivity in Wireless Sensor Networks. The objective function is derived from energy function minimization methodologies commonly applied in bounded space circle packing problems. The performance of the novel algorithm is not only evaluated through both simulation and statistical tests that demonstrate the efficacy of the proposed methodology but also compared against that of relative algorithms. Finally, concluding remarks are drawn on the potential extensibility and actual use of the algorithm in real-world scenarios.

Keywords: wireless sensor networks; coverage; k-coverage optimization; connectivity; 1-connectivity; circle packing; particle swarm optimization

Citation: Siamantas, G.; Kandris, D. Particle Swarm Optimization for *k*-Coverage and 1-Connectivity in Wireless Sensor Networks. *Electronics* **2024**, *13*, 4841. https://doi.org/10.3390/electronics13234841

Academic Editors: Binod Vaidya and Byung Rae Cha

Received: 6 November 2024
Revised: 30 November 2024
Accepted: 2 December 2024
Published: 8 December 2024

1. Introduction

A Wireless Sensor Network (WSN) is a group of devices that communicate wirelessly while being disseminated over an area of interest. Typically, it contains a number of sensor nodes along with at least one sink node usually referred to as the base station. The sensor nodes are micro-electromechanical systems that are able to sense ambient conditions, collect and process the relative information and transmit the corresponding data to other nodes or/and the base station. The base station is a device which, thanks to its enhanced energy, computational and communication resources, is able to process received from the network nodes, perform supervisory control of the WSN and communicate with the end user and/or other networks [1,2]. The architecture of a typical WSN is illustrated in Figure 1.

A WSN, taking advantage of the combined abilities of the nodes and the base station(s), is capable of both monitoring the ambient conditions at areas of interest of almost any kind and transmitting related information to destinations no matter how far away they are. For this reason, WSNs have been the basis of the Internet of Things (IoT) and support several areas of human activities such as industry, flora and fauna, environment, healthcare, military sector, and urban development [3]. Consequently, the already wide list of WSNs' applications keeps on continuously growing [4–14].

On the other hand, WSNs face challenges not only due to inborn problems of wireless communications but also because of the severe restrictions of the sensor nodes regarding their resources for energy, storage and processing, not to mention various other application-related difficulties. Hence, there are many scientific challenges and issues that need to be addressed [15]. This is why the development of optimization and multi-objective optimization algorithms is necessary [16]. These algorithms aim to enhance the operation of WSNs by optimizing both individual and multiple performance metrics, subject to a set of constraints, such as coverage and connectivity optimization [17–19], energy sustainability [20], congestion avoidance [21], quality of service attainment [22], security provision [23], data aggregation [24], fault tolerance [25] and node localization [26].

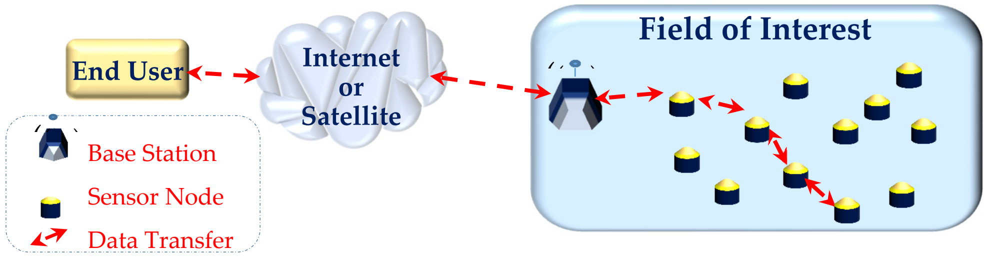


Figure 1. Architecture of a typical WSN.

This research article focuses on coverage and connectivity optimization. The coverage optimization problem can be identified as the problem of locating a certain quantity of sensor nodes, which have a given sensing range, in the finest arrangement pursuing the maximization of the area covered by them and consequently minimizing the coverage holes. Specifically in WSNs, three kinds of coverage may be recognized, namely, area (or regional) coverage, target (or point) coverage and barrier (or path) coverage. Area coverage refers to the monitoring of an area of interest in a pattern that guarantees that, within this area, all points are observed all the time. Target coverage denotes the continuous monitoring of specific points of interest. Barrier coverage refers to the ability to always detect the movement across a barrier of sensor nodes [27]. Actually, the most investigated coverage optimization problem in WSNs is that of area coverage. It can be stated either as 1-coverage or as *k*-coverage (where k is an integer greater than 1), depending on the minimum number of sensor nodes that must always monitor the specific area of interest simultaneously.

Area coverage maximization is pursued in every WSN containing a definite number of sensor nodes. Nevertheless, the area coverage problem in WSNs is recognized as a non-trivial research problem because there are several issues that influence area coverage [28]. For instance, the distribution of sensor nodes within the area of interest can be either arbitrary or deterministic. Similarly, the sensing area may be either probabilistic or deterministic. Also, the sensitivity of the sensor nodes may be either probabilistic or Boolean. Likewise, the communication range of the network nodes may be either variable or invariable. Moreover, the coverage pattern taken on may be either distributed or centralized. Furthermore, sensor nodes in a network may be either static or mobile. At the same time, the optimal distribution of sensor nodes in a network that maximizes coverage may cause connectivity losses to the sensor nodes of this network.

The goal of this research article is to find, under constraints, the optimal spatial deployment of a predetermined set of sensor nodes of varying sensing and communication capabilities to maximize the sensed coverage of a specified area. One constraint might be that of the *k*-coverage of the specific area points, meaning that for the deployment to be valid, there needs to be at least *k* sensor nodes covering each of the designated target points. The other constraint might be that of m-connectivity, which indicates that all pairs of sensor nodes have at least m independent communication paths among them. Multiple connectivity amongst nodes is very desirable because it improves the resilience of the WSN in case of multiple sensor node failures. Unfortunately, maximal connectivity negatively affects the possible area coverage ratio as it requires denser deployments of sensor nodes. Usually, a tradeoff must be made that depends on the specific WSN application requirements. In this research work, only the 1-connectivity case is studied, i.e., the maximization of the covered area is pursued while ensuring that there exists at least one connecting communication path for every pair of sensor nodes [29,30].

In this context, the rest of this article is organized as follows. In Section 2, the theoretical background of this research work is established. In Section 3, the algorithm developed is described. The simulation procedure developed and the corresponding results produced for the algorithm evaluation are presented in Section 4. Finally, in Section 5, concluding remarks are drawn and future research work is proposed.

2. Theoretical Framework

2.1. Fundamental Concepts

There are several sensor sensing and communication models [31–33]. In the context of the research work presented in this article, the model used for both sensing and communication is deterministic This means that a point in the area of interest is assumed to be "sensed" if its Euclidean distance from a sensor node of the WSN is smaller than a specified threshold. The same logic applies to the communication model. If the Euclidean distance between two sensor nodes is less than a given threshold, these nodes are assumed to be able to communicate. Thus, geometrically, the sensing and communication areas of a sensor node are modeled as disks with predefined radii. The corresponding mathematical model is represented in (1).

$$\begin{aligned} P_{\text{sensed}}\left(\text{point}_i, \text{sensor}_j\right) &= \begin{cases} 1, & \text{Distance}\left(\text{point}_i,\ \text{sensor}_j\right) \leq \text{radius}_j \\ 0, & \text{otherwise} \end{cases} \\ P_{\text{comm}}\left(\text{sensor}_i, \text{sensor}_j\right) &= \begin{cases} 1, & \text{Distance}\left(\text{sensor}_i,\ \text{sensor}_j\right) \leq \text{threshold} \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{1}$$

where P_{sensed} and P_{comm} are the probabilities of the area point i to be sensed by the sensor node j and the probabilities of the sensor nodes i and j to communicate, respectively.

In this research work, the Particle Swarm Optimization (PSO) methodology, which is a well-established metaheuristic method used extensively in practice in various fields, is adopted in order to achieve area coverage maximization [34,35]. In this approach, the particle swarm shows a tendency to converge to an optimal solution, which is unfortunately not always guaranteed due to particle interactions. Each particle communicates its current position to its neighbors and, through random perturbations, adjusts its position based on its velocity; the difference between its current position and its recorded personal best position; and the difference between its current position and the best recorded position globally. As the PSO algorithm progresses, the swarm collectively moves toward a region of the search space that potentially contains the global optimum or at least some good solutions to the given optimization problem.

The term particle refers to a possible solution to the problem, i.e., in this research work, the final optimal locations of sensor nodes. The term swarm refers to the set of all particles. As the execution of the algorithm progresses particles are essentially moving on the objective function range trying to find a possible solution. The particle movement is achieved through an exchange of their current positions through communication and a velocity update calculation according to the following equations:

$$\begin{aligned} &v_i \leftarrow wv_i + c_1\mathbf{R_1} \otimes (p_i - x_i) + c_2\mathbf{R_2} \otimes \left(p_g - x_{\boldsymbol{i}}\right) \\ &x_i \leftarrow x_i + v_i \end{aligned} \tag{2}$$

where v_i denotes the velocity of the i-th particle, x_i denotes the position of this particle, p_i denotes the current best particle's position and p_g is the best swarm position found till the current update. $\mathbf{R_1}$ and $\mathbf{R_2}$ are uniform random number vector generators, w is the inertial weight, c_1, c_2 are the acceleration coefficients and $\otimes$ is the symbol of point-wise vector multiplication. The velocity update equation is comprised of three terms, namely, the momentum term that attracts the solution towards the previous direction, the cognitive term that pushes the solution towards the personal best and the social term that attracts the solution close to the global best. A general PSO algorithm is shown in Algorithm 1.

Algorithm 1 PSO Algorithm Outline

Randomize set of particles
while termination conditions unmet **do**
  for all particles **do**
  if $f(x_i) < f(p_i)$ **then**
    $p_i \leftarrow x_i$
   end if
  if $f(x_i) < f(p_g)$ **then**
    $p_g \leftarrow x_i$
   end if
   Update velocity
   Update position
  end for
end while

2.2. Related Work

The optimization of coverage and connectivity problems in WSNs remains a topic of study during all decades of the WSNs' evolution. Also, the field of metaheuristics in optimization, where PSO is categorized, has been expanding for decades and includes dozens of methodology types and hundreds of varieties [36]. Currently, an enormous amount of research activity takes place around the world concerning the problems that arise in WSN deployment both in the theoretical context and at the practical application level. Some theoretical limits concerning coverage and connectivity in sensor networks can be found in [37].

In [38], the maximization of coverage in WSNs is pursued by combining a grid-based strategy along with a PSO algorithm. Specifically, the particles in the fitness functions used are evaluated via a square grid configuration. Also, the maximization of coverage is used in order to achieve the minimization of energy consumption.

A two-pass approach to the WSN coverage problem is given in [39], where a PSO algorithm is first used to deploy optimally the sensor nodes, and at a second pass, a Voronoi diagram method is employed to evaluate the solution fitness. Specifically, the PSO algorithm is used to minimize the area of coverage holes by taking into consideration the distances between the points of interest and the sensors.

In [40], the coverage of a square area is maximized using two approaches, one based on Genetic Algorithms (GAs) and one on PSO. In both approaches, the fitness function was chosen to be the area of the unmonitored part of the target region. Via simulation tests in the seven case studies examined, the efficiency of these two algorithms developed regarding the coverage percentage achieved was evinced to be not only better than that of [38,39] but also close to the ideal one.

Tossa et al. [41] introduce a Genetic Algorithm for area coverage maximization (GAFACM), which works for both regular and irregular areas, with a predefined number of sensor nodes, while guaranteeing their connectivity. An improved social spider optimization (SSO) algorithm is proposed in [42], where its global convergence speed is improved through a population initialization method based on chaos theory and by improving the neighborhood search, global search and matching radius of the algorithm (CSSO).

An improved PSO that tackles possible local convergence problems is presented in [43], where the objective function incorporates coverage and distance metrics, and the velocity update equation is augmented by a gravitational and a Coulomb term. A similar approach is presented in [44], where a Virtual Force-directed PSO (VFPSO) algorithm is employed to achieve maximal coverage. A more involved velocity update PSO equation is presented in [45] with the introduction of the Virtual Force Individual Particle Optimization VFIPO algorithm, which is a combination of a Virtual Force (VF) algorithm, an Individual Particle Optimization (IPO) algorithm and a VFPSO variant. A distributed virtual forces sensor redeployment algorithm (DVFA) that does not require sensor node synchronization is presented in [46]. The algorithm redeploys an initially randomly deployed swarm of sensor nodes in such a way to ensure the uniform coverage of the designated area and the connectivity of the network.

A Genetic Algorithm for area maximization, called MIGA, is proposed in [47]. It uses a novel heuristic initialization procedure, a new fitness function based on an exact integral area calculation and a combinatorial approach to the usage of the crossover operators. A Virtual Force Algorithm (VFA) is employed at the final stage to refine the final solution.

Du, in [48], proposes the usage of a Virtual Force Algorithm (VFA), in combination with a distributed PSO variant (DPSO), for the solution of the 3D coverage of a designated spatial volume by 3D sensing sensor nodes while retaining connectivity. A fusion of various algorithms is presented in [49] where a Resampling Particle Swarm Optimization (RPSO) algorithm is combined with a Particle Swarm Optimization algorithm based on coefficient adjustment (PSO-D) and an improved Virtual Force (VF) algorithm. A method that combines chaos optimization theory and Particle Swarm Optimization in [50] uses PSO first in order to move the sensor nodes close to their optimal positions, and then a Variable Domain Chaos Optimization Algorithm (VDCOA) is employed in order to improve the coverage rate. In [51], a hybrid algorithm (CFL-PSO) is introduced. It is based on combining an enhanced Fick's Law (FL) algorithm (a diffusion-type metaheuristic algorithm) with a Comprehensive Learning (CL) algorithm (another metaheuristic algorithm) and a PSO algorithm in order to achieve maximal area coverage and node connectivity by combining the strengths of all the aforementioned metaheuristic methodologies.

3. Proposed Methodology

3.1. PSO Algorithm Implementation

The main objective of the novel algorithm proposed in this article is to maximize the area coverage by the sensors; k-coverage and 1-connectivity are seen as additional constraints addressed separately and in sequence. So, after the velocity and position updates, the computed particle solutions are checked whether they fulfill the k-coverage constraint or not, that is, whether the sensors cover their assigned targets or not. If not, they are bound to their previous positions. Next, the 1-connectivity constraint is imposed so that particle solutions and their sensors are not 1-connected to their previous solutions. The k-coverage and 1-connectivity algorithms are, respectively, described in Sections 3.3 and 3.4. In this work, only the 1-connectivity case is addressed, as m-connectivity would further constrain the space of feasible particle solutions and require significantly greater computation time to achieve the desired level of optimization convergence. This also applies to the k-coverage constraint with large k values. Yet, the quality of the solution depends also on the covered area geometry and the positions of the target points in the area.

The approach used in the implementation of the PSO algorithm is based on [35]. The sensing areas and the communication range patterns of the sensor nodes are considered to be circles of given radii. The number of particles is denoted as n. Each particle consists of the positions of the q sensors inside the search area. So, the data that each particle should retain are the $(x,\ y)$ coordinate pairs of each sensor, totaling $2{\cdot}q$ numbers. The main data structures that hold the required information for the particle swarm optimizer are as follows:

- An $n \times 2q$ array where each row corresponds to one of the n particles and each column corresponds to one of the q sensor nodes in that particle. The particles are considered vectors of the form $\left[x_1,\ y_1, x_2,\ y_2,\ \cdots, x_q,\ y_q,\right]$, where each $x_i,\ y_i$ pair corresponds to the position of the ith sensor node.
- An $n \times 2q$ array which, similar to the structure of the positions array, holds the n vectors of the velocity and x, y components of the q sensor nodes of each particle $\left[v_{x1},\ v_{y1},\ \cdots, v_{xq},\ v_{yq}\right]$.
- An $n \times 2q$ array holds each particle's personal best position vectors.
- An n-element vector holds each particle's personal best objective function value.

- A variable contains the objective function value of the leader (global best) particle.
- A $2q$-element vector holds the leader particle sensor node positions.

There are also variables that hold the optimization parameters. These include the vectors holding the sensor nodes' communication and sensing ranges. The vectors contain the region of interest geometry. The structure contains the number of particles, the number of sensor nodes, the value of the inertia weight w, the values of the two acceleration coefficients c_1, c_2, the parameter values of the objective function λ, m (see Section 3.2), the values of the velocity bounds v_{lim} (the same for both axes) and the values of the maximum optimization iterations i_{max}. The values of the two convergence parameters, stagnation iteration limit s_{lim}, percentage of change $s_{\%}$ (i.e., convergence is achieved if the objective function value stays inside an $s_{\%}$ band for s_{lim} iterations) and the value of the border-bound percentage $b_{\%}$ where 100% denotes that the bounds on the allowed sensor node positions are the same as the region of interest borders. The last parameter is usually used to avoid small unmonitored areas at the borders if the combined sensing area of all the sensor nodes exceeds the area to be covered. All spatial quantities are normalized.

The particle positions and velocities are initially randomized. In order to assess the performance of the proposed algorithm, that is, how much of the region of interest is covered by the sensor nodes, a Monte Carlo-type estimation algorithm is used. A predetermined number of random points in the region are generated and it is checked whether they fall inside the sensing range of any of the sensor nodes. If they do, they are considered to be covered. The ratio of the number of covered points to the total number of generated points gives the estimate of the covered area. The more random points generated, the greater the accuracy of the estimation, which is at the expense of a higher computation time.

An outline of PSO implementation is shown in Algorithm 2, where the particles' velocities and position updates are implemented according to (2). Each update is followed by checks of bounds. The bounds of the velocity updates are those entered in the v_{lim} variable, as previously described. The bounds of position updates are determined by the area geometry and $b_{\%}$, also described previously in this subsection. Similarly, k-coverage bounds and 1-connectivity constraints are described in detail in Sections 3.3 and 3.4.

Algorithm 2 PSO Implementation

```
for all particles do
    Randomize initial sensor nodes positions and velocities
end for

while no convergence or max iterations not exceeded do
    for all particles do
        o ← calculate particle's objective function
        if o < particle's personal best objective function then
            Make o current particle's personal best objective function
            Make current sensor distribution the particle's personal best
        end if
        Make leader the particle with the current minimum objective function
        Update particle velocities
        Enforce particle velocity bounds
        Update particle positions
        Enforce particle position bounds
        Enforce k-coverage bounds
        Enforce 1-connectivity constraints
    end for
end while
```

3.2. Objective Fitness Function

The problem addressed in this work belongs to the category of sensor cover applications in WSNs. Specifically, the maximization of the total sensed area by a set of sensors in a target-bounded area is pursued by varying their positions inside that area under constraints. The first constraint is the bounds of the given target area. The second is the possibility of having certain target points (also called Points of Interest) that require multiple sensor coverage for redundancy (called k-coverage). The third is that the sensor topology is 1-connected, meaning that there exists at least one connecting communication path for every pair of sensor nodes. The sensing model of each sensor is considered a disk of some radius with a binary probability of detection as explained in Section 2.1.

In such a WSN formulation, the total covered area is maximized when the overlapping sensed area by the sensors is minimized. Ways to tackle this problem that are used extensively in the WSN literature include meta-heuristics optimization methods with objective functions based on Virtual Force methods or direct covered area minimization methods in various variants and mixes [40,48–50].

In this work, the proposed objective function metric stems from the field of discrete geometry optimization, especially that of finding the densest packing of similar objects in a bounded two-dimensional space.

The main idea involves treating the sensing areas of the sensor nodes as disks of the same or different radii, an approximation usually performed in practice, so as to approximate the WSN coverage problem with that of the dense packing of circles inside a given square, a well-studied discrete geometry problem, which of course also requires no overlap between the circles and has both structural similarities in optimization variables and spatial considerations.

This problem has been addressed in various ways [52], one of which involves the minimization of an energy function. This objective function formulation, which is similar to previous Virtual Force-based ones but much clearer and compact in its implementation in a PSO setting, is used.

The following derivation of the energy function is from [52,53]. It begins with the original discrete geometry circle-packing problem and concludes with the derivation of the energy function, which is suitable for use in the PSO implementation. So, the problem of packing circles in a square can be expressed as the problem of locating n points existing within a unit square, such that the minimum distance d_{ij} between any two points i, j is maximal.

The above circle packing problem can be expressed in short as a $2n + 1$ dimensional continuous nonlinear constrained global optimization problem in the following form:

$$\max_{\mathbf{s}_k \in [0,1]^2,\ 1 \le k \le n} \ . \min_{1 \le i < j \le n} \left\| \mathbf{s}_i - \mathbf{s}_j \right\|, \tag{3}$$

where $\mathbf{s}_i$ and $\mathbf{s}_j$ denote the position vectors of the centers of two circles, and $\left\| \mathbf{s}_i - \mathbf{s}_j \right\|$ is the Euclidean distance between them.

The generalized mean converges to the minimum as m tends to negative infinity:

$$\min_{1 \le i < j \le n} \left\| \mathbf{s}_i - \mathbf{s}_j \right\| = \lim_{m \to -\infty} \left(\sum_{1 \le i < j \le n} \left\| \mathbf{s}_i - \mathbf{s}_j \right\|^m \right)^{\frac{1}{m}}, \tag{4}$$

So, the maximization problem in (3) is relaxed into a minimization of a sum of smooth differentiable functions:

$$\min_{\mathbf{s}_i \in [0,1]^2,\ 1 \le i \le n} \sum_{1 \le i < j \le n} \frac{1}{\left\| \mathbf{s}_i - \mathbf{s}_j \right\|^m}, \tag{5}$$

The selection of a large m in (5) increases the significance of the smallest distance in the sum. This sum of powered inverse distances resembles an energy function. So, the whole problem is similar to the energy minimization of repulsing electric charges distributed spatially, with the charges situated at the circles' centers.

A similar energy function E that incorporates a scaling factor λ and an exponent m in the inverse-power interaction was proposed in [53]:

$$E = \sum_{1 \le i < j \le n} \left(\frac{\lambda}{\left\| \mathbf{s}_i - \mathbf{s}_j \right\|^2} \right)^m, \tag{6}$$

The scaling factor λ adjusts for numerical stability when the distance between the sensor nodes is small, while the exponent m, as explained previously, regulates how strongly the interaction diminishes as the sensor node distance increases. Both λ and m were used as additional adjustment parameters in the Particle Swarm Optimization process and can help improve the algorithm's performance.

In order to smooth out E more and improve numerical stability when encountering very large or very small numbers, a logarithmic transformation is used. In this way, the optimization fitness function used in the PSO implementation becomes

$$E_{\text{obj}} = \ln \left[\sum_{1 \le i < j \le n} \left(\frac{\lambda}{\left\| \mathbf{s}_i - \mathbf{s}_j \right\|^2} \right)^m \right], \tag{7}$$

So, the problem formalization, as tested in the simulations against the benchmarks, with a normalized square area boundary, is as follows:

$$\begin{array}{ll} \min\limits_{\{\mathbf{s}_i\}} & f(\{\mathbf{s}_i\}) = \ln \left[\sum_{1 \le i < j \le n} \left(\frac{\lambda}{\|\mathbf{s}_i - \mathbf{s}_j\|^2} \right)^m \right] \\ \text{subject to} & (1) \text{ Boundary Constraints :} \\ & x_i \in [0,1], \;\; y_i \in [0,1], \;\; \text{for } i = 1, \ldots, n \\ & (2)\; k\text{−Coverage Constraints :} \\ & \sum_{i=1}^{n} P_{\text{sensed}}(\text{point}_l, \text{sensor}_i) \ge k, \;\; \text{for } l = 1, \ldots, L \\ & (3) \text{ Connectivity Constraint :} \\ & \{\mathbf{s}_i\} \text{ forms a } 1 - \text{connected network under comm radii } R_{Ci} \end{array} \tag{8}$$

where $\mathbf{s}_i, \mathbf{s}_j$ are sensors' coordinates, P_{sensed} is defined in (1), R_{Ci} are the sensors' communication radii, L is the number of target points (POIs) requiring k-coverage and point_l are the coordinates of each target point.

3.3. k-Coverage Implementation Algorithm

In order to achieve the desired k-coverage in some of the specified points in the sensed area, the initial positions of k of each particle's sensor nodes are assigned to the position of each of the specified points that require coverage from k sensor nodes. At each optimization iteration, a check is performed to determine whether the distances of the assigned sensor nodes from the k-covered points exceed the nodes' radii. If this happens, their distance is bound to that of their corresponding sense radius. The assigned sensor node's bounded coordinates are calculated to be on the line connecting the specified k-covered point with the assigned node, as shown in Figure 2. The previous bound checks are performed for all pairs of k-covered points and their assigned k nodes.

In Figure 2, the k-coverage sensor point geometry is depicted. Specifically, point A represents the specified k-covered point, while point B is one of its assigned sensor nodes. Also, the radius of the circle corresponds to the sensing area radius R of that node. Likewise, point C denotes the intersection of their distance direction line with the sensing area circle. The coordinates of point C are the new bounded coordinates of the assigned node as its

Euclidean distance d from the k-covered point exceeds that of the sense radius R, and its coordinates are calculated as follows:

$$\begin{aligned} x_c &= x_A + \tfrac{R}{d}(x_B - x_A) \\ y_c &= y_A + \tfrac{R}{d}(y_B - y_A), \end{aligned} \tag{9}$$

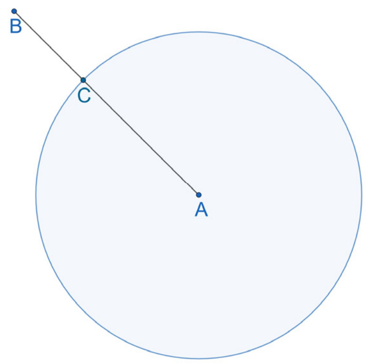


Figure 2. k-coverage sensor point geometry.

It is the same to center the sense circle at A, although it should be centered at B, to calculate the bounded coordinates relative to B. So, the assigned nodes will always cover their respective k-covered points and maximize their mutual non-overlapping coverage area.

When the sensor nodes in the WSN have different sense radii, they are assigned to their corresponding k-covered points in ascending sense radius order to provide the larger sense radius nodes flexibility to cover larger parts of the area of interest. It is also more beneficial when the k-covered points of interest lie close to the area border where larger sense radii assigned nodes will have a greater possibility to have more mutually overlapped covered areas. The k-coverage implementation is illustrated in Algorithm 3.

Algorithm 3 k-Coverage Implementation

$p \leftarrow$ number of k-coverage Points of Interest (POIs)
$k \leftarrow$ number of demanded k-coverage Sensor Nodes (SNs)
$nn \leftarrow p \times k$ ▷ number of assigned SNs in each PSO particle

for all particles **do**
    Initialize first nn SN positions to the requested POI positions
end for

while in PSO loop **do**
    ...
    for all particles **do**
        for all nn assigned SNs **do**
            Calculate distance from corresponding POI position
            if distance $>$ SN's sense radius **then**
                Compute intersection coords of sense circle with distance line
                Bound SN's coords to circle intersection coords
            end if
        end for
    end for
    ...
end while

3.4. 1-Connectivity Implementation Algorithm

The communication range pattern of each sensor node is considered circular with a radius R_C double that of their sensing range R_S in the case where two nodes have the same sensing ranges. The ratio R_C/R_S should be at least 2 for the network to remain connected and to make it possible to achieve maximum coverage. If this ratio is less than 2, the overlapping of sensor disks is needed to maintain communication, thus resulting in reduced area coverage. Accordingly, if two nodes have different sensing ranges, then their communication range should be equal to the sum of their respective sensing ranges. For the sensor nodes to achieve the largest possible covered area, the communication range of each node should be chosen so that it is equal to the sum of its sensing range, the sensing range of the node and the largest sensing range. This way, a set of nodes of different sense radii could cover the maximal area with their sensor nodes while making it possible to maintain 1-connectivity (see examples in Figure 3).

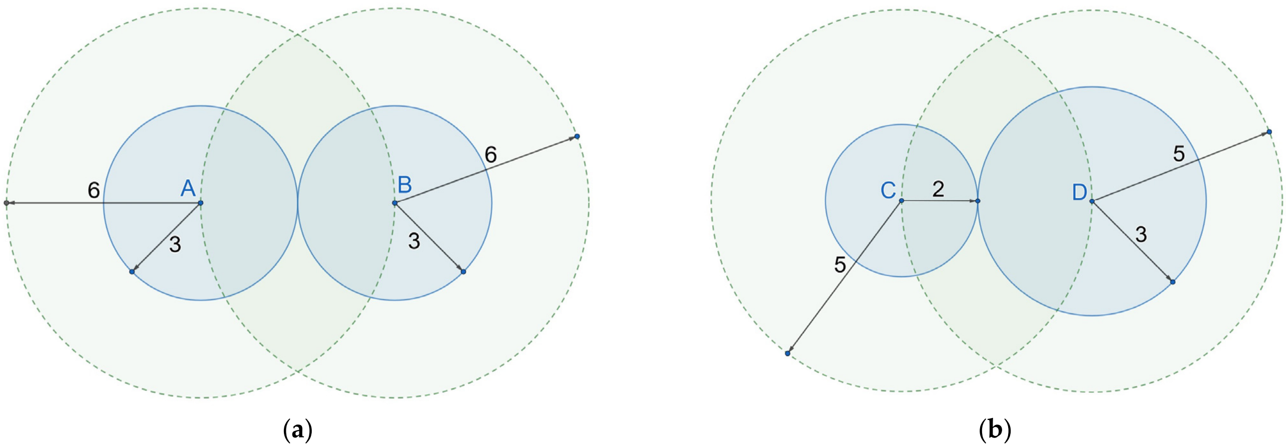


Figure 3. Calculation example of two sensor nodes communication ranges (dashed lines) based on their sensing ranges (solid lines) where the letters denote the locations of these sensor nodes: (**a**) case where both sensing ranges = 3 and both communication ranges = 6; (**b**) case where sensing ranges = 2 and 3, and both communication ranges = 5.

To test the 1-connectivity of the sensor network, a depth-first search (DFS) algorithm [54] is used on its distance-weighted graph in order to identify the non-connected sensor nodes with the given sensor node communication ranges. In the PSO initialization phase, it is ensured that the initial random particle sensor node distributions are 1-connected. To achieve this, the sensor nodes' communication to sensing range ratios should be chosen according to the method stated previously. If this is not the case, the algorithm will abort after some predetermined number of tries if 1-connectivity is not achieved. In order to keep the sensor network 1-connected during the optimization process, each particle is checked for at least 1-connectivity for all optimization iterations. If at some iteration a particle's spatial sensor node distribution checks that it is not at least 1-connected, then it is maintained at its most recent 1-connected configuration. This continues till the convergence of the optimization process is achieved and ensures that the final optimal sensor node distribution will be at least 1-connected. The 1-connectivity constraint is executed after the k-coverage constraint (see Section 3.3) in the PSO process.

The 1-connectivity implementation is shown in Algorithm 4.

Algorithm 4 1-Connectivity Implementation

```
sr ← Sensor Nodes (SNs) sensing ranges array
cr ← SNs communication ranges array
Sort sr, cr in ascending order

for all particles do
    repeat
        Randomize particle's SNs positions
    until SNs are 1-connected OR abort PSO if tries limit is exceeded
end for

while in PSO loop do
    . . .
    for all particles do
        previous_particle_pos_vector←particle_pos_vector
        particleposvector ← update particle's position vector
        enforce k-coverage constrains
        if particle SNs are not 1-connected then
            particleposvector ←previous_particle_pos_vector
        end if
    end for
    . . .
end while
```

4. Simulation Tests and Performance Evaluation

In order to accomplish the performance evaluation of the proposed algorithm, thorough simulation tests were conducted in the MathWorks MATLAB version R2023b environment in regards of seven case studies. The specific case studies were selected with the intention to be used as a reference because they are exactly the same as the ones that are used in a relevant and well-known research work [40], which have been proven to outperform other distinguished coverage maximization methodologies [38,39].

In all case studies, the areas to be covered by the sensors are square. Square areas are used extensively in the literature for first evaluations of meta-heuristic algorithms in these kinds of problems. Each case enables the testing of the proposed algorithm in experiments with varying sensing and communication sensor characteristics.

One difference among these cases regards their dimension ratio r_d, which is the ratio of the sensing range radius to the square area side; this ratio serves as a metric of the size difference between the geometries of the sensor and the area. The second difference factor among the case studies concerns the area ratio r_a, which is the ratio of the maximum possible sensing area by all sensors to the area of the whole region to be sensed. This specific metric gives a measure of the redundancy of the number of sensors used to cover the given area, e.g., ratios lower than 1 denote that the given sensor number will at most cover only $r_a \times 100\%$ of the target square area. It has been shown experimentally that these two factors affect the convergence performance of PSO algorithms on this kind of optimization problem.

More specifically, case study 1 simulates an environment with a dimension ratio r_d of 0.075 and an area ratio r_a of 0.618, while the sensing range of sensors is fixed. Case study 2 differentiates from case study 1 only due to the fact that the sensors have varying sensing ranges. Case study 3 tests the k-coverage algorithm on six target points (k = 6), while the sensing range is fixed. Case study 4 is similar to case 3, except for the fact that sensors have varying sensing ranges. Case study 5 simulates an environment with a dimension ratio r_d of 0.1 and an area ratio r_a of 1.256. Case study 6 simulates an environment with a dimension ratio r_d of 0.1 and an area ratio r_a of 0.628. Case study 7 simulates an environment with a dimension ratio r_d of 0.17 and an area ratio r_a of 1.745. The details of each case are presented in their corresponding subsection. A summary of all case studies' characteristics is shown in Table 1.

Table 1. Case studies' characteristics.

Case No.	Sensing Range	*k*-Coverage	r_d	r_a
1	Fixed	No	0.075	0.618
2	Varying	No	Varying	Varying
3	Fixed	Yes	0.075	0.618
4	Varying	Yes	Varying	Varying
5	Fixed	No	0.1	1.256
6	Fixed	No	0.1	0.628
7	Fixed	No	0.17	1.745

Because the PSO methodology is a probabilistic (stochastic) method, in each case study, the simulation test was conducted 30 times. The mean value and the standard deviation of the area coverage percentage were calculated and are presented in the corresponding tables. Additionally, the highest area coverage achieved in each case is included with the ideal coverage, which is the combined coverage that all the sensor nodes can provide in an unbounded environment.

In each of the case studies analyzed, a comparative assessment of the investigated PSO algorithms was performed using the t-test methodology. This statistical technique allows for the evaluation of hypotheses related to a population by determining the *p*-value, which quantifies the degree of agreement between the data and the null hypothesis. The null hypothesis, in this scenario, assumes that there is no difference in the mean results produced by the two competing algorithms. The statistical tests compared the proposed algorithm against the PSO algorithms and not the Genetic Algorithms.

The PSO parameters used have fixed values during the whole simulation. Their values were calculated through meta-optimization around an initial parameter vector on the first test case and then manually rounded off and tuned around these calculated values for each case [55]. The optimization method used was a direct search method based on a pattern search strategy, which is an algorithm designed to solve nonlinear optimization problems without requiring gradient information [56,57]. The whole test case was entered as a function of the pattern search optimizer with the particle swarm parameters as the optimization variables. The particle swarm parameters used in the pattern search optimizer were the value of the inertia weight, the values of the two acceleration coefficients, the two parameter values of the objective function (see Section 3.2) and the values of the velocity bounds. All other parameters (particle number, maximum iterations, and optimization termination tolerance) were chosen to be as close as possible to the respective values used in [40] in order to better compare the performance of the algorithms.

In all the case studies examined, the simulation results and the optimal node placement are depicted in the accompanying tables and figures. Specifically, in the figures depicting the optimal node locations with 1-connectivity, the cyan disks show the sensing area of each node, the green circles show the communication range of each node, the red lines show the established communication connections between the nodes and the numbers above the sensor node positions show how many other nodes this sensor node is connected with.

4.1. Case Study 1

In this case study, the primary objective was to maximize the coverage of a two-dimensional square area measuring 20×20 units. A total of 35 sensor nodes were deployed, each equipped with a sensing range of 1.5 units and a communication range of 3.0 units. The Particle Swarm Optimization (PSO) algorithm was configured with a particle size of

200 in the scenario with no connectivity required and 600 particles in the scenario requiring the 1-connectivity constraint.

The PSO parameters in the no connectivity requirement case were as follows: $w = 0.2$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.7$, $m = 10$, $v_{lim} = 0.05$, $i_{max} = 130$, $s_{max} = 20$, $s_{\%} = 1\%$, $b_{\%} = 100\%$, while, in the case with the 1-connectivity requirement, they were as follows: $w = 0.2$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.7$, $m = 10$, $v_{lim} = 0.005$, $i_{max} = 130$, $s_{max} = 40$, $s_{\%} = 1\%$, $b_{\%} = 100\%$.

In Table 2, the results of the corresponding simulation tests are shown.

Table 2. Case study 1: simulation results.

Parameter	GA [40]	PSO [40]	PSO No Conn	PSO 1-Conn
Mean Value	61.17	60.92	61.79	61.49
Standard Deviation	0.28	0.46	0.12	0.15
Best Fitness	61.56	61.49	61.86	61.73
p-Value			0.00	
Ideal Coverage	61.86			

It is evident that both algorithms with and without 1-connectivity requirements had better mean values and standard deviations of the area coverage percentage compared to the Genetic Algorithm (GA) and PSO-based algorithms in [40] and were close to the ideal coverage. The p-value in the PSO with no connectivity requirement was negligible, signifying that the proposed algorithm is better.

In Figures 4 and 5, the optimal node locations with and without 1-connectivity requirements are correspondingly shown.

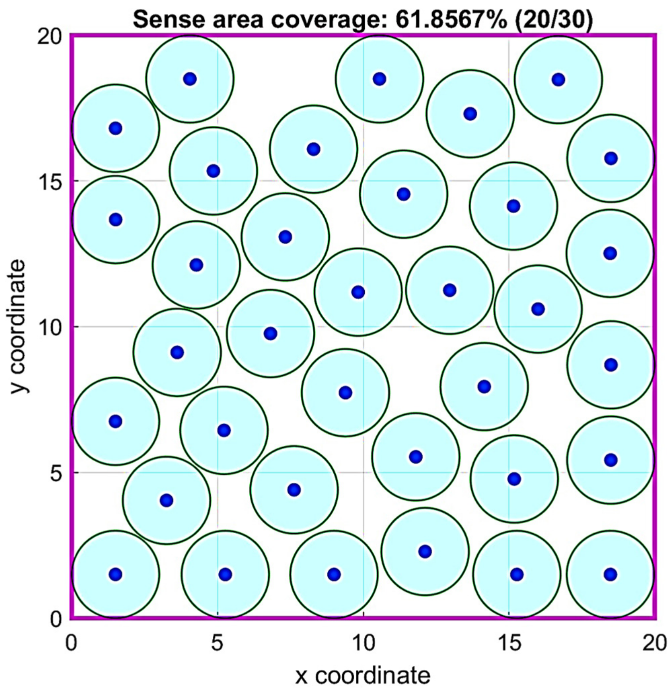


Figure 4. Case study 1: optimal node locations.

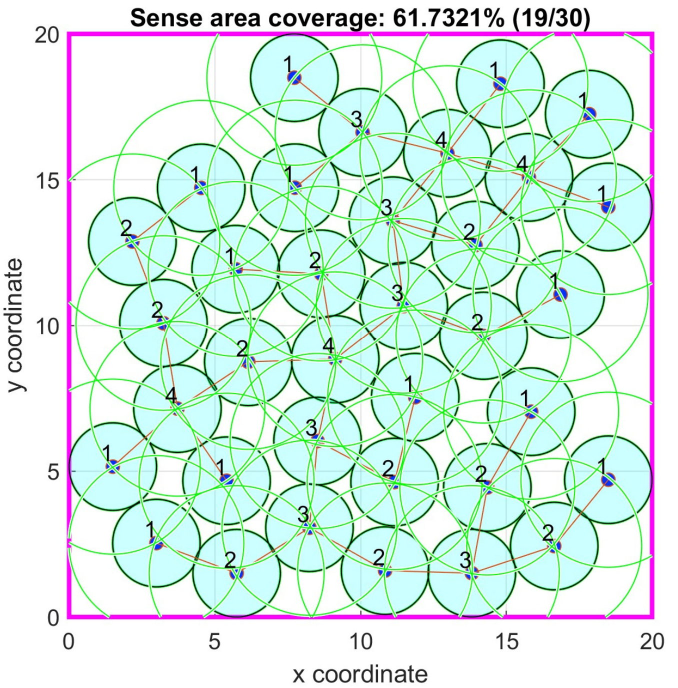


Figure 5. Case study 1: optimal node locations with 1-connectivity.

In Figure 6, the objective function value iterations for both requirements are illustrated.

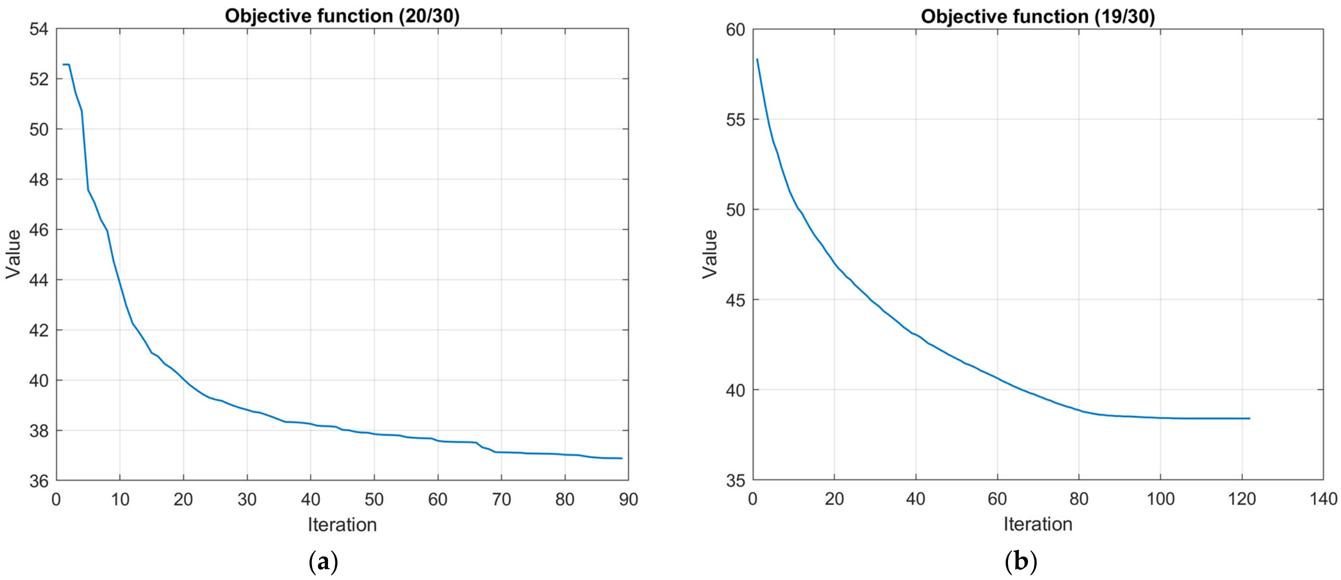


Figure 6. Case study 1: objective function iterations: (**a**) no connectivity; (**b**) 1-connectivity.

4.2. Case Study 2

In this case study, the primary objective was to maximize the coverage of a two-dimensional square area measuring 20 × 20 units. A total of 32 sensor nodes were deployed with varying sensing ranges. Five sensor nodes had a sensing range of 0.8 units and a communication range of 2.8 units. Twenty sensor nodes had a sensing range of 1.5 units and a communication range of 3.5 units. Seven sensor nodes had a sensing range of 2 units and a communication range of 4 units.

The Particle Swarm Optimization (PSO) algorithm was configured with a particle size of 200 in the scenario with no connectivity required and 600 particles in the scenario requiring the 1-connectivity constraint.

The PSO parameters in the no connectivity requirement case were as follows: $w = 0.5$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.7$, $m = 10$, $v_{lim} = 0.01$, $i_{max} = 130$, $s_{max} = 20$, $s_{\%} = 1\%$, $b_{\%} = 100\%$, while the parameters in the case with the 1-connectivity requirement were as follows: $w = 0.2$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.7$, $m = 10$, $v_{lim} = 0.075$, $i_{max} = 130$, $s_{max} = 40$, $s_{\%} = 1\%$, $b_{\%} = 100\%$.

In Table 3, the results of the corresponding simulation tests are shown.

Table 3. Case study 2: simulation results.

Parameter	GA [40]	PSO [40]	PSO No Conn	PSO 1-Conn
Mean Value	59.37	58.83	59.44	57.77
Standard Deviation	0.18	0.38	0.27	0.62
Best Fitness	59.69	59.32	59.85	58.85
p-Value			0.00	
Ideal Coverage		59.85		

It is obvious that the algorithm without the 1-connectivity requirement had better mean values and standard deviations of the area coverage percentage compared to the PSO-based algorithm in [40] and was close to the ideal coverage. The p-value in the PSO with no connectivity requirement was negligible, signifying that the proposed algorithm is better.

In Figures 7 and 8, the optimal node locations with and without 1-connectivity requirements are correspondingly shown.

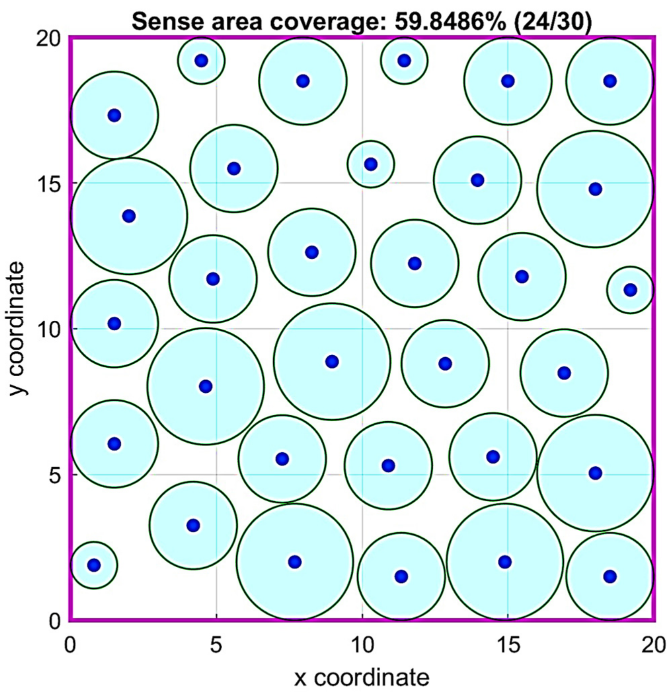


Figure 7. Case study 2: optimal node locations.

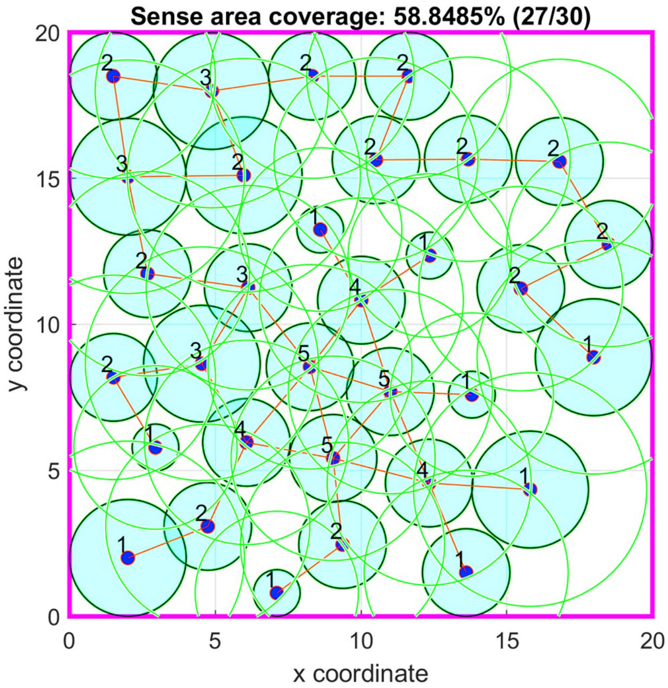


Figure 8. Case study 2: optimal node locations with 1-connectivity.

In Figure 9, the objective function value iterations for both requirements are illustrated.

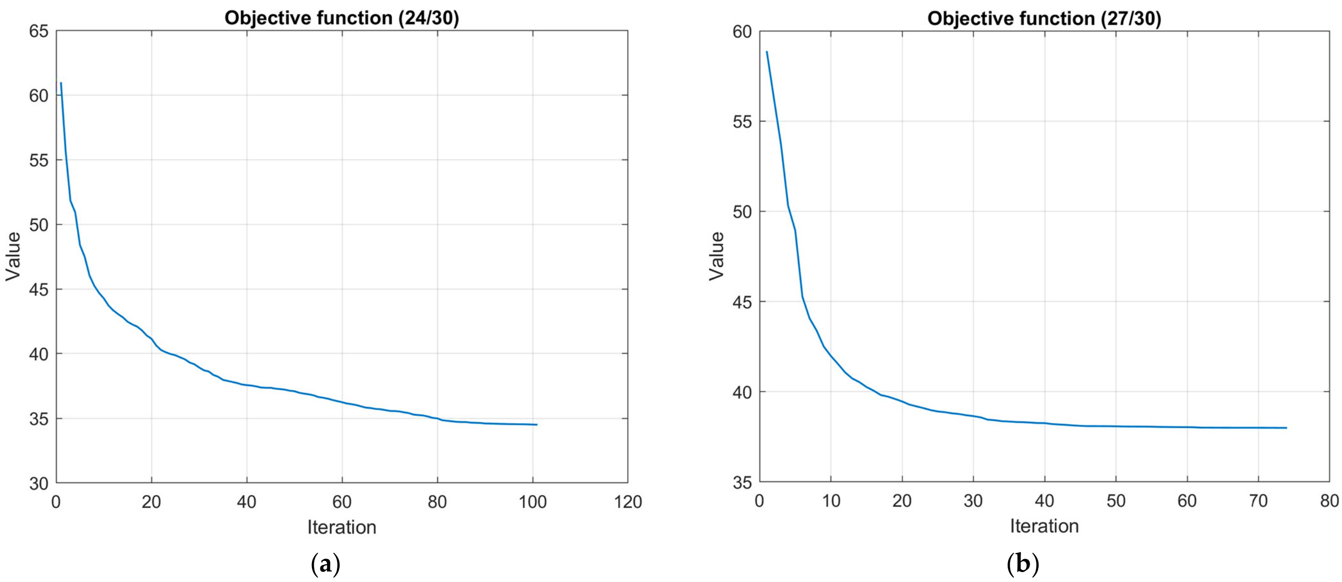


Figure 9. Case study 2: objective function iterations: (**a**) no connectivity; (**b**) 1-connectivity.

4.3. Case Study 3

In this case study, the primary objective was to maximize the coverage of a two-dimensional square area measuring 20×20 units. A total of 45 sensor nodes were deployed, each equipped with a sensing range of 1.5 units and a communication range of 3.0 units. Additionally, it was required to have a 3-coverage constraint for six area points at positions (5, 5), (10, 5), (15, 5), (5, 15), (10, 15) and (15, 15).

The Particle Swarm Optimization (PSO) algorithm was configured with a particle size of 600 in the scenario with no connectivity required and 1200 particles in the scenario requiring the 1-connectivity constraint.

The PSO parameters in the no connectivity requirement case were as follows: $w = 0.2$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.7$, $m = 10$, $v_{lim} = 1.0$, $i_{max} = 130$, $s_{max} = 20$, $s_{\%} = 1\%$, $b_{\%} = 100\%$, while, in the case with the 1-connectivity requirement, they were as follows: $w = 0.2$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.7$, $m = 10$, $v_{lim} = 0.2$, $i_{max} = 130$, $s_{max} = 40$, $s_{\%} = 1\%$, $b_{\%} = 100\%$.

In Table 4, the results of the corresponding simulation tests are shown.

Table 4. Case study 3: simulation results.

Parameter	GA [40]	PSO [40]	PSO No Conn	PSO 1-Conn
Mean Value	73.07	72.13	74.21	72.69
Standard Deviation	0.66	0.85	0.98	1.45
Best Fitness	74.28	73.77	75.76	75.05
p-Value			0.00	
Ideal Coverage	79.52			

It is apparent that both algorithms with and without 1-connectivity requirements had better mean values of the area coverage percentage than the PSO-based one in [40] and the 3-coverage constraint was satisfied at all required points. The p-value in the PSO with no connectivity requirement was negligible, signifying that the proposed algorithm is better.

In Figures 10 and 11, the optimal node locations with and without 1-connectivity requirements are shown correspondingly.

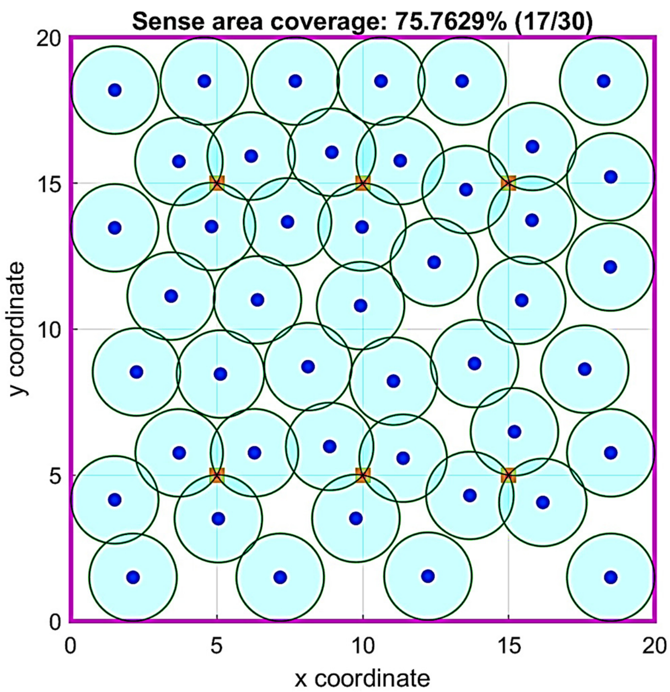


Figure 10. Case study 3: optimal node locations.

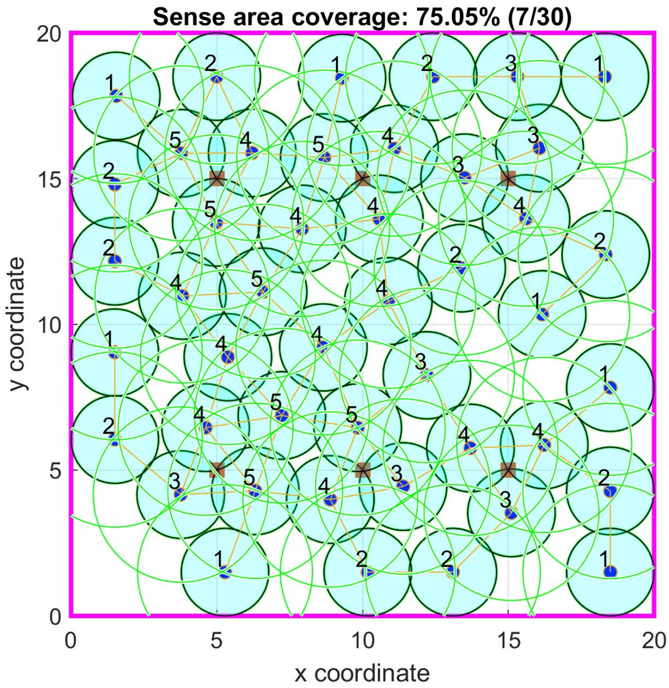
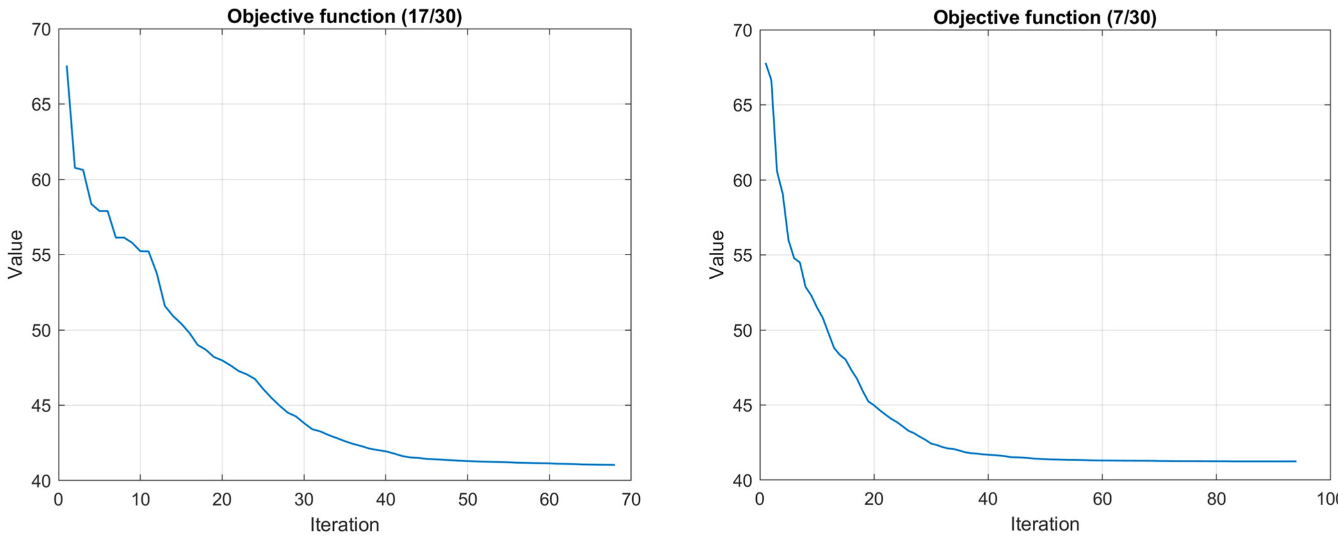


Figure 11. Case study 3: optimal node locations with 1-connectivity.

In Figure 12, the objective function value iterations for both requirements are illustrated.



(**a**) (**b**)

Figure 12. Case study 3: objective function iterations: (**a**) no connectivity; (**b**) 1-connectivity.

4.4. Case Study 4

In this case study, the primary objective was to maximize the coverage of a two-dimensional square area measuring 20×20 units. A total of 45 sensor nodes were deployed this time with varying sensing ranges. Eighteen sensor nodes had a 1.0 unit sensing range and a communication range of 3.0 units. Twenty sensor nodes had a sensing range of 1.5 units and a communication range of 3.5 units. Seven sensor nodes had a sensing range of 2 units and a communication range of 4 units. Also, it was required to have a 3-coverage constraint for six area points at positions (5, 5), (10, 5), (15, 5), (5, 15), (10, 15) and (15, 15).

The Particle Swarm Optimization (PSO) algorithm was configured with a particle size of 600 in the scenario with no connectivity required and 1200 particles in the scenario requiring the 1-connectivity constraint.

The PSO parameters in the no connectivity requirement case were as follows: $w = 0.47$, $c_1 = 1.59$, $c_2 = 1.53$, $\lambda = 0.24$, $m = 3.15$, $v_{lim} = 1.75$, $i_{max} = 240$, $s_{max} = 100$, $s_{\%} = 0.01\%$, $b_{\%} = 100\%$, while, in the case with the 1-connectivity requirement, they were as follows: $w = 0.5$, $c_1 = 1.56$, $c_2 = 1.56$, $\lambda = 0.24$, $m = 3$, $v_{lim} = 0.05$, $i_{max} = 240$, $s_{max} = 40$, $s_{\%} = 1\%$, $b_{\%} = 100\%$.

In Table 5, the results of the relevant simulation tests are shown.

Table 5. Case study 4: simulation results.

Parameter	GA [40]	PSO [40]	PSO No Conn	PSO 1-Conn
Mean Value	67.39	69.89	67.69	65.72
Standard Deviation	0.45	1.15	0.97	1.64
Best Fitness	68.24	71.46	69.19	68.41
p-Value			0.066 *	
Ideal Coverage		71.47		

* p-Value comparing with the GA [40].

It is apparent that the PSO algorithm without the 1-connectivity requirement has a slightly better mean value of the area coverage percentage than the GA [40], while the 3-coverage constraint is satisfied on all the required points. The p-value between the proposed PSO with no connectivity requirement and the GA [40] is 0.066, signifying that there is some evidence that the proposed algorithm is better in this case. Although this test case is computationally expensive due to its complexity, further tuning of the PSO algorithm's parameters is feasible and will further increase its performance.

In Figures 13 and 14, the optimal node locations with and without 1-connectivity requirements are shown, respectively.

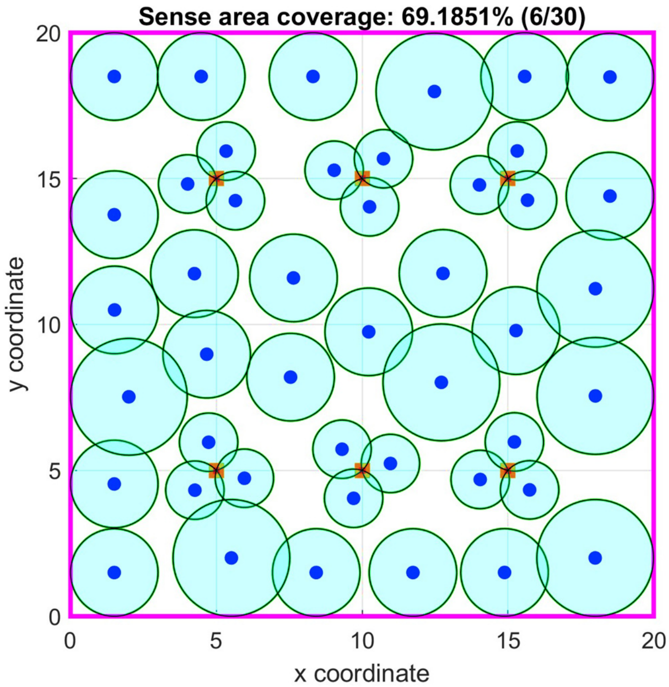


Figure 13. Case study 4: optimal node locations.

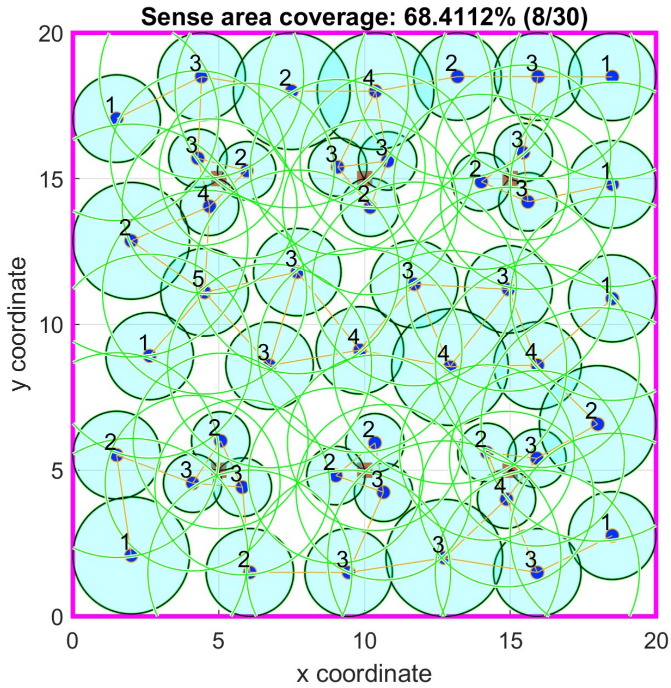


Figure 14. Case study 4: optimal node locations with 1-connectivity.

In Figure 15, the objective function value iterations for both requirements are shown.

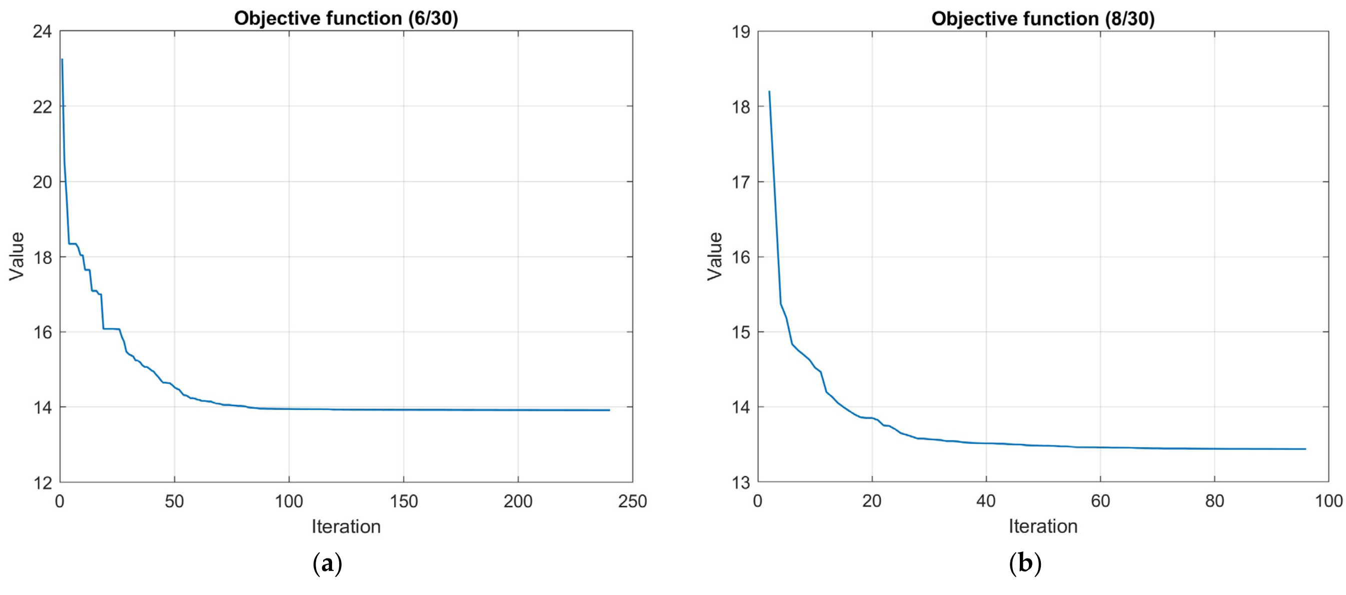


Figure 15. Case study 4: objective function iterations: (**a**) no connectivity; (**b**) 1-connectivity.

4.5. Case Study 5

In this case study, the primary objective was to maximize the coverage of a two-dimensional square area measuring 50 × 50 units. A total of 40 sensor nodes were deployed, each equipped with a sensing range of 5.0 units and a communication range of 10.0 units. The Particle Swarm Optimization (PSO) algorithm was configured with a particle size of 200 for both the scenarios with no connectivity required and the scenario requiring the 1-connectivity constraint.

The PSO parameters in the no connectivity requirement case were as follows: $w = 0.5$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.2133$, $m = 3$, $v_{lim} = 1.0$, $i_{max} = 200$, $s_{max} = 200$, $s_{\%} = 1\%$, $b_{\%} = \sqrt{2}/2 \times 100\%$, while in the case with the 1-connectivity requirement, they were as follows: $w = 0.5$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.2133$, $m = 3$, $v_{lim} = 0.02$, $i_{max} = 250$, $s_{max} = 250$, $s_{\%} = 1\%$, $b_{\%} = \sqrt{2}/2\times$ 100%. In this case, there was no convergence limit, and the border-bound percentage was set in such a way to allow the sensor nodes to move closer to the area boundary and leave no unmonitored parts at the border as the combined sensing area of the sensor nodes exceeds the total target area.

In Table 6, the results of the corresponding simulation tests are shown.

Table 6. Case study 5: simulation results.

Parameter	GA [40]	PSO [40]	PSO No Conn	PSO 1-Conn
Mean Value	96.40	95.53	97.37	97.58
Standard Deviation	0.59	0.66	0.29	0.20
Best Fitness	-	-	97.95	98.02
p-Value			0.00	
Ideal Coverage		100		

It is evident that both algorithms with and without 1-connectivity requirements had better mean values and standard deviations of the area coverage percentage than the Genetic Algorithm (GA) and PSO-based algorithms in [40] and were quite close to the ideal coverage. The p-value in the PSO with no connectivity requirement was negligible, signifying that the proposed algorithm is better.

In Figures 16 and 17, the optimal node locations with and without 1-connectivity requirements are shown correspondingly.

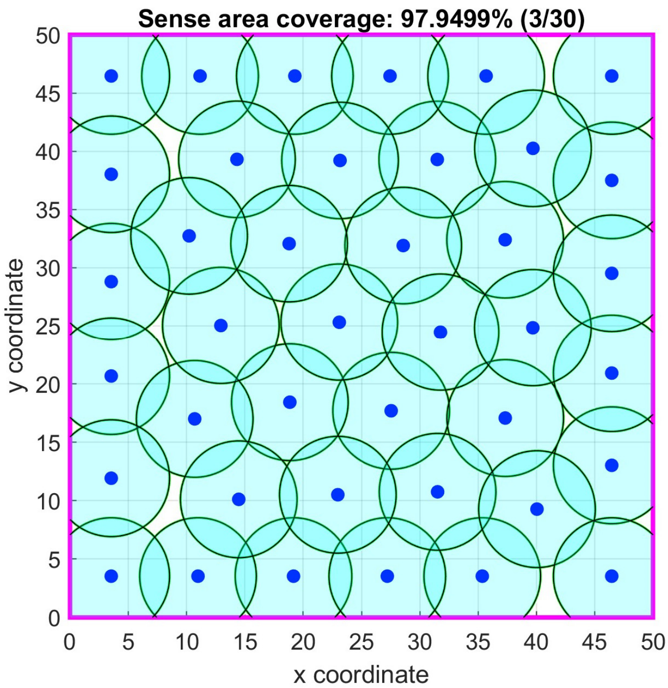


Figure 16. Case study 5: optimal node locations.

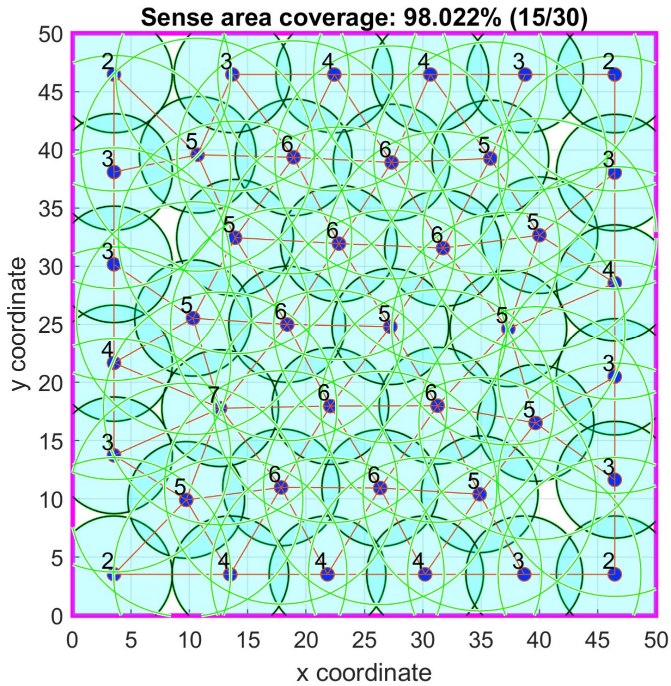


Figure 17. Case study 5: optimal node locations with 1-connectivity.

In Figure 18, the objective function value iterations for both requirements are illustrated.

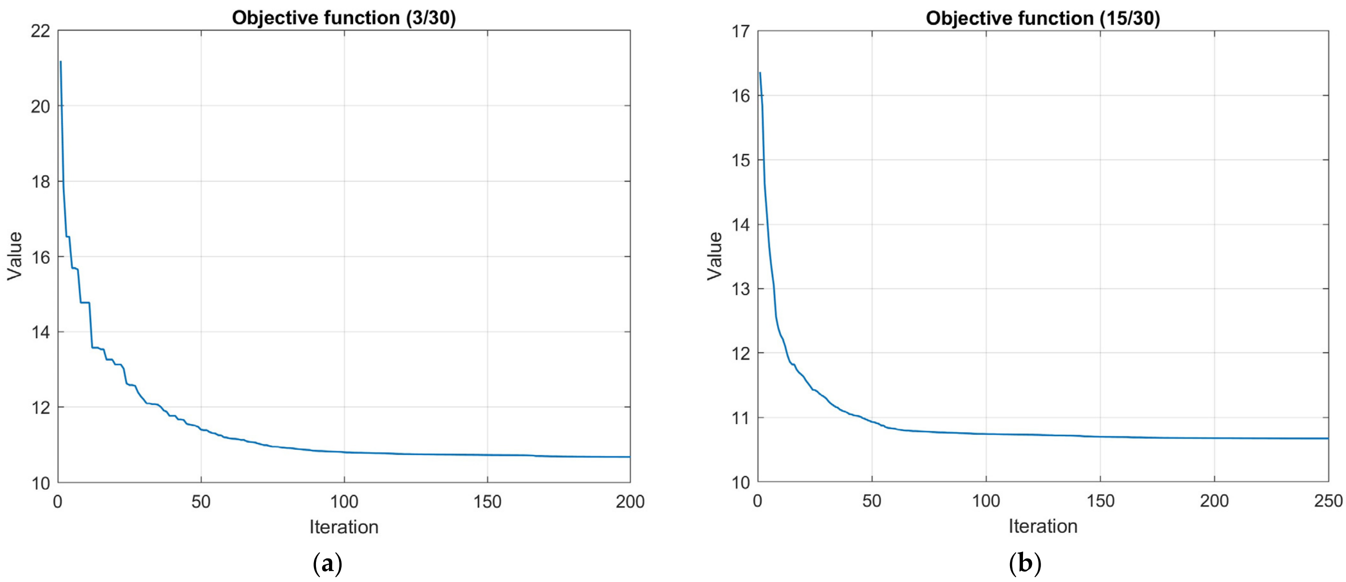


Figure 18. Case study 5: objective function iterations: (**a**) no connectivity; (**b**) 1-connectivity.

4.6. Case Study 6

In this case study, the primary objective was to maximize the coverage of a two-dimensional square area measuring 50×50 units. A total of 20 sensor nodes were deployed, each equipped with a sensing range of 5.0 units and a communication range of 10.0 units. The Particle Swarm Optimization (PSO) algorithm was configured with a particle size of 200 for both the scenario with no connectivity required and the scenario requiring the 1-connectivity constraint.

The PSO parameters in the no connectivity requirement case were as follows: $w = 0.5$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.2133$, $m = 3$, $v_{lim} = 0.16$, $i_{max} = 150$, $s_{max} = 150$, $s_{\%} = 1\%$, $b_{\%} = 100\%$, while, in the case with the 1-connectivity requirement, they were as follows: $w = 0.5$, $c_1 = 1.5$, $c_2 = 1.5$, $\lambda = 0.2133$, $m = 3$, $v_{lim} = 0.016$, $i_{max} = 200$, $s_{max} = 200$, $s_{\%} = 1\%$, $b_{\%} = 100\%$. In this case, there was no convergence limit.

In Table 7, the results of the corresponding simulation tests are shown.

Table 7. Case study 6: simulation results.

Parameter	GA [40]	PSO [40]	PSO No Conn	PSO 1-Conn
Mean Value	62.50	62.38	62.83	61.95
Standard Deviation	0.23	0.18	0.006	0.30
Best Fitness	-	-	62.84	62.38
p-Value			0.00	
Ideal Coverage		62.84		

It is apparent that the algorithm without the 1-connectivity requirement had better mean values and standard deviations of the area coverage percentage than the Genetic Algorithm (GA) and PSO-based algorithms in [40] and was close to the ideal coverage. The p-value in the PSO with no connectivity requirement was negligible, signifying that the proposed algorithm is better.

In Figures 19 and 20, the optimal node locations with and without 1-connectivity requirements are shown correspondingly.

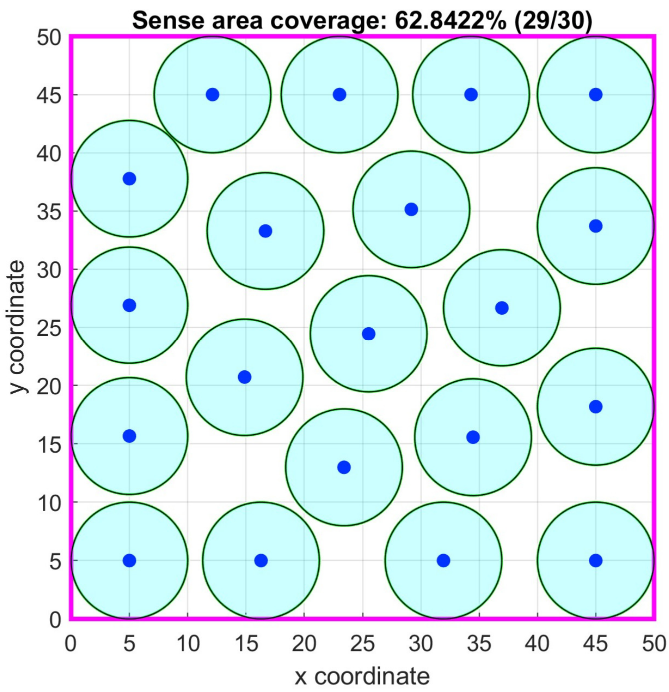


Figure 19. Case study 6: optimal node locations.

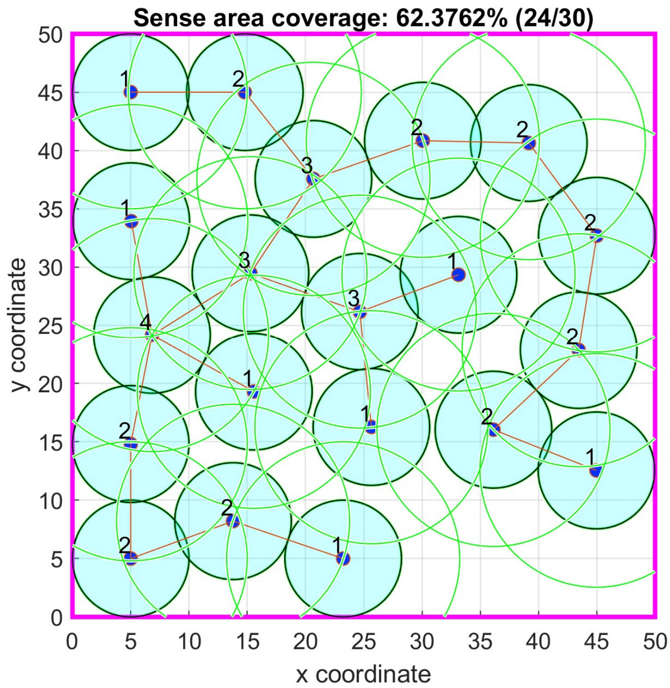


Figure 20. Case study 6: optimal node locations with 1-connectivity.

In Figure 21, the objective function value iterations for both requirements are illustrated.

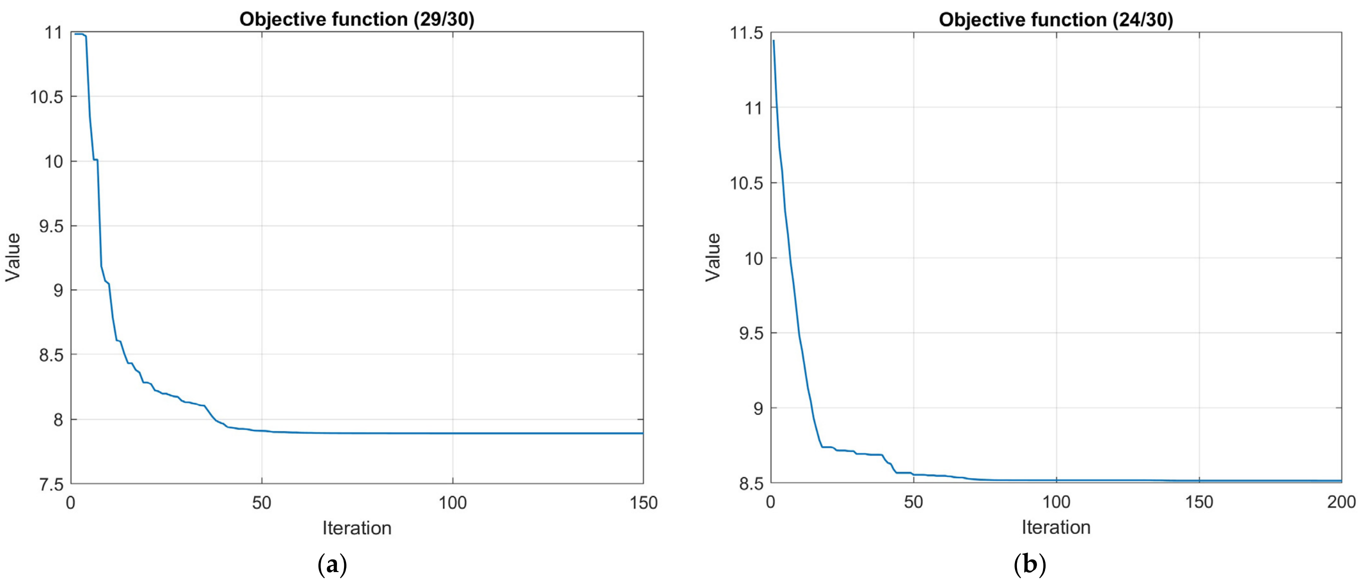


Figure 21. Case study 6: objective function iterations: (**a**) no connectivity; (**b**) 1-connectivity.

4.7. Case Study 7

In this case study, the primary objective was to maximize the coverage of a two-dimensional square area measuring 30 × 30 units. A total of 20 sensor nodes were deployed, each equipped with a sensing range of 5.0 units and a communication range of 10.0 units. The Particle Swarm Optimization (PSO) algorithm was configured with a particle size of 50 for both the scenario with no connectivity required and the scenario requiring the 1-connectivity constraint.

The PSO parameters in the no connectivity requirement case were as follows: w = 0.4, c_1 = 1.5, c_2 = 1.575, λ = 0.3, m = 1.5, v_{lim} = 16.67, i_{max} = 1000, s_{max} = 1000, $s_{\%}$ = 1%, $b_{\%} = \sqrt{2}/2 \times 100\%$, while, in the case with the 1-connectivity requirement, they were as follows: w = 0.4, c_1 = 1.5, c_2 = 1.575, λ = 0.3, m = 1.5, v_{lim} = 16.67, i_{max} = 500, s_{max} = 500, $s_{\%}$ = 1%, $b_{\%} = \sqrt{2}/2\times$ 100%. In this case, there was no convergence limit and the border-bound percentage was set in such a way to allow the sensor nodes to move closer to the area boundary and leave no unmonitored parts at the border as the combined sensing area of the sensor nodes exceeds the total target area.

In Table 8, the results of the corresponding simulation tests are shown.

Table 8. Case study 7: simulation results.

Parameter	GA [40]	PSO [40]	PSO No Conn	PSO 1-Conn
Mean Value	99.76	99.55	99.93	99.90
Standard Deviation	0.10	0.17	0.06	0.07
Best Fitness	-	-	99.96	99.96
p-Value			0.00	
Ideal Coverage		100		

It is apparent that both algorithms with and without 1-connectivity requirements had better mean values and standard deviations of the area coverage percentage than the Genetic Algorithm (GA) and PSO-based algorithms in [40] and were quite close to the ideal coverage. The p-value in the PSO with no connectivity requirement was negligible, signifying that the proposed algorithm is better.

In Figures 22 and 23, the optimal node locations with and without 1-connectivity requirements are shown correspondingly.

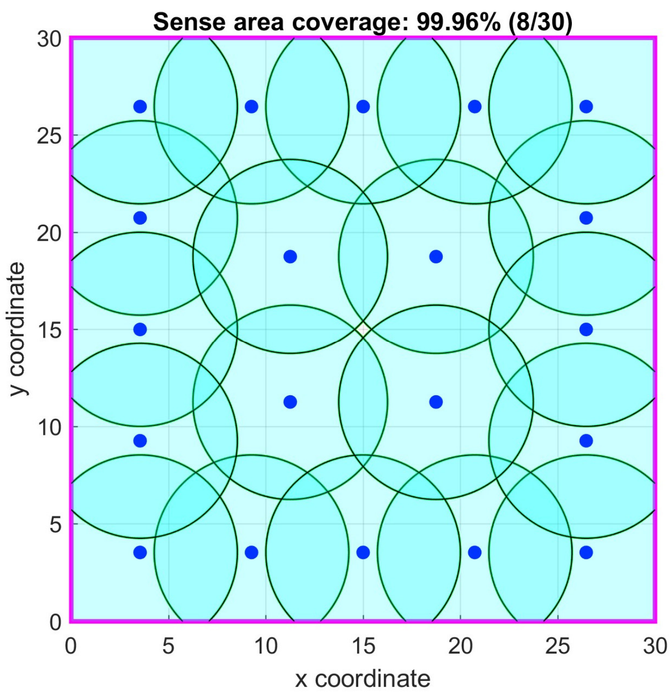


Figure 22. Case study 7: optimal node locations.

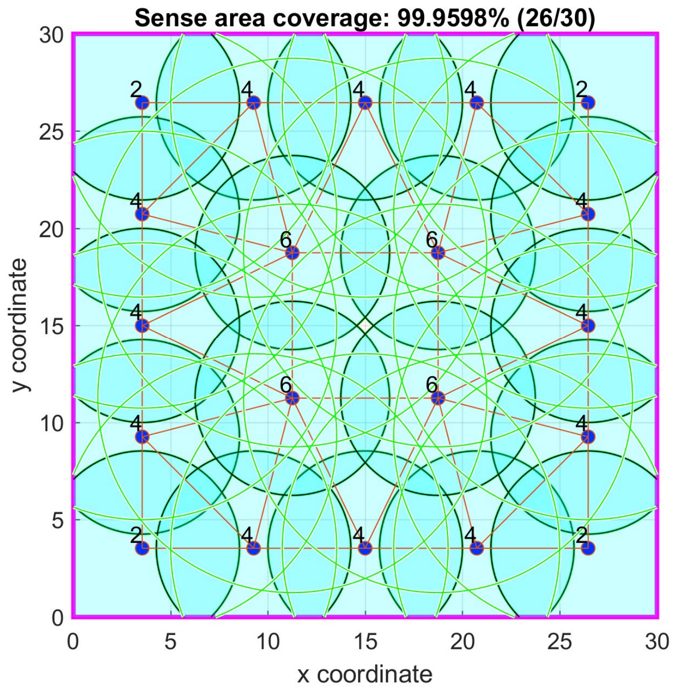


Figure 23. Case study 7: optimal node locations with 1-connectivity.

In Figure 24, the objective function value iterations for both requirements are illustrated.

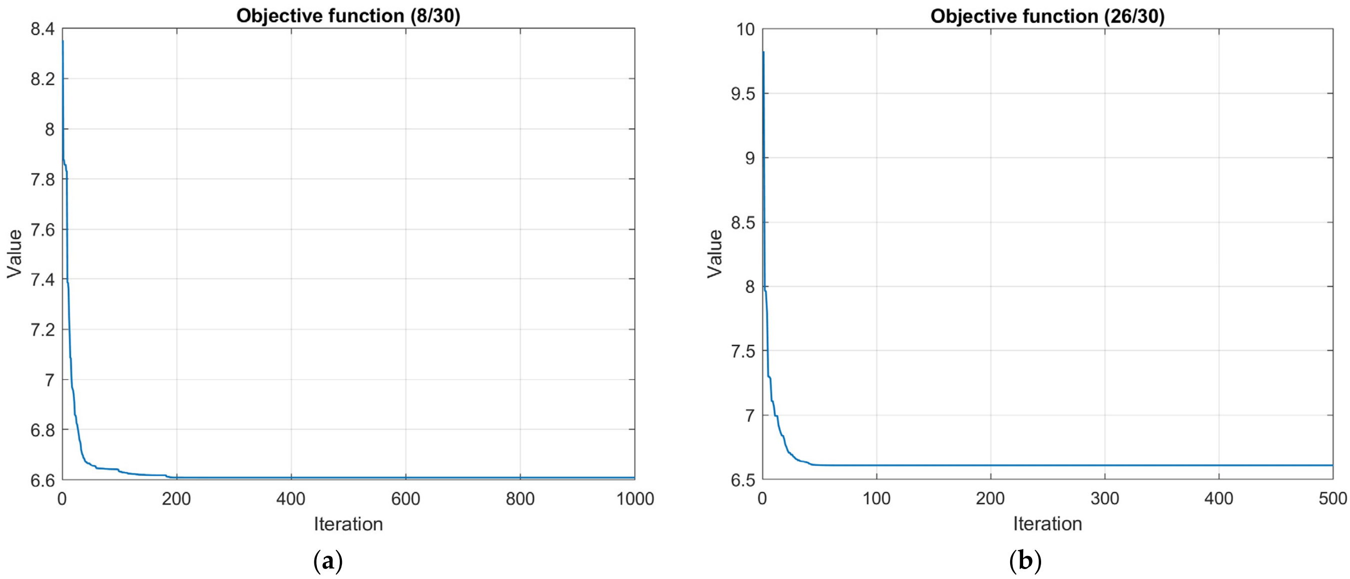


Figure 24. Case study 7: objective function iterations: (**a**) no connectivity; (**b**) 1-connectivity.

4.8. Summary of Performance Evaluation

Below is a concise overview of the key simulation outcomes for each case. All the case study results are compared with the results of their corresponding benchmark cases found in [39,40].

- Case 1
  No connectivity constraint: An improvement in coverage mean values, standard deviation and best fitness results compared to both benchmark GA and PSO algorithms.
  1-connectivity constraint: The constraint was attained, and there was also an im-

provement in coverage mean values, standard deviation and the best fitness results compared to both benchmark GA and PSO algorithms.

- Case 2
  No connectivity constraint: An improvement over GA and PSO benchmarks in coverage mean values. An improvement over the PSO benchmark in standard deviation. An improvement over GA and PSO benchmarks in the best fitness results.
  1-connectivity constraint: The constraint was attained with no improvements in coverage mean values, standard deviation and the best fitness results.
- Case 3
  No connectivity constraint: An improvement over GA and PSO benchmarks in coverage mean values. An improvement over GA and PSO benchmarks in the best fitness results.
  1-connectivity constraint: The constraint was attained with improvement in the best fitness results over GA and PSO benchmarks.
- Case 4
  No connectivity constraint: An improvement over the GA benchmark in coverage mean values. An improvement over the PSO benchmark in standard deviation. An improvement over the GA benchmark in the best fitness results.
  1-connectivity constraint: The constraint was attained with improvement in the best fitness results over the GA benchmark.
- Case 5
  No connectivity constraint: An improvement in coverage mean values and standard deviations compared to both GA and PSO benchmark algorithms.
  1-connectivity constraint: The constraint was attained, and there was also an improvement of coverage mean values and standard deviations compared to both benchmark GA and PSO algorithms.
- Case 6
  No connectivity constraint: An improvement in coverage mean values and standard deviations compared to both GA and PSO benchmark algorithms.
  1-connectivity constraint: The constraint was attained with no improvements in coverage mean values and standard deviation results.
- Case 7
  No connectivity constraint: An improvement in coverage mean values and standard deviations compared to both GA and PSO benchmark algorithms.

1-connectivity constraint: The constraint was attained, and there was also an improvement of coverage mean values and standard deviations compared to both benchmark GA and PSO algorithms.

5. Conclusions and Future Work

In this paper, a novel Particle Swarm Optimization (PSO) approach was proposed for the optimal placement of a predefined number of sensor nodes within a square target area. This approach adheres to k-coverage and 1-connectivity constraints. Also, a new objective function derived from circle packing geometric problems is introduced. This function serves as an alternative to traditional area coverage minimization objectives and simplifies implementation. The objective function resembles the repulsion force-based methods but with a much cleaner definition and simpler implementation. A limitation of the method is that only 1-connectivity is ensured, not m-connectivity. Also, the number k of redundant sensor coverage is the same for all target points of interest. Moreover, the k-coverage of the POIs is tight, meaning that the algorithm strives for covered area maximization, i.e., all POIs are k-covered but they might be positioned at the edge of the k sensors ranges.

The novel method was tested against seven benchmark test cases presented in [40]. Two of the test cases included a 3-coverage constraint of six predefined points in the region, and two test cases required different sensing and communication ranges of the sensor nodes. Each test case was executed 30 times and involved the computation of the mean and standard deviation of the area coverage percentage, thereby accounting for the stochastic nature of the PSO algorithm.

The findings of the simulation tests indicate that in six out of the seven test cases, the proposed PSO method was better in terms of either or both the mean value and the standard deviation of the area coverage percentage. The statistical significance of this was confirmed using the t-test methodology. In one test case involving the 3-coverage of six points of interest, the proposed methodology demonstrated better performance compared to the Genetic Algorithm-based approach used in [40]. This test case is quite challenging and computationally intensive, especially with the 1-connectivity constraint. It is anticipated that further tuning of the PSO parameters and changing its implementation methodology could enhance its performance. Such changes might include using a variable inertia weight w, adjusting other PSO parameters across algorithm iterations or selecting a different static or dynamic population topology.

The presented methodology showed promising results and could be generalized and extended in various different aspects in the future in the following ways:

- Generalization and testing of the proposed method on various geometric shapes beyond square regions. This involves adapting the algorithm to handle irregular boundaries or even obstacles within the deployment area.
- Generalization of sensor node sensing patterns, i.e., make changes to accommodate diverse sensor node sensing patterns, not limited to circular ones. This could be achieved by incorporating directional parameters into the objective function, thus making possible the modeling of anisotropic sensing fields, which is common in practical WSN applications.
- General communication radiation patterns, i.e., extending the new method to handle realistic, non-circular communication range patterns. This could be achieved by employing Laplacian eigenvalue methodologies used extensively in robotics [58,59] instead of the direct search methods used in this work. This could also facilitate the incorporation of m-connectivity constraints, thus improving network robustness and fault tolerance.
- Application in other metaheuristics, i.e., to investigate the effectiveness of the proposed objective function within other metaheuristic optimization frameworks, except the PSO method.
- Minimal sensor deployment, i.e., to adapt this method to not only maximize coverage but also to determine the minimal number of sensor nodes required for a given coverage and connectivity application. This could have significant implications for cost reduction and resource efficiency.
- Energy consumption modeling, i.e., to try to integrate energy models into the proposed optimization framework, as energy efficiency impacts network lifespan [60].
- Robustness against failures, i.e., to assess the resilience of the deployment strategy under varying environmental conditions and node failures and incorporate dynamic adaptation mechanisms to enhance its performance.
- Validation of this method through real-world deployments in order to assess the practical feasibility and performance of this approach in operational environments.

Author Contributions: Conceptualization, G.S. and D.K.; methodology, G.S. and D.K.; software, G.S.; validation, G.S. and D.K.; formal analysis, G.S.; investigation, G.S. and D.K.; resources, G.S. and D.K.; data curation, G.S. and D.K.; writing—original draft preparation, G.S.; writing—review and editing G.S. and D.K.; visualization, G.S. and D.K.; supervision, D.K.; project administration, G.S. and D.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless Sensor Networks: A Survey. *Comput. Netw.* **2002**, *38*, 393–422. [CrossRef]
2. Yick, J.; Mukherjee, B.; Ghosal, D. Wireless Sensor Network Survey. *Comput. Netw.* **2008**, *52*, 2292–2330. [CrossRef]
3. Kandris, D.; Nakas, C.; Vomvas, D.; Koulouras, G. Applications of Wireless Sensor Networks: An Up-to-Date Survey. *Appl. Syst. Innov.* **2020**, *3*, 14. [CrossRef]
4. Sunehra, D.; Rajasri, S. Automatic Street Light Control System Using Wireless Sensor Networks. In Proceedings of the IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), Chennai, India, 21–22 September 2017. [CrossRef]
5. Papadakis, N.; Koukoulas, N.; Christakis, I.; Stavrakas, I.; Kandris, D. An IoT-Based Participatory Antitheft System for Public Safety Enhancement in Smart Cities. *Smart Cities* **2021**, *4*, 919–937. [CrossRef]
6. Noel, A.; Abdaoui, A.; Elfouly, T.; Ahmed, M.H.; Badawy, A.; Shehata, M. Structural Health Monitoring Using Wireless Sensor Networks: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 1403–1423. [CrossRef]
7. Orfanos, V.A.; Kaminaris, S.D.; Papageorgas, P.; Piromalis, D.; Kandris, D. A Comprehensive Review of IoT Networking Technologies for Smart Home Automation Applications. *J. Sens. Actuator Netw.* **2023**, *12*, 30. [CrossRef]
8. Nikolidakis, S.A.; Kandris, D.; Vergados, D.D.; Douligeris, C. Energy efficient automated control of irrigation in agriculture by using wireless sensor networks. *Comput. Electron. Agric.* **2015**, *113*, 154–163. [CrossRef]
9. Arshad, J.; Siddiqui, T.A.; Sheikh, M.I.; Waseem, M.S.; Nawaz, M.A.B.; Eldin, E.T.; Rehman, A.U. Deployment of an Intelligent and Secure Cattle Health Monitoring System. *Egypt. Inform. J.* **2023**, *24*, 265–275. [CrossRef]
10. Jabeen, T.; Jabeen, I.; Ashraf, H.; Jhanjhi, N.Z.; Yassine, A.; Hossain, M.S. An Intelligent Healthcare System Using IoT in Wireless Sensor Network. *Sensors* **2023**, *23*, 5055. [CrossRef]
11. Majid, M.; Habib, S.; Javed, A.R.; Rizwan, M.; Srivastava, G.; Gadekallu, T.R.; Lin, J.C.-W. Applications of Wireless Sensor Networks and Internet of Things Frameworks in the Industry Revolution 4.0: A Systematic Literature Review. *Sensors* **2022**, *22*, 2087. [CrossRef]
12. Christakis, I.; Tsakiridis, O.; Kandris, D.; Stavrakas, I. Air Pollution Monitoring via Wireless Sensor Networks: The Investigation and Correction of the Aging Behavior of Electrochemical Gaseous Pollutant Sensor nodes. *Electronics* **2023**, *12*, 1842. [CrossRef]
13. Pantazis, N.A.; Nikolidakis, S.A.; Kandris, D.; Vergados, D.D. An Automated System for Integrated Service Management in Emergency Situations. In Proceedings of the 2011 15th Panhellenic Conference on Informatics, Kastonia, Greece, 30 September–2 October 2011; pp. 154–157.
14. Đurišić, M.P.; Tafa, Z.; Dimić, G.; Milutinović, V. A Survey of military applications of wireless sensor networks. In Proceedings of the 2012 Mediterranean Conference on Embedded Computing (MECO), Bar, Montenegro, 19–21 June 2012; pp. 196–199.
15. Kandris, D.; Anastasiadis, E. Advanced Wireless Sensor Networks: Applications, Challenges and Research Trends. *Electronics* **2024**, *13*, 2268. [CrossRef]
16. Kandris, D.; Alexandridis, A.; Dagiuklas, T.; Panaousis, E.; Vergados, D.D. Multiobjective Optimization Algorithms for Wireless Sensor Networks. *Wirel. Commun. Mob. Comput.* **2020**, *2020*, 4652801. [CrossRef]
17. Al-Karaki, J.N.; Gawanmeh, A. The Optimal Deployment, Coverage, and Connectivity Problems in Wireless Sensor Networks: Revisited. *IEEE Access* **2017**, *5*, 18051–18065. [CrossRef]
18. Tripathi, A.; Gupta, H.P.; Dutta, T.; Mishra, R.; Shukla, K.K.; Jit, S. Coverage and Connectivity in WSNs: A Survey, Research Issues and Challenges. *IEEE Access* **2018**, *6*, 26971–26992. [CrossRef]
19. Farsi, M.; Elhosseini, M.A.; Badawy, M.; Arafat Ali, H.; Zain Eldin, H. Deployment Techniques in Wireless Sensor Networks, Coverage and Connectivity: A Survey. *IEEE Access* **2019**, *7*, 28940–28954. [CrossRef]
20. Evangelakos, E.A.; Kandris, D.; Rountos, D.; Tselikis, G.; Anastasiadis, E. Energy Sustainability in Wireless Sensor Networks: An Analytical Survey. *J. Low Power Electron. Appl.* **2022**, *12*, 65. [CrossRef]
21. Bohloulzadeh, A.; Rajaei, M. A Survey on Congestion Control Protocols in Wireless Sensor Networks. *Int. J. Wirel. Inf. Netw.* **2020**, *27*, 365–384. [CrossRef]
22. Chiwariro, R.; Thangadurai, N. Quality of Service Aware Routing Protocols in Wireless Multimedia Sensor Networks: Survey. *Int. J. Inf. Technol.* **2020**, *14*, 789–800. [CrossRef]
23. Yu, J.-Y.; Lee, E.; Oh, S.-R.; Seo, Y.-D.; Kim, Y.-G. A Survey on Security Requirements for WSNs: Focusing on the Characteristics Related to Security. *IEEE Access* **2020**, *8*, 45304–45324. [CrossRef]
24. Randhawa, S.; Jain, S. Data Aggregation in Wireless Sensor Networks: Previous Research, Current Status and Future Directions. *Wirel. Pers. Commun.* **2017**, *97*, 3355–3425. [CrossRef]
25. Adday, G.H.; Subramaniam, S.K.; Zukarnain, Z.A.; Samian, N. Fault Tolerance Structures in Wireless Sensor Networks (WSNs): Survey, Classification, and Future Directions. *Sensors* **2022**, *22*, 6041. [CrossRef]
26. Sneha, V.; Nagarajan, M. Localization in Wireless Sensor Networks: A Review. *Cybern. Inf. Technol.* **2020**, *20*, 3–26. [CrossRef]
27. Yadav, J.; Mann, S. Coverage in wireless sensor networks: A survey. *Int. J. Electron. Comput. Sci. Eng.* **2013**, *2*, 465–471.
28. Ammari, H.M. Coverage in Wireless Sensor Networks: A Survey. *Netw. Protoc. Algorithms* **2010**, *2*, 27–53. [CrossRef]
29. Cardei, M. Coverage Problems in Sensor Networks. In *Handbook of Combinatorial Optimization*; Springer: New York, NY, USA, 2013; pp. 899–927. [CrossRef]

30. Wang, Y.; Zhang, Y.; Liu, J.; Bhandari, R. Coverage, Connectivity, and Deployment in Wireless Sensor Networks. In *Recent Development in Wireless Sensor and Ad-hoc Networks*; Signals and Communication Technology; Springer: New Delhi, India, 2014; pp. 25–44. [CrossRef]
31. Amutha, J.; Sharma, S.; Nagar, J. WSN Strategies Based on Sensors, Deployment, Sensing Models, Coverage and Energy Efficiency: Review, Approaches and Open Issues. *Wirel. Pers. Commun.* **2019**, *111*, 1089–1115. [CrossRef]
32. Ammari, H.M. *Theory and Practice of Wireless Sensor Networks: Cover, Sense, and Inform*; Springer International Publishing: Cham, Switzerland, 2022. [CrossRef]
33. Ojeda, F.; Méndez, D.; Fajardo, A.; Ellinger, F. On Wireless Sensor Network Models: A Cross-Layer Systematic Review. *J. Sens. Actuator Netw.* **2023**, *12*, 50. [CrossRef]
34. Engelbrecht, A.P. Particle Swarm Optimization. In *Computational Intelligence: An Introduction*; John Wiley: Chichester, UK; Hoboken, NJ, USA, 2007; pp. 289–358.
35. Poli, R.; Kennedy, J.; Blackwell, T. Particle Swarm Optimization. *Swarm Intell.* **2007**, *1*, 33–57. [CrossRef]
36. Ezugwu, A.E.; Shukla, A.K.; Nath, R.; Akinyelu, A.A.; Agushaka, J.O.; Chiroma, H.; Muhuri, P.K. Metaheuristics: A Comprehensive Overview and Classification along with Bibliometric Analysis. *Artif. Intell. Rev.* **2021**, *54*, 4237–4316. [CrossRef]
37. Brass, P. Geometric Problems on Coverage in Sensor Networks. In *Geometry—Intuitive, Discrete, and Convex*; Bolyai Society mathematical studies; Springer: Berlin/Heidelberg, Germany, 2013; pp. 91–108. [CrossRef]
38. Xiaoling, W.; Lei, S.; Jie, Y.; Hui, X.; Cho, J.; Lee, S. Swarm Based Sensor Deployment Optimization in Ad Hoc Sensor Networks. In *Embedded Software and Systems*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2005; pp. 533–541. [CrossRef]
39. Aziz, N.; Mohemmed, A.; Alias, M.H. A Wireless Sensor Network Coverage Optimization Algorithm Based on Particle Swarm Optimization and Voronoi Diagram. In Proceedings of the 2009 International Conference on Networking, Sensing and Control, Okayama, Japan, 26–29 March 2009. [CrossRef]
40. Tarnaris, K.; Preka, I.; Kandris, D.; Alexandridis, A. Coverage and K-Coverage Optimization in Wireless Sensor Networks Using Computational Intelligence Methods: A Comparative Study. *Electronics* **2020**, *9*, 675. [CrossRef]
41. Tossa, F.U.; Abdou, W.; Ansari, K.; Ezin, E.C.; Gouton, P. Area Coverage Maximization under Connectivity Constraint in Wireless Sensor Networks. *Sensors* **2022**, *22*, 1712. [CrossRef] [PubMed]
42. Cao, L.; Yue, Y.; Cai, Y.; Zhang, Y. A Novel Coverage Optimization Strategy for Heterogeneous Wireless Sensor Networks Based on Connectivity and Reliability. *IEEE Access* **2021**, *9*, 18424–18442. [CrossRef]
43. Wang, Y.; Li, M. Coverage Control Optimization Algorithm for Wireless Sensor Networks Based on Combinatorial Mathematics. *Math. Probl. Eng.* **2021**, *2021*, 6066379. [CrossRef]
44. Wang, X.; Wang, S.; Bi, D. Virtual Force-Directed Particle Swarm Optimization for Dynamic Deployment in Wireless Sensor Networks. In *Advanced Intelligent Computing Theories and Applications—With Aspects of Theoretical and Methodological Issues*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 292–303. [CrossRef]
45. Dirafzoon, A.; Salehizadeh, S.M.A.; Emrani, S.; Menhaj, M.B. Virtual Force Based Individual Particle Optimization for Coverage in Wireless Sensor Networks. In Proceedings of the CCECE 2010, Calgary, AB, Canada, 2–5 May 2010; pp. 1–4. [CrossRef]
46. Mougou, K.; Mahfoudh, S.; Minet, P.; Laouiti, A. Redeployment of Randomly Deployed Wireless Mobile Sensor Nodes. In Proceedings of the 2012 IEEE Vehicular Technology Conference (VTC Fall), Quebec City, QC, Canada, 3–6 September 2012; pp. 1–5. [CrossRef]
47. Hanh, N.T.; Binh, H.T.T.; Hoai, N.X.; Palaniswami, M.S. An Efficient Genetic Algorithm for Maximizing Area Coverage in Wireless Sensor Networks. *Inf. Sci.* **2019**, *488*, 58–75. [CrossRef]
48. Du, Y. Method for the Optimal Sensor Deployment of WSNs in 3D Terrain Based on the DPSOVF Algorithm. *IEEE Access* **2020**, *8*, 140806–140821. [CrossRef]
49. Qi, X.; Li, Z.; Chen, C.; Liu, L. A Wireless Sensor Node Deployment Scheme Based on Embedded Virtual Force Resampling Particle Swarm Optimization Algorithm. *Appl. Intell.* **2021**, *52*, 7420–7441. [CrossRef]
50. Zhao, Q.; Li, C.; Zhu, D.; Xie, C. Coverage Optimization of Wireless Sensor Networks Using Combinations of PSO and Chaos Optimization. *Electronics* **2022**, *11*, 853. [CrossRef]
51. Amer, D.A.; Soliman, S.A.; Hassan, A.F.; Zamel, A.A. Enhancing Connectivity and Coverage in Wireless Sensor Networks: A Hybrid Comprehensive Learning-Fick's Algorithm with Particle Swarm Optimization for Router Node Placement. *Neural Comput. Appl.* **2024**, *36*, 21671–21702. [CrossRef]
52. Szabó, P.G.; Markót, M.C.; Csendes, T. Global Optimization in Geometry—Circle Packing into the Square. In *Essays and Surveys in Global Optimization*; Springer: Boston, MA, USA, 2005; pp. 233–265. [CrossRef]
53. Nurmela, K.J.; Östergård, P.R.J. Packing up to 50 Equal Circles in a Square. *Discret. Comput. Geom.* **1997**, *18*, 111–120. [CrossRef]
54. Cormen, T.H.; Leiserson, C.E.; Rivest, R.L.; Stein, C. *Introduction to Algorithms*; The Mit Press: Cambridge, MA, USA, 2022.
55. Pedersen, M.E.H.; Chipperfield, A.J. Simplifying Particle Swarm Optimization. *Appl. Soft Comput.* **2010**, *10*, 618–628. [CrossRef]
56. Kolda, T.G.; Lewis, R.M.; Torczon, V. Optimization by Direct Search: New Perspectives on Some Classical and Modern Methods. *SIAM Rev.* **2003**, *45*, 385–482. [CrossRef]
57. Find Minimum of Function Using Pattern Search—MATLAB Patternsearch. Available online: https://www.mathworks.com/help/gads/patternsearch.html (accessed on 28 October 2024).

58. Olfati-Saber, R.; Fax, J.A.; Murray, R.M. Consensus and Cooperation in Networked Multi-Agent Systems. *Proc. IEEE* **2007**, *95*, 215–233. [CrossRef]
59. Zavlanos, M.M.; Egerstedt, M.B.; Pappas, G.J. Graph-Theoretic Connectivity Control of Mobile Robot Networks. *Proc. IEEE* **2011**, *99*, 1525–1540. [CrossRef]
60. Nakas, C.; Kandris, D.; Visvardis, G. Energy Efficient Routing in Wireless Sensor Networks: A Comprehensive Survey. *Algorithms* **2020**, *13*, 72. [CrossRef]