IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 29 September 2022, accepted 18 October 2022, date of publication 26 October 2022, date of current version 3 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3217200

SURVEY

A Survey of Artificial Intelligence Based WSNs Deployment Techniques and Related Objectives Modeling

KHAOULA ZAIMEN[1,2], MOHAMED-EL-AMINE BRAHMIA[1], (Member, IEEE), LAURENT MOALIC[2], ABDELHAFID ABOUAISSA[2], AND LHASSANE IDOUMGHAR[2], (Member, IEEE)

[1]CESI LINEACT, 67380 Strasbourg, France
[2]UR 7499, Institut de Recherche en Informatique, Mathématique, Automatique et Signal, 68093 Mulhouse Cedex, France

Corresponding author: Khaoula Zaimen (khaoula.zaimen@uha.fr)

This work was supported in part by the CESI LINEACT Laboratory, Strasbourg, France; and in part by the Institut de Recherche en Informatique, Mathématique, Automatique et Signal, University of Haute-Alsace, Mulhouse, France.

ABSTRACT Recent advances in hardware and communication technologies have accelerated the deployment of billions of wireless sensors. This transformation has created a wide range of applications adapted to the evolving trends of our daily life requirements. Wireless sensor networks (WSNs) could be deployed in several target areas including buildings, forests, oceans, and smart cities. Nevertheless, finding the optimal location for each sensor node is a challenging task, typically when the environment involves heterogeneous obstacles. Many approaches and methods have been proposed to deal with the problem of WSN deployment, each addressing one or more objectives and constraints, such as network coverage, lifetime, connectivity, and energy consumption. The purpose of this survey paper is to provide the needed background to understand and study the WSNs deployment problem with a focus on its two key aspects: the optimization model and the solving methods based on artificial intelligence (AI). Additionally, it covers recent works on WSNs deployment and identifies their advantages and limitations. Furthermore, simulation experiments were carried out to compare the performance of widely used algorithms in the context of WSNs deployment problem, primarily genetic algorithm, particle swarm optimization, flower pollination, and ant colony optimization. Finally, this paper discusses and highlights several open challenges and research issues that should be explored in the future.

INDEX TERMS Artificial intelligence, machine learning, metaheuristics, objectives modeling, optimization model, wireless sensor networks, WSNs deployment, sensing models.

I. INTRODUCTION

Wireless sensor network technology represents a promising paradigm of networking and computing. It consists of homogeneous or heterogeneous sensor nodes that sense physical environments and transmit data to a base station. A sensor node comprises typically four units [1]: sensing unit, processing unit, communication unit, and power unit (see Fig. 1), yet additional components, such as mobilizer and location finding system, may be added to fulfill other tasks. The sensing unit contains a sensor and an analog-to-digital converter (ADC). Thus, sensors detect events that occur within their sensing range and then convert analog signals into digital signals for further analysis in the processing unit. According to the sensor sub-unit, we distinguish two types of sensor nodes: contact and noncontact sensor nodes. Contact sensor nodes require physical contact with the target in order to perform their measurements; examples of contact sensors include thermocouples, thermistors, and resistance temperature detectors. In contrast, noncontact sensors rely on physical effects that do not require any physical contact with the target, such as the Hall effect and the Magnetoresistive effect.

The associate editor coordinating the review of this manuscript and approving it for publication was Emanuele Crisostomi.

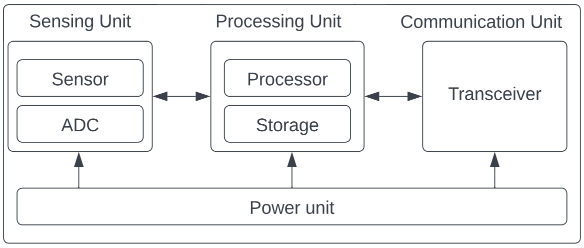


FIGURE 1. **Sensor node architecture.**

The second component is the processing unit, which consists of a processor for executing programs and a tiny storage unit for storing gathered data. The communication unit connects the sensor node to the network by enabling data transmission and reception. The last component is the power unit which supplies power to all operating parts of the sensor node.

Sensor nodes are classified based on three principal factors: the embedded sensing technology, the sensing type, and the sensing direction. Fig. 2 depicts the sensor sub-classes regarding each factor. Sensing technology defines the sensor behavior in response to a physical, chemical, or biological stimulus. This behavior can be translated into a variation of its electrical resistance, capacitance, inductance, etc., and then mapped into a suitable output value [2]. In addition, the sensor behavior is also affected by the sensing direction. A directional sensor node can perceive in only one direction at a fixed view [3]. In contrast, an omnidirectional sensor node has the ability to sense in all directions [3]. Fig. 3 illustrates an example of each type.

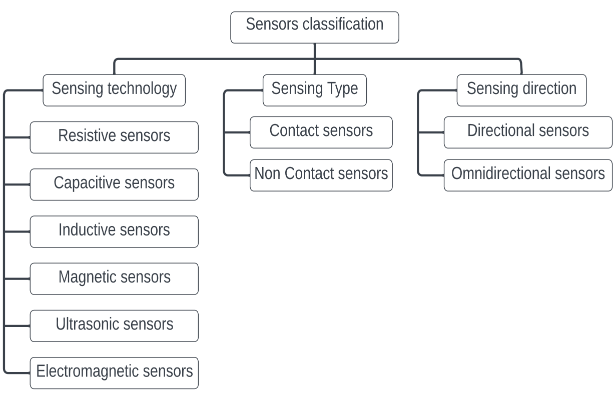
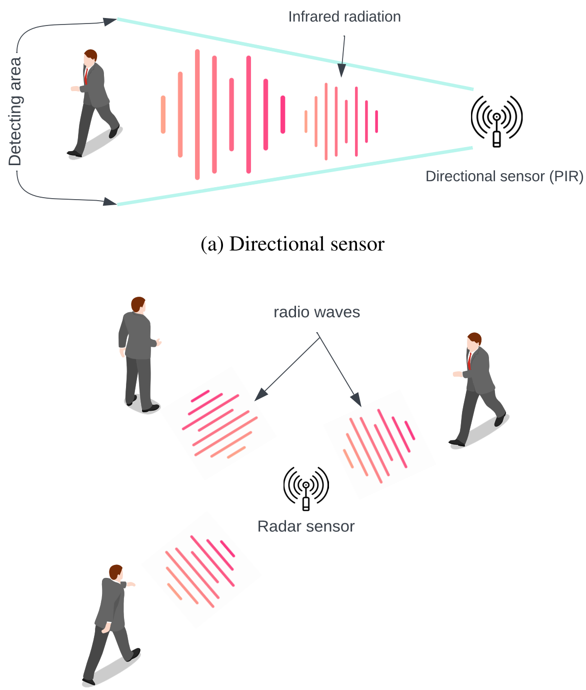


FIGURE 2. **Classification of sensor nodes.**

Diverse sorts of sensor nodes are available on the market, including light sensors, temperature sensors, pollution sensors, pressure sensors, gas sensors, etc. Therefore, they are presently employed in many fields, namely military monitoring and tracking, health care, industry, environmental monitoring, smart agriculture, and smart buildings. Each application domain has its own specifications and requirements for the quality of service (QoS) metrics that should be met when deploying the sensor network.



(a) Directional sensor

(b) Omnidirectional sensor

FIGURE 3. **Directional sensor vs Omnidirectional sensor.**

A. MOTIVATION

The performance of a sensor network is strongly linked to its deployment scheme. Thus, several factors should be considered when computing the positions of the sensors, such as the connectivity constraint, the coverage holes, the deployment cost, and energy consumption [4]. Moreover, the reliability of the final deployment can be reinforced by incorporating real characteristics of the deployment environment, including the dimension, shape, obstacles heterogeneity, and obstacles characteristics.

Literature has addressed the deployment problem of WSNs with a variety of assumptions. Some of the proposed solutions seek to meet multiple conflicting objectives simultaneously by finding the best trade-off. Others were based on strong assumptions regarding the network and the area of interest to reduce the complexity of the problem. This simplification can result in false estimates of the performance of deployed networks. Hence, we believe this area of research still requires further effort in order to obtain a reliable and directly applicable deployment in a real-world setting. In this context, it is important to examine all aspects and parameters involved in the deployment design, from the optimization model to the used method.

Through this paper, we aim to provide the research community with a complete survey on the WSN deployment problem. Our main motivation is to help researchers quickly understand the current state of the art, identify which topics need further research, and assist them in the process of conceiving their solutions. To the best of our knowledge, this is the first paper that addresses the deployment problem comprehensively. Compared to similar research papers [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], our paper presents a general reference optimization model and discusses the application of AI to solve it. We have extended our scope to include a wide variety of AI-based methods. Our study has included metaheuristics not commonly discussed in survey papers concerning sensor deployment, such as Cuckoo Search, Grey Wolf Optimizer, Bat algorithm, and Flower Pollination algorithm. Furthermore, we have examined hybrid metaheuristics, machine learning, and fuzzy logic-based approaches in the same context.

Along with the aforementioned contributions, we believe that the originality of the present work lies also in the inclusion of important factors that are typically neglected in the literature, namely the impact of the target area modeling and the sensing model on the final deployment scheme.

B. CONTRIBUTIONS OF THE SURVEY

In this paper, we thoroughly study both the optimization model and the AI-based approaches used to address the deployment of WSNs. Additionally, we carry out simulation experiments to evaluate the performance of six metaheuristics in finding the optimal deployment. This paper's significant contributions are summarized below:

- We introduce a general reference optimization model for WSNs deployment problem including the decision variables, objective functions modeling, and feasible constraints.
- we present an overview of the sensing mathematical models with their parameters and their limits.
- We discuss the target area modeling and highlight its impact on the final result.
- We survey the AI-based solutions for WSNs deployment problem and present relevant statistical analyses to better understand the current research directions on this topic.
- We implement and test the most commonly used metaheuristics in addressing WSNs deployment problem to evaluate and compare their performance.
- We emphasize the open issues and challenges related to WSNs deployment problem.

C. ORGANIZATION OF THE SURVEY

The remaining sections of this survey are organized as follows: Section III discusses a general reference optimization model for WSNs deployment problem, including the decision variables, the main WSN objective functions modeling, and sensing models existing in the literature. Target area modeling is described in Section IV. Section V presents the AI-based approaches for solving the WSN deployment problem. It mainly classifies them into three categories: metaheuristic-based techniques, hybrid metaheuristic-based techniques, and Machine learning (ML) and fuzzy logic (FL) based techniques. Section VI is devoted to simulations and comparisons of the widely used metaheuristics in solving the problem of WSN deployment as well as an analysis of their obtained outcomes. In the final section, we recap the work and discuss the challenges and open issues related to sensor nodes deployment. Fig. 4 depicts the organization of the present survey.

Table 1 summarizes the abbreviations used in this paper.

TABLE 1. Table of abbreviations.

Abbreviation	Description
AI	Artificial intelligence
WSN	Wireless Sensor Network
EA	Evolutionary Algorithms
SI	Swarm Intelligence
GA	Genetic Algorithm
PSO	Particle Swarm Optimization
NSGA-II	Non-dominated Sorting Genetic Algorithm
MOEA/D	Multi-objective evolutionary algorithm based on decomposition
CS	Cuckoo search algorithm
ML	Machine Learning
FL	Fuzzy Logic
ACO	Ant colony optimization
ALO	Ant lion optimizer
ABC	Artificial Bee colony optimization
GWO	Grey wolf Optimizer
BA	Bat algorithm
FPA	Flower pollination algorithm
GOA	Grasshopper Optimization Algorithm
BIM	Building Information Modeling
DEM	Digital Elevation Model
QoS	Quality of Service

II. RELATED SURVEYS

WSNs deployment problem has been tackled in various surveys, most of them focused on the deployment strategies and approaches without detailing the optimization model and the environment modeling. The authors of [7] explained the related concepts of WSN coverage. They discussed its design concerns and challenges, such as sensor nodes mobility and heterogeneity, as well as the dimension of the target region. The authors also surveyed two ways to solve coverage problems: computational geometry-based approaches for deterministic deployment and probabilistic approaches for random deployment.

In [8], the authors discussed the use of metaheuristics to solve the WSN deployment problem. They believe that modifying the transition, local search, and determination operators improve the quality of the final solution. Additionally, the authors argued that solution representation, solution initialization, and hybridization of algorithms are other trends to enhance the performance of the metaheuristics in solving the problem. The systematic literature review in [9] surveyed WSN deployment strategies published between 2004 and 2016 and classified them into two main categories: deterministic and non-deterministic deployment. Besides, the authors reported some statistics regarding journals and conferences where studied papers are published. The survey in [4] focuses on WSN deployment strategies based on four objectives that must be optimized while computing sensor nodes positions. These objectives are coverage maximization, connectivity

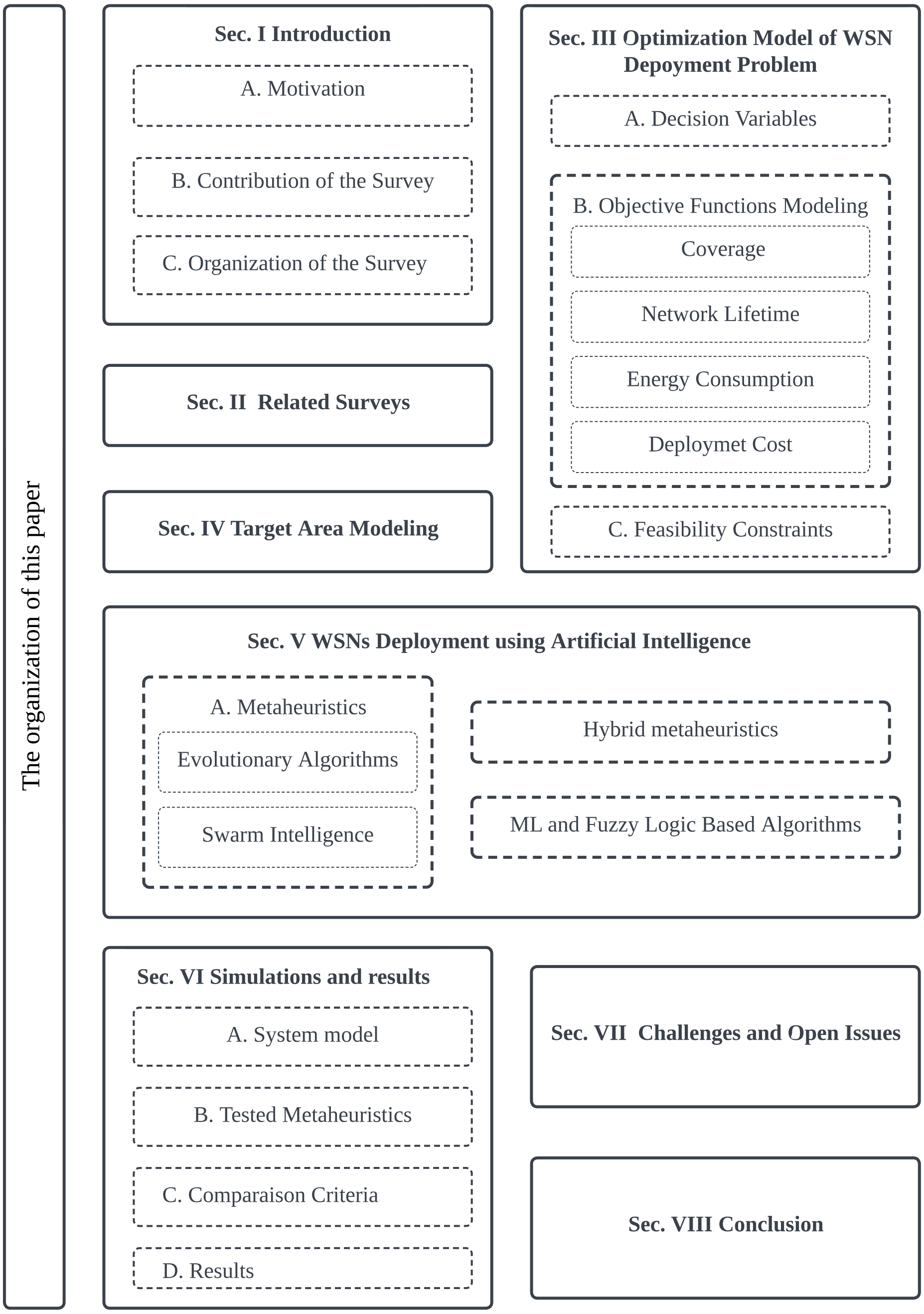


FIGURE 4. **The organization of the survey.**

enhancement, energy efficiency, and lifetime optimization. The authors also investigated some approaches proposed in the literature to deal with each objective and identified their advantages, weaknesses, and related open issues. In [10], the authors discussed four problem domains in WSN, which are: optimal coverage, data aggregation, sensor localization, energy-efficient clustering, and routing. Moreover, they presented a couple of their proposed solutions based on three nature-inspired algorithms: genetic algorithm, particle swarm optimization, and ant colony optimization. In a second contribution, the authors evaluated the ability of two metaheuristic algorithms to achieve optimum coverage: the Lion Optimization (LO) algorithm and an improved Genetic Algorithm combined with the Binary Ant Colony Algorithm (GA-BACA). The results of their evaluation confirm that LO outperforms GA-BACA in both network coverage and convergence rate.

In the survey [11], authors mainly focused on the coverage deployment strategies for dynamic coverage based on virtual force and Voronoi diagram; and static coverage presented by efficient coverage area, k-coverage, and path coverage. They also discussed two sleep scheduling mechanisms for preserving sensor network energy: disjoint dominating sets and self-scheduling strategy, as well as the adjustable coverage radius as a power-saving technique in target coverage. Moreover, the research highlighted certain research problems in WSN coverage and connectivity that must be addressed. The paper presented in [6] provides an overview of the multi-objective optimization in WSN. It elaborates the mathematical models of some objective functions in WSN such as coverage, network connectivity, network lifetime, and energy consumption. Further, the authors reviewed the multi-objective optimization approaches for solving multi-objective problems in WSNs. They focused on the mathematical programming-based scalarization methods, including the linear weighted-sum method and epsilon-constraints methods, as well as the nature-inspired metaheuristic algorithms such as evolutionary algorithms and swarm intelligence optimization. In [12], authors reviewed algorithms used in deterministic WSN deployment. They classified them into four main mathematical approaches: genetic algorithms, computational geometry, artificial potential fields-based algorithms, and particle swarm optimization. Next, the authors analyzed the proposed solutions in the literature based on each approach and compared those solutions in terms of objectives, sensing models, and sensor types. Another paper presented in [13] outlines the connectivity and coverage challenges and concerns and organizes the techniques for WSN coverage maximization in four major parts: computational geometry-based methods, force-based techniques, grid-based techniques, and metaheuristic-based techniques. Also, the authors listed and compared the simulators used in WSN and explored the open-research issues and directions. Table 2 summarizes the main contributions of the surveys presented in this section.

III. OPTIMIZATION MODEL OF WSN DEPLOYMENT PROBLEM

WSN design and deployment is a complex task since it has a direct impact on the performance of the network and consequently on the applications using it. Furthermore, several critical applications such as health care, military, or even environment monitoring applications require a specific degree of quality of service (QoS), namely coverage, cost, connectivity, and network lifetime. Therefore, the research community has suggested several optimization objectives to tackle the problem of WSNs deployment. Each objective may have various mathematical models, each with a certain accuracy level.

In this section, we will present a general reference optimization model for the WSN deployment problem, including the decision variables, the salient optimization objectives, and the feasible constraints considered in the WSNs deployment problem.

A. DECISION VARIABLES

Decision variables of the WSN deployment problem refer to the locations of sensor nodes in the target area. There are mainly two representations of the decision variables, the vector representation, and the grid representation. In the former representation, the deployment scheme is defined as an array of Cartesian coordinates where the cell i represents the position (x_i, y_i) of the sensor i. Each decision variable (x_i, y_i) must fulfill the constraints of upper and lower bounds of the area of interest. In the latter representation, the deployment solution is defined by a grid of L rows and W columns. Each cell c_i of the grid corresponds to xm^2 in the real environment. Also, each value $V_{i,j}$ of the $cell_{i,j}$ represents a binary decision variable, defined as follows: $V_{i,j} = 1$, if the cell contains a sensor node, whereas $V_{i,j} = 0$ otherwise.

Fig. 5 and Fig. 6 illustrate the encoding of the deployment solution for both of mono-objective and multi-objective algorithms.

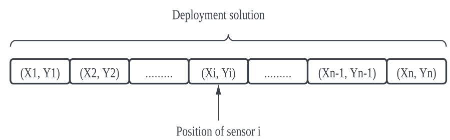


FIGURE 5. The vector representation of the decision variables.

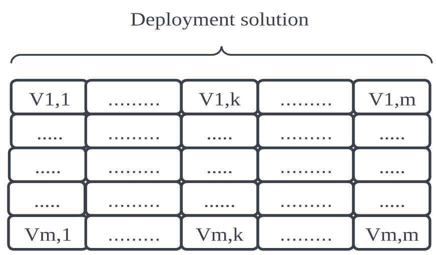


FIGURE 6. The grid representation of the decision variables.

B. WSN OBJECTIVE FUNCTIONS MODELING

1) COVERAGE

The primary function of WSNs is to monitor the environment and sense specific events. Therefore, coverage is identified as a salient performance metric that must be prioritized in WSNs deployment design. It is defined as the ratio of the supervised area by the sensor network to the entire area of interest [15]. Mainly, coverage is classified into three types: point (target) coverage, barrier coverage, and area coverage. In point coverage, sensor nodes are deployed to monitor a set of target points that could be static [16], or mobile [17]. This type of coverage is widely used in military applications in which a set of locations must be controlled. Another variant of the target coverage problem is Q-coverage [18]. This latter adds QoS requirements, such as each target point should be covered by a predefined number of sensors. Moreover, a periodic target coverage called sweep coverage was tackled in the literature [19], [20], [21]; it seeks to deploy fewer mobile sensors to monitor a set of target points. The real challenge with this coverage type is scheduling a small number of

TABLE 2. Comparison between the related surveys.

Related surveys	Year	Objective functions modeling						AI based Techniques for WSNs deployment			Simulation tests
		Detailed sensing models	Coverage	Network Lifetime	WSN Connectivity	Energy consumption	Cost	Based on metaheuristics	Based on hybrid metaheuristics	Based on ML and FL	
[11]	2012	x	✓	x	✓	✓	x	x	x	x	x
[12]	2013	x	✓	x	x	x	x	✓	x	x	x
[8]	2015	x	✓	x	x	✓	x	✓	✓	x	x
[4]	2016	x	x	x	x	x	x	✓	x	x	x
[6]	2016	x	✓	✓	✓	✓	✓	✓	x	x	x
[14]	2016	x	✓	x	x	x	x	✓	x	x	x
[7]	2018	x	✓	x	x	x	✓	x	x	x	x
[10]	2021	x	✓	x	x	x	x	✓	x		✓
[5]	2021	x	x	x	✓	✓	✓	✓	x	x	x
Our survey	2022	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

mobile sensors to periodically supervise target points while consuming as little energy as possible.

Barrier coverage: It is used to protect the borders of critical regions or infrastructures such as territory frontiers from intruders that try to penetrate them [22]. A strong barrier coverage is provided by deploying sensor nodes in irregular belt shape to form a barrier with no gaps so that intruders can not traverse the region whatever the crossing path they use [22].

Area coverage: The goal of area coverage is to monitor a target region so that every location is within the sensing range of one or more sensor nodes [23]. Classical area coverage methods often assume a 1-coverage, meaning that every point in the region of interest must be covered by at least one sensor node. Yet, critical applications such as gas leakage explosions require higher accuracy. For that, researchers use the k-coverage technique [24] to ensure that every location in the target area is within the sensing ranges of k sensor nodes. The coverage function depends mainly on the sensing model and the environment modeling:

- Impact of the sensing model: According to [4], the sensing model is the mathematical formula used to estimate the probability that a target point is within the detection zone of a sensor node. Thus, the coverage function applies it to appraise the detection zone for each sensor in the network. Consequently, any inaccuracy of the sensing model will result in erroneous estimation of the full network coverage since it could lead to a significant disparity between real and predicted sensed data and skew any information derived from this data. This will be a serious network performance issue, especially for mission-critical applications requiring high QoS.
- Impact of the environment modeling: Accurate target area modeling enables measuring the detection zone of each sensor node in relation to the surrounding obstacles, thereby improving coverage estimation of the coverage function. A basic environment modeling, on the other hand, imposes strong assumptions that may not be met in reality, and so the coverage estimation may not reflect the true coverage of the sensor network.

Several sensing models have been proposed in the literature to estimate the detection zone of sensor nodes [25], [26], [27], [28]. These models are categorized into omnidirectional and directional sensing models, depending on the direction of the sensing range [3]. Most of these models do not consider the environmental impact (shadowing and signal attenuation) and sensor characteristics simultaneously. This issue affects the computation of sensors locations since the models, in most cases, do not reflect real-world scenarios. In what follows, we will define the most frequently used sensing models, which we have divided into two categories: deterministic models and probabilistic models. We compare between this models in Table 3.

2) DETERMINISTIC SENSING MODEL

Also called the Boolean model or the Binary model, is the most commonly used model in the literature because of its simplicity. This model assumes that the detection zone of a sensor node is a uniform disk of radius R_s (R_s is the sensing range of the sensor). That is to say, any event that occurs within the disk will be captured by the sensors; otherwise, it can not be detected. This model considers only the Euclidean distance between the sensor node and events or target points and does not consider other external factors such as obstacles or signal strength. The probability detection of this model is shown in Eq.1:

$$P_{det}(P, s) = \begin{cases} 1 & \text{if } d(P, s) \leq R_s \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where $d(P, s)$ refers to the Euclidean distance between the point P and sensor s, R_s is the sensing range of sensor s, and $P_{det}(P, s)$ is the probability that the target point or the event P is within the sensing range of sensor s.

3) PROBABILISTIC SENSING MODEL

It assumes that a probabilistic distribution models the sensing range of a sensor and any event that occurs within the sensing zone will be detected with a certain probability.

This latter depends on the sensing model applied. In what follows, we will present the principal probabilistic models found in the literature.

- Sigmoid model: This model was used in [29] to conceive membership functions for sensing range (distance) and sensing angles, and then the final detection probability is the multiplication of membership functions. The probabilistic membership function of distance is given as follows:

$$P_{det}(d) = 1 - \frac{1}{1 + e^{\beta(d-R_s)-t_d}} \tag{2}$$

where d is the Euclidean distance between the event or the target point and the sensor, β and t_s are two adjustable parameters according to the characteristics of the sensor, and R_s is the sensor sensing range.

- Attenuated disk model: This model assumes that the sensing ability of a sensor decreases when the distance separating it from the event (or target point) gets longer [30]

$$P_{det}(d) = \frac{\lambda}{d^{\alpha}} \tag{3}$$

where λ is a constant, d represents the Euclidean distance between the sensor and the target point, and α is the path attenuation exponent reliant on the environment.

- Probabilistic model with noise: this sensing model is similar to the attenuated disk model, yet it considers the impact of the environment on the sensing ability of the sensor. For this, it includes a noise energy η that follows the Gaussian distribution.

$$P_{det}(d) = \frac{\lambda}{d^{\alpha}} + \eta \tag{4}$$

- Exponential model: this model measures the sensing attenuation based on the distance d between the sensor and the target point. [31].

$$P_{det}(d) = e^{-\alpha d^{\beta}} \tag{5}$$

α and β represent the degree of sensing attenuation.

- Shadow fading model: in this model, the sensing ability of a sensor node is not regular in all directions because of the existence of obstacles [32].

$$P_{det}(d) = Q(\frac{10\eta log_{10}(d/r_s)}{\sigma}) \tag{6}$$

where,

$$Q(x) \triangleq \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-y^2/2} dy$$

η is the path loss exponent, σ is the shadowing parameter, and d and r_s are respectively the distance between the sensor and the event and the sensing radius.

- Elfes sensing model: This model takes into consideration both the distances between the sensor and the event and the sensor's physical properties. It is defined as follows [32]:

$$P_{det}(d) = \begin{cases} 1, & d \leq R_1 \\ e^{\lambda(d-R_1)^{\gamma}}, & R_1 < d < R_{max} \\ 0, & d \geq R_{max} \end{cases} \tag{7}$$

where R_{max} is the maximum sensing radius of the sensor, R_1 represents the certainty zone of the sensor detection, λ and γ are fixed based on the sensor's physical characteristics.

- hybrid model: this model was proposed in [33], it combines the Elfes sensing model and Shadow fading model for the purpose of considering both the sensor characteristics and the environmental factors simultaneously.

$$P_{det}(d) = \begin{cases} Q(\frac{10nlog_{10}(d/r_s)}{\sigma}), & 0 \leq d \leq R_1 \\ \min(e^{\lambda(d-R_1)^{\gamma}}, \\ \frac{10nlog_{10}(d/r_s)}{\sigma}), & R_1 < d < R_{max} \\ 0, & d \geq R_{max} \end{cases} \tag{8}$$

Here d represents the Euclidean distance between the target and the sensor node, and the remaining parameters are the same as in the Elfes and Shadow fading sensing models.

4) COVERAGE MODEL

In this section, we will focus on the area coverage models. There are primarily two approaches for assessing the network's overall coverage: the grid (matrix) model and the sensing zones aggregation model. The matrix technique depicts the region of interest as a grid, with sensors positioned in the center of cells. Each cell is meant to be covered if its center is within the detection zone of a sensor node. In this case, the coverage model will be the ratio of covered cells to the total number of cells:

$$Coverage(Z, S) = \frac{\sum_{i=1}^{H \times W} Cov(cell_i, S)}{H \times W} \tag{9}$$

where H and W represent the length and the width of the matrix (area of interest) respectively, S is the set of sensor nodes to be deployed, Z is the whole zone (all the cells in the matrix), and $Cov(cell_i, S)$ is the probability that the $cell_i$ is covered by the set S and it has two values depending on the sensing model used to assess the probability that a sensor s covers $cell_i$, in what follows, we refer to this probability as $cov(cell_i, s)$:

- Deterministic sensing model:

$$Cov(cell_i, S) = \begin{cases} 1 \ \exists s \in S \text{ where } cov(cell_i, s) = 1 \\ 0 \text{ otherwise.} \end{cases} \tag{10}$$

- Probabilistic sensing model:

$$Cov(cell_i, S) = 1 - \Pi_{j=1}^{\|S\|}(1 - cov(cell_i, s_j)) \tag{11}$$

If the cells of the metrics have different degrees of importance, then each coverage probability of a given cell will be multiplied by a preset weight. The sensing zones aggregation model considers the union of all sensors detection zones. This method is more accurate than the grid method since it computes the covered regions geometrically conversely to the first method, which assumes that a cell is covered if its centre

TABLE 3. Sensing models in the literature.

Sensing model	Parameters	Considered sensor physical features	Considered elements	limits
Deterministic model	None	Sensing range	distance d	unrealistic model Does not consider sensor characteristics Does not consider the impact of the environment(shadowing and signal attenuation) Overestimation of the sensing ability
Sigmoid model	β, t	Sensing range directional angles other hardware characteristics	Distance d Reference angles	Does not consider the impact of the environment Assumes a uniform sensing ability in all directions Overestimation of the sensing ability
Attenuated disk model	λ, α	None	Distance d	Assumes idealistic environment without obstacles Does not consider sensor characteristics Overestimation of the sensing ability
Exponential model	α, β	None	Distance	Assumes idealistic environment Does not consider sensor characteristics Assumes a uniform sensing ability in all directions Overestimation of the sensing ability
Probabilistic model with noise	λ, α	None	Distance	Does not consider sensor characteristics
Shadow fading model	$\eta\ \sigma$	None	Distance d	Does not consider sensor characteristics The path loss is the same in all directions
Elfes model	λ, R_{max}, R_1	Sensing range other hardware characteristics	Distance d	Does not consider the impact of the environment (shadowing and signal attenuation)

is covered.

$$Cov(Z, S) = \frac{\| \cup_{j=1}^{\|S\|} detZ(s_j)\|}{Z} \tag{12}$$

Here the $detZ$ function computes the area of a sensor detection zone.

5) NETWORK LIFETIME

WSN lifetime represents the duration in which the network can fulfill its mission properly. It has several definitions that coexist in the literature. It can be described as the time duration of the sensor network until the first sensor node runs out of energy [6]. Its mathematical model can be formulated as follows:

$$Lifetime_{network} = \min(lifetime(node_i)_{i=1,\ldots,N}) \tag{13}$$

The lifetime is also defined as the ratio of the time until one of the sensor nodes runs out of energy, i.e. the time until the first sensor node failure to the maximum lifetime of the sensor network. It is also modeled as follow:

$$Lifetime_{network} = \frac{\min\{T_{failure_i}\}_{i=1,\ldots,N}}{T_{max}} \tag{14}$$

where $\min(T_{failure_i})_{i=1,..,N}$ represents the maximum number of sensing cycles before the first sensor node runs out of energy and T_{max} is the maximum sensing cycles of the network.

6) ENERGY CONSUMPTION

Energy consumption is a crucial concern in WSNs since sensor nodes are energy-constrained devices. They consume energy while sensing the environment, processing, transmitting and receiving data. Moreover, in some situations, the communication subsystem could engender other sources of wasted energy [34], such as in the case of packets collisions, overhearing, control packet overhead, idle listening and interference [34]. Therefore, various solutions for energy conservation were proposed in the literature to extend the WSN lifespan [35], [36], [37], [38], [39]. Each solution deals with a specific aspect such as data reduction-based techniques, duty cycling technique, and energy-efficient routing. We distinguish two major approaches in data reduction-based techniques: data prediction and data compression. The data prediction approach attempts to describe the sensed data by establishing a model. This latter will be exploited to generate data instead of using real gathered ones, and it can be built using stochastic approaches, time series forecasting, and algorithmic approaches [40]. On the contrary, the data compression approach uses the real sensed data while decreasing the number of bits that must be transferred; hence, the energy used for communication will be preserved. Duty cycling techniques schedule the activity of sensor nodes [34] so that sensors are switched off when they do not impact the network's functionality. Another technique used for energy conservation is the design of energy-efficient routing protocols that intend to find the most effective path for end-to-end packets transmission while considering the residual energy for each sensor node. The mathematical model [41] used to describe the energy E_{cons} consumed by a set of sensor nodes in a given path is:

$$E_{cons} = \sum_{k=1}^{N} (t_k^{access} + t_k^{process}) * E_k^{operate} + E_k^{trans} * t^{msg} \tag{15}$$

where t_k^{access} and $t_k^{process}$ correspond to the time needed by node k to acquire and process data respectively, N represents the number of nodes in the path, t^{msg} is the message transmission time duration, $E_k^{operate}$ and E_k^{trans} are the operational power and transmission power of node.

7) DEPLOYMENT COST

The deployment cost is an essential factor in designing the WSN deployment scheme. It can be defined as the total cost of purchasing and positioning all sensor nodes in the target area. Its mathematical model is represented as follows:

$$Cost_{wsn} = cost_{sensor} * |SN| + cost_{sink} * |SIN| \quad (16)$$

where SN and SIN represent the number of sensor nodes and sink nodes respectively, and $cost_{sensor}$ and $cost_{sink}$ represent the purchasing and positioning of s,ensor nodes and sink nodes, respectively.

C. FEASIBILITY CONSTRAINTS

1) NETWORK CONNECTIVITY

WSNs are deployed to sense and collect measurements from the surrounding environment and send them back to the base station for further processing. In order to achieve this objective, it is essential to ensure full connectivity between all sensor nodes to avoid losing information. Two sensor nodes are connected if they can exchange data in both directions. Furthermore, a k-connectivity with ($k >= 1$) means that there exist at least k distinct communication paths between each pair of sensors. Several mathematical models have been proposed in the literature to quantify the connectivity between sensor nodes. The commonly used model is the binary communication model (see Eq. 17), which considers that a sensor node can send data to another sensor if the Euclidean distance between the two nodes is less or equal to the minimum of their communication ranges.

$$P_{con}(s_i, s_j) = \begin{cases} 1 & \text{if } d(s_i, s_j) \leq R_c \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

Other reliable models based on signal propagation have been proposed to assess the connectivity between nodes mathematically. Radio propagation models aim to estimate the behavior of signal spreading in different environments [42]. Indeed, a signal may encounter several types of obstacles according to the environment it crosses, and therefore, it could be scattered, refracted, reflected, and diffracted [43]. According to [44], signal propagation modeling methods are mainly categorized into four types:

- Deterministic models: There are very high accuracy models that simulate the signal propagation in a specific location since they apply physical laws on 3D data describing the environment. These models are costly in terms of computing resources and time, and the commonly used models of this category are Ray-Tracing and Ray Launching.
- Stochastic models: These models use random variables to describe the randomness of the radio channels [44]. Hence, they are highly employed in large scale fading and small scale fading modeling. The Rayleigh fading model and Rice fading model are the most known ones in this category.
- Empirical models: These are the most used models in the field of network design because of their simplicity and low computational time. An empirical model is based on a huge collection of measurements related to a specific situation (system parameters, environment and type of communication system) [45] to predict the path loss of the signal.
- Semi-deterministic models: These are a combination of deterministic models and stochastic or empirical models [44]; thus, they are assumed to be more precise than stochastic or empirical models and consume lower computational resources than a deterministic model; an example of this category is the Dominant path model.

IV. TARGET AREA MODELING

The deployment scheme strongly depends on the target area characteristics, namely its form, dimensions, type (indoor/outdoor), and obstacles. The form and dimensions are used to outline the borders and define the potential deployment locations within the area of interest. The obstacles allow for a more refined selection of the possible deployment points by excluding the locations where they are present. Therefore, having a reliable data source that covers all the features of obstacles, precisely their thickness, materials, widths, and heights, is necessary for assessing their impact on the sensing and communication zones of the sensor nodes and hence obtaining a realistic deployment result. Indeed, the phase of target area modeling is often neglected in the proposed solutions as most of them suppose a 2D free obstacles layout. When considering obstacles, these are illustrated as dispersed regular or irregular polygons and could be homogeneous or heterogeneous:

- Homogeneous obstacles: Described as opaque objects [46], [47], [48], [49], [50], [51] that completely hinder the signal transmission. Hence, the deployment solution avoids positioning sensors in the vicinity of obstacles to maximize coverage even further.
- Heterogeneous obstacles: They attempt to simulate a real-life environment by incorporating many sorts of obstacles, each with a different attenuation value that defines how the signal intensity is affected [52], [53], [54].

A. INDOOR ENVIRONMENT VS OUTDOOR ENVIRONMENT

Most of the reported WSNs deployment solutions do not specify the type of environment, whether indoor or outdoor and assume that the same deployment scheme can be adopted in both. However, the two environments have different properties, as explained below:

- Types of obstacles: One of the main differences that should be highlighted in the solution design of an indoor and an outdoor deployment is the types of obstacles existing in both environments. An indoor environment refers to all types of buildings (houses, hospitals, malls, schools, etc.), and it is primarily characterized by the construction materials constituting walls and ceilings, woods, glass, etc. In contrast, an outdoor environment could refer to a city,

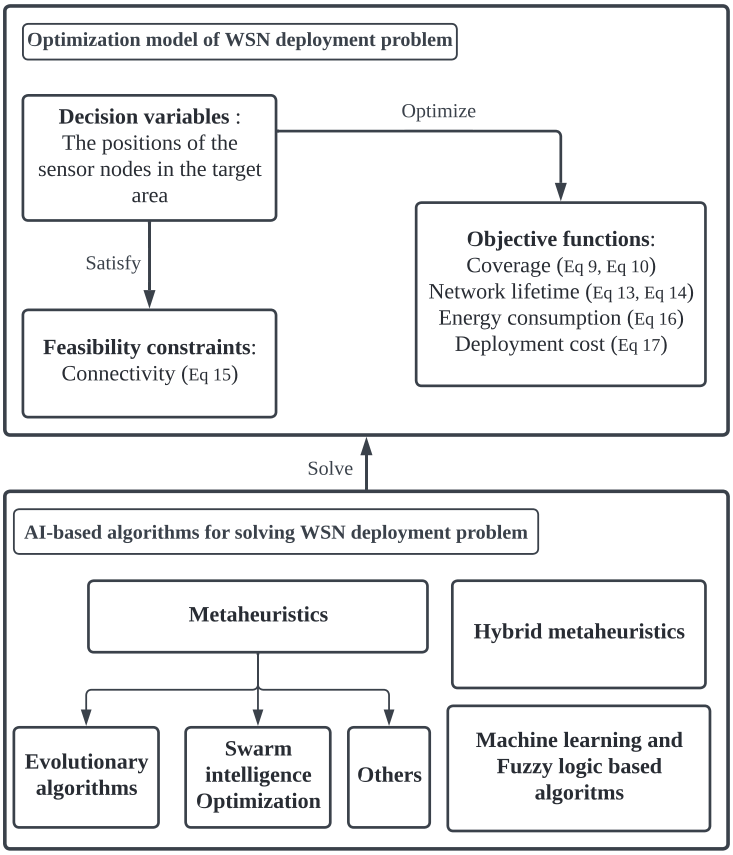


FIGURE 7. A general reference optimization model for WSN deployment problem.

TABLE 4. Data sources used to model the target area in WSNs deployment problem.

Data source	Type of environment	Dimension of environment	Type of obstacles	Shape of obstacles	Position of obstacles
BIM (IFC file)	Indoor	3D	Included	Included	Included
Grid model of DEM	Outdoor	3D	Not included	Not included	Included
TIN model of DEM	Outdoor	3D	Not included	Not included	Included
Contour model of DEM	Outdoor	3D	Not included	Not included	Included
Satellite image	Outdoor	2D	Included	Included	Included

a forest, a mountain or even an ocean, and it may contain different types of obstacles: trees, buildings, water, rocks, etc. Therefore, the WSN placement should be adjusted according to the target area obstacles in order to have better performance.

- Data source describing the target area: As mentioned before, having a data source describing the area of interest is very important to achieve a realistic deployment. A complete data source that can describe a building accurately and provide the necessary information (separators, plans, windows, materials, etc.) needed to model the target area is the Building Information Modeling (BIM) tool [55]. For outdoor environment, researchers use mainly the Digital Elevation Model (DEM) [46], [56], [57] and raster and vector modelings [58] which is a 3D representation of the terrain topology. This data source is still inaccurate since it does not contain all the information related to the target area, namely the terrain type. Table 4 summarizes the commonly used target area modeling used in literature.

B. 2D VS 3D ENVIRONMENTS

The area dimension is another important criterion to be considered in the process of target area modeling. Indeed, most of the existing research works assume a 2D flat area divided according to a regular pattern as the grid representation [52], [54] or using a computational geometry approach such as Voronoi diagram and Delaunay triangulation [59], [60], [61]. Other approaches do not adopt any area division technique but define a set of distributed deployment points where sensors could be placed. Both representations cannot describe real-world scenarios since they do not lead to a realistic assessment of the coverage, connectivity, and deployment cost. Thus, a more complex 3D modeling is required to simulate the real behavior of WSN [62], mainly the propagation of its communication and sensing signals. Indeed, 3D area modeling allows us to consider the elevation of obstacles and terrain at each point in the target area and thus, their impacts on the line of sight between target points and sensors as well as the obstruction degree of the sensing and communication zones of the sensors. The 3D WSN deployment has been proven to be more challenging and necessitates more sensor nodes to reach the same coverage rate as a 2D WSN deployment [63]. The 3D grid division [64], [65], [66] and Digital elevation model (DEM) [46], [56], [67] are the two widely used target area modelings in 3D environment.

V. WSNs DEPLOYMENT USING ARTIFICIAL INTELLIGENCE

Artificial intelligence (AI) is the branch of computer science that focuses on finding solutions to complex real-world problems requiring human intelligence. In the scope of the WSNs deployment problem, metaheuristics are the most used AI techniques by the research community to compute the relevant deployment scheme. Very few initiatives are based on machine learning namely the Q-learning to heal coverage holes by redeploying sensors. In this section, we survey WSNs deployment solutions based on various AI techniques namely evolutionary algorithms, swarm intelligence-based algorithms, hybrid algorithms, machine learning algorithms, and fuzzy logic-based algorithms.

A. METAHEURISTICS

Exact methods are algorithms that guarantee convergence to an optimal solution. The commonly used exact methods are Linear Programming [68], Dynamic programming [69], and the family of Branch and X [70]. They are more adapted to optimization problems with small instances [71], however, their convergence times are too excessive. To cope with this problem, academics have suggested metaheuristic algorithms that can find a near-optimal solution in a reasonable amount of time [8]. Metaheuristics are defined as algorithmic structures that can solve a wide range of optimization problems with only a few adjustments to match the given problem [72]. Several metaheuristic algorithms have been proposed in recent years; a lot of them are metaphor-based algorithms, such as biology-based, physics-based, chemistry-based, etc. [72]. This leads to a difficult comprehension of algorithms since the terminology used for each one originates from its domain of inspiration rather than the domain of optimization [73], [74]. A number of WSN-related challenges, including the deployment of sensors, have been addressed using metaheuristic algorithms. In what follows, we will focus on two types of commonly used metaheuristics in solving sensors deployment: evolutionary algorithms and swarm intelligence optimization algorithms. Table 5 exposes these metaheuristics-based approaches according to several comparison criteria.

1) EVOLUTIONARY ALGORITHMS

Evolutionary intelligence is a branch of bio-inspired algorithms that relies on population concept and biological heredity [75] which means transferring features from parents' generation to children's generation. The population concept allows for the simultaneous search for the optimal solution in more than one direction using individuals. An individual represents the encoding of a solution for a given optimization problem. The individuals of iteration i are called parents, and the individuals of iteration $i+1$ are called children. Parents share the search information with children through evolutionary reproduction operators. Each individual has a score that denotes how well it solves a particular problem. Individuals with a high score will replace parents in the next population and cooperate in producing new individuals with evolved performances. There have been numerous applications of evolutionary algorithms in solving real-world problems [76], including the problem of WSN deployment. [46], [52], [53], [54], [77], [78], [79], [80], [81], [82], [83], [84], [85], [86], [87], [88], [89]. In what follows, we will emphasize the most prominent evolutionary algorithms with their related WSNs deployment approaches.

Algorithm 1 Pseudo Code of the Genetic Algorithm

```
Initialize population
Compute fitness for each chromosome
while Termination condition is not satisfied do
    Selection of parents to generate offspring
    Recombine parents (crossover)
    Mutate children
    Compute fitness for new individuals and update population
end while
```

a: GENETIC ALGORITHM (GA)

It is one of the evolutionary algorithms that have been widely applied in diverse problem domains. Its pseudo-code is depicted in algorithm 1. Numerous research works have used GA to solve the WSNs deployment problem. Each work tried to propose sophisticated operators and individual coding. In [52], authors developed an optimizer based on a constrained multi-objective genetic algorithm. They considered the weighted sum method to combine two objectives of maximizing coverage and minimizing cost under the limited budget, the coverage, and connectivity degrees constraints. Additionally, the authors proposed a new individual encoding to model the position of heterogeneous sensors within the area of interest and used the elitist selection and new mutation operator that allows removing, adding, and moving the sensor nodes. The same authors applied the GA and the weighted sum technique in [53] to tackle the problem of WSNs deployment in an indoor environment. They combined both coverage and connectivity according to predefined network topology. The deployment space is depicted as a matrix, and each chromosome's gene defines whether the associated cell of the matrix is occupied by a sensor node, a router, or empty. Authors in [77] conceived a multi-objective genetic algorithm to optimize the placement of sensor nodes. The proposed algorithm aims to widen the coverage range and reduce energy consumption by decreasing the number of active nodes. The authors used the binary coding that considers the active state of sensor nodes, i.e. a gene at position i is set to 0 if the sensor i is in sleeping mode; otherwise, it is set to 1, and the corresponding sensor is on working mode. Furthermore, the authors presented an enhanced fitness function that analyzes the number of potential additional covered target locations before deciding whether or not to activate a sensor node, and they compared the single-point and multi-point crossover operators. The simulation results show that the multi-point cross-over allows for achieving better results in terms of convergence rate and optimization objectives. Another work in [78] has applied the GA to tackle the problem of the heterogeneous sensor nodes deployment within a 2D area with obstacles. The proposed model focuses on area coverage optimization under the connectivity constraint. Moreover, it includes an improved population initialization based on a modified virtual force algorithm and a fitness function that considers the coverage overlap between each pair of sensors and the overlaps between sensors and obstacles. In [90], the authors aimed to prolong the network lifetime using caching mechanism as a means to reduce data transmission and network latency. Thus, they applied the GA to compute the optimal positions of cache nodes that allow covering a maximum number of requesting nodes. Simulation results showed that the proposed solution achieved better performance in terms of average latency and the total number of messages compared with other existing methods. Kosar and Ersoy have tackled in [91] the problem of sink node placement in a 3D environment for border surveillance. The basic scheme of their approach is based on a Discrete WSN simulator and the GA-based optimizer. The former component is used to compute the network lifetime which represents the fitness function. It simulates the sensing and communication functions of the WSN in a given terrain elevation map while computing the network lifespan. In the optimizer component, an individual represents the location of the sink node. The results show that the proposed method achieves better performance in terms of lifespan gain compared to existing heuristics.

b: NON-DOMINATED SORTING GENETIC ALGORITHM (NSGA-II)

Another evolutionary algorithm that has been widely applied in solving WSNs deployment problem [46], [54], [80], [81], [82], [83] is NSGA-II [92]. This metaheuristic is a multi-objective optimization algorithm with two main aspects: fast non-dominating sorting and crowded distance. Authors in [46], conceived a two objectives NSGA-II based approach with a guided crossover and mutation operations for WSNs deployment problem in a 3D environment. In their work, the authors suggested an individual coding scheme that incorporates information on sensor nodes' locations and directions, and they computed coverage using a probabilistic sensing model presented in [29] with an improved visibility function based on the Bresenham line-of-sight algorithm. Benatia et al. [54], sought to find the near-optimal solution for WSNs deployment in smart buildings through two evolutionary algorithms: GA and NSAG-II. Therefore, the proposed approach deals with multiple objectives: deployment cost, coverage, connectivity, and over-coverage. According to the authors, the choice of the adequate metaheuristic algorithm depends on the user requirements; the NSGA-II is recommended when the approach is not a-priori.

Dahmane et al. [80], dealt with the problem of deploying temperature sensor nodes in smart buildings; their approach is based on NSGA-II with two objectives to optimize coverage and cost. In their solution, the authors included BIM database information to model obstacles of the building. Indeed, their coverage model depends mainly on the distance between the target point and the sensor node and the heat flow resistance of materials constituting obstacles. Khalesian and Delavar proposed in [81] a constrained Pareto-based multi-objective evolutionary approach that attempts to reach a trade-off between the network coverage and the energy consumption objectives while maintaining the sensors connectivity. They modeled the sensors network as a connected graph of k sensor nodes and $k-1$ edges referring to the communication links between nodes. Then, they conceived two crossover operators that allow for generating feasible solutions. The first operator combines two parent graphs to form a graph with $2k$ nodes and $2k-2$ edges; next, it randomly selects $k-1$ edges and removes them one by one under the constraint of network connectivity. The second offspring is generated by restarting the same process from the edges that have not been picked yet. The second crossover operator allows preserving the locations of sensors without transmitting the parents' communication links to offspring. Then, each sensor node in a parent must find its matching sensor node in another parent in order to form the edges of the offspring. Simulation results indicate that the first crossover operator produces better solutions than the second one. According to the authors, this could be due to the ability of the first operator to conserve the topological and geometric characteristics of parents.

In [93], the authors tackled the problem of relay nodes placement into an existing static WSN. They assumed that the target area is a 2D rectangle without obstacles, and the sensor nodes are battery-powered devices. The solution is modeled by a vector containing the Cartesian coordinates of the relay nodes, and the fitness function comprises three objectives: energy consumption, coverage area, and network lifetimes. The classical versions of NSGA-II and SPEA2 evolutionary algorithms and MO-VNS trajectory algorithm were implemented and compared in terms of hypervolume and set coverage. Simulation tests showed that the NSGA-II is outperformed by the mentioned trajectory algorithm.

c: MULTI-OBJECTIVE EVOLUTIONARY ALGORITHM BASED ON DECOMPOSITION (MOEA/D)

It is a commonly used evolutionary multi-objective optimization algorithm for solving WSNs deployment problem. This method divides the problem into several sub-problems and optimizes them all at the same time [94]. Wei proposed in [95] a multi-objective approach based on MOEA/D, which considers the average energy consumption, the average sensitivity area, and the network reliability objectives. The method uses a uniform design in generating weight vectors so as to obtain a uniform distribution of optimization sub-problems. In the work of Sengupta et al. [96], authors adapted the MOEA/D-DE algorithm, which is an extension of MOEA/D with differential evolution-based reproduction operations. The adaptation was made through the integration of fuzzy dominance to compare solutions. The proposed multi-objective solution tries to calculate the positions of the sensors within the zone of interest to optimize coverage and network lifetime and minimize energy consumption and sensors number under the connectivity constraint. Konstantinidis et al. in [97] attempted to enhance the coverage and the lifetime of the WSN. To do so, they proposed a MOEA/D-based solution to which they incorporated problem-specific knowledge. Indeed, in their approach, each sub-problem has an objective preference to which the evolutionary operators are adapted dynamically. In their work [98], authors improved their solution by adding the k-connectivity constraint. For that, they integrated a problem-specific heuristic to generate a feasible initial population and a repair heuristic for connectivity constraint handling.

Authors of [99] adopted the MOEA/D among other algorithms to address the problem of relay nodes placement in a previously deployed WSN. They aimed at minimizing the energy consumption and maximizing the network coverage and network lifetime. To this end, they defined the chromosome as a 1D array containing the Cartesian coordinates of each sensor node within a 2D rectangular area, and they applied a decomposition approach that combines the Normal Boundary Intersection (NBI) method and the Tchebycheff method in order to manage the scales of the considered objective functions. In [100], the authors explored the potential for bio-inspired algorithms to solve multi-objective sets covering problems in WSNs. In this study, two major contributions are made. First, the authors developed a multi-objective set covers (MOSC) formulation that addresses three issues: network lifetime, target coverage, and network connectivity. As a second step, they developed and elaborated four well-known multi-objective optimization algorithms from the evolutionary and swarm intelligence communities. Indeed, MOEA/D, NSGA-II, MOPSO, and NSPSO frameworks are adapted for solving formulated problems by adjusting their characteristic components. The simulation results clearly demonstrated the merits of the proposed self-adaptive heuristic operator. Moreover, according to the authors, deterministic mathematical methods and bio-inspired meta-heuristics for solving WSN deployment are almost completely disjointed. Consequently, they recommend coupling their desirable features in one hybrid and heuristic algorithm to bridge the gap between these two methods.

d: CUCKOO SEARCH ALGORITHM (CS)

This evolutionary algorithm was proposed by Yang and Deb in 2009 [101]. It mimics the obligate brood parasitic behavior of some cuckoos. The solution of the CS algorithm is encoded in the cuckoo's eggs, which are thrown in host nests of other bird species. If the host bird recognizes the alien cuckoo egg, it will either toss it or quit its own nest in favor of a newly established nest. According to [101], the CS algorithm is based on three idealized rules: 1) Each cuckoo lays one egg at a time and deposits it in a nest that is picked at random; 2) The best nests with the highest quality eggs will be passed down to future generations; 3) The number of possible host nests is predetermined, and the host bird discovers the cuckoo's egg with a probability $p_a \in [0, 1]$. The CS algorithm has been used in several works [84], [87], [88]. In [84], the authors proposed a two-stage mobile sensor deployment approach based on the CS algorithm. This approach aims to ensure maximum coverage with a minimum number of mobile sensor nodes and average mobile distance. Its first stage tries to maximize coverage by positioning the sensors in a target area digitized into a 2D grid without obstacles. For that, the authors used the CS algorithm with a Levy flight search mechanism to randomly select nests. Then the deployment scheme is optimized in the second stage by reducing the number of sensors and the moving distance. Another solution based on the CS algorithm was suggested in [87], the authors considered the coverage maximization of heterogeneous sensors network. Each cuckoo's egg is depicted as an array of the sensors' coordinates in a 2D area grouped by sensor types. The authors also improved the CS algorithm by adjusting the levy flight parameters; this allows creating new solutions with much smaller step lengths so that the new individuals will not be pulled away from the best solutions. According to the experimental results, the improved CS algorithm provides a good solution in a short time compared to other metaheuristics. In [88], authors developed an improved CS algorithm to tackle the k-target coverage problem of randomly deployed sensor nodes with adjustable sensing radius. The problem was formulated as a non-linear integer programming problem to optimize both coverage and network lifetime. For that, the authors assumed a corresponding energy consumption for each sensing range and divided the sensors into a set of non-disjoints covers activated alternatively to expand the network lifespan.

2) SWARM INTELLIGENCE OPTIMIZATION ALGORITHMS

Swarm intelligence (SI) is a sub-field of artificial intelligence that has been widely applied to solve nonlinear problems related to several real-world domains [102]. SI is based on the collective behavior of agents. Each agent represents a solution

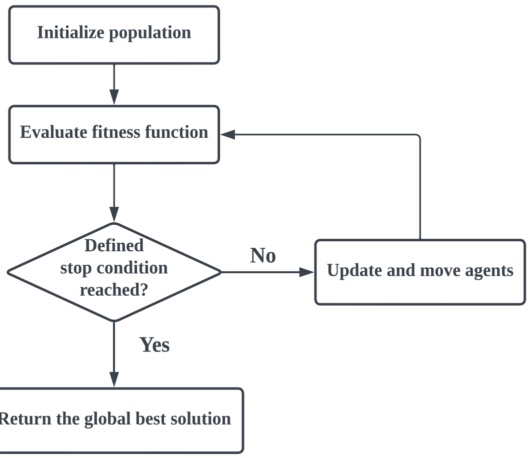


FIGURE 8. Swarm intelligence framework [104].

Algorithm 2 Pseudo Code of PSO Algorithm

Initialize particles
Compute fitness for each particle
while termination condition is not satisfied **do**
  for each particle $Particle_i$ in the swarm **do**
    Compute Fitness($Particle_i$)
    Update Personal Best Fitness($Particle_i$)
    Update Personal Best Position($Particle_i$)
  end for
  Update Global Best Solution()
  for each particle $Particle_i$ in the swarm **do**
    UpdateVelocity($Particle_i$)
    UpdatePosition($Particle_i$)
    Update Personal Best Position($Particle_i$)
  end for
end while

to the problem and it adapts its behavior autonomously. Moreover, a SI system has two essential features, self-organization and labor division [103]. Self-organization characterizes a swarm's ability to grow over iterations via the interaction of its components without the need for external intervention, whereas labor division refers to the simultaneous execution of several tasks by the swarm's agents. There are common phases between EA and SI systems namely, population initialization, defining stop condition, and evaluating fitness function [104]. Yet, each SI algorithm has its own strategy for updating the movement of its agents. Fig. 8 illustrates the general framework of a swarm intelligence system.

a: PARTICLE SWARM OPTIMIZATION ALGORITHM (PSO)

PSO was proposed by Kennedy and Eberhart in 1995 [105]. It is a nature-inspired algorithm based on swarm intelligence [6]. This algorithm mimics the social behavior of a flock of birds in searching for the optimal solution in search space. A particle in PSO encodes a potential solution, and it has two characteristics: velocity and position, which are updated at each iteration according to Eq. 18 and Eq. 19 respectively. The position of a particle refers to its current solution; therefore, the best position is retained as a self-experience. Velocity allows the particle to combine its experience with the swarm experience to calculate its new position. The pseudo-code of PSO is depicted in Algorithm 2.

$$\begin{aligned} v_i(t+1) &= w \times v_i(t) + c_1 \times rand_1 \times (P_{pbest_i} - P_i(t)) \\ &\quad + c_2 \times rand_2 \times (P_{gbest} - P_i(t)) \qquad (18) \\ P_i(t+1) &= P_i(t) + v_i(t+1). \qquad (19) \end{aligned}$$

where w, c_1 and c_2 represent inertia weight, cognitive acceleration and social acceleration respectively. $rand_1$ and $rand_2$ are two random numbers uniformly distributed in [0,1]. P_{pbest_i} denotes the best position reached by the particle, $P_i(t)$ refers to the current position of the particle and P_{gbest} is the global best position of the swarm. Several works have applied PSO algorithm and its variants to find the optimal placements of WSNs [47], [59], [106], [107], [108], [109], [110], [111], [112], [113], [114], [115]

Du [106], considered the WSN deployment problem in 3D terrain. To solve this problem, the author proposed a combination solution of the distributed particle swarm optimization and the 3D virtual force algorithm to maximize the coverage. The 3D virtual force algorithm helps avoid obstacles in the zone of interest and maintain network connectivity. Furthermore, the author has developed a heuristic to manage the communication limits of the sensor nodes by clustering the network so that each sensor node can communicate with the base station. Authors in [107] proposed a PSO-based solution for deploying a WSN used by environmental and health applications. They intended to find the optimal locations of sensors to optimize network coverage and lifetime while considering the connectivity constraint. The Minimum Spanning Tree (MST) routing protocol was applied to reduce the network's energy consumption, thereby extending its lifespan. Qi et al. in [108] studied the WSN redeployment problem to increase the network coverage and reduce the moving distance of mobile sensors. The proposed approach is based on the PSO algorithm and adapts a new nonlinear decreasing inertia weight as an improvement to avoid falling in local optima. Next, the deployment scheme is adjusted using the virtual force algorithm. Li et al. in [109] adapted the discrete binary particle swarm optimization for WSN deployment to optimize three objectives: network coverage, dormancy rate for boosting network lifetime, and coverage uniformity. The authors adjusted the particle velocity expression introducing a dynamic regulation of inertia weight, cognitive, and social factors. Also, they appended an escape operator that introduced a random position in the search space to avoid the local optima. Simulation experiments show that the proposed algorithm outperforms other solutions reported in the literature regarding the number of active nodes, coverage uniformity, and network energy consumption when the area coverage rate is more than 90%. Yarinezhad et al. [110], conceived a solution for WSNs deployment to solve the target coverage problem while considering the network lifetime. They used two versions of PSO: the cooperative PSO, which fits mainly the large-scale problems, and the cooperative PSO using fuzzy logic to adjust the acceleration factors dynamically. The simulation results demonstrate that the two suggested algorithms outperform GA, PSO, and the artificial bee colony technique in terms of network lifetime. Authors in [111] proposed an energy-saving solution for maximizing coverage in WSNs deployment. The solution is based on the PSO algorithm combined with an intelligent heuristic called Quasi physical force to decrease the overlapped coverage. Furthermore, the authors applied a dynamic balancing strategy to reduce energy consumption by shortening the sensing radius of some sensors in the network. Ni et al. in [112] tackled the problem of dynamic deployment, which aims at adjusting mobile sensors placements while considering both coverage maximization and moving distance minimization. They presented a heterogeneous multi-swarm PSO algorithm where the measurement of the traveling distance of a particle is computed using a discrete PSO, and the population is divided into three sub-swarms. Each swarm has different evolutionary strategies: PSO with inertia weight, PSO with constriction factor, and dynamic probabilistic PSO. This allows for boosting the population diversity and balances the exploration and exploitation of the algorithm. Furthermore, the authors compared the performance of their solution to the performance of two other PSO algorithms(classical PSO and co-evolutionary PSO), and the simulation results demonstrate that the multi-swarm PSO provided superior solutions with a reduced moving distance and a maximum coverage rate than the two PSO methods. A variant of PSO called Social Class Multi-objective Particle Swarm Optimization (SC-MOPSO) was applied in [115] to deal with the problem of WSN deployment. It aims to minimize both the uncovered area and the deployment cost. The particles have variable lengths depending on the number of deployed sensors. SC-MOPSO splits the population into several classes where each class contains particles of the same length. The interaction between classes is done by moving the particles from the less performing class to the higher one. According to the results, SC-MOPSO performs significantly better than other benchmarks in terms of dominating solutions.

b: ANT COLONY OPTIMIZATION (ACO)

ACO is another nature-inspired algorithm based on swarm intelligence. It was first used by Dorigo et al. [116] as a metaheuristic to solve combinatorial optimization problems. In this algorithm, a group of agents called artificial ants colony cooperate together to stimulate the forging behavior of some ant species [117]. In the ACO algorithm, an optimization problem is designed as a connected graph $G(V, E)$ where the graph's vertices V and the graph's edges E are defined according to the problem variables. Each artificial ant generates a potential solution to the problem by incrementally constructing its tour in the graph. The ant chooses the next vertex during its tour construction process according to a probabilistic transition rule. This latter depends on three parameters: neighborhood definition, heuristic information, and pheromone trail. In nature, some ants species deposit pheromone trails on the path between the colony and food source. Therefore, this path will be more likely to be crossed by other ants in search of food to avoid traveling randomly. This concept is applied in the ACO algorithm, and the best tour found by the ants will have more pheromone trail. To better explain the principle of the ACO algorithm, let's assume that we have two paths, A and B, between the nest and the food, as the example given in [118]. Path A was used by $n_A(t)$ ants and path B was used by $n_B(t)$ ants at iteration t then the probability P_A that an ant chooses path A at iteration $t+1$ is given as follows:

$$P_A(t+1) = \frac{(c+n_A(t))^\alpha}{(c+n_A(t))^\alpha + (c+n_B(t))^\alpha} = 1 - P_B(t+1). \tag{20}$$

Various works have applied the ACO to compute the optimal positions of wireless sensors in a given zone [119], [120], [121], [122], [123], [124]. Authors in [119] considered the specified reliability metric value to calculate the locations of the sensors in the area of interest. Their main idea is to find the non-overlapping minimal connected covers (no redundant sensor nodes) at reduced deployment cost with respect to the reliability metric. The problem was solved with the ACO algorithm combined with a local search heuristic which is applied to each complete ant tour as a means to lessen its cost. Sun et al. [120], conceived an ACO based framework for WSN deployment. This latter guides the ACO through the use of a culture algorithm in order the reinforce the ACO stability and speed up its searching process. The final results guarantee the network connectivity with higher network coverage. Liu et al. [121], proposed a solution for WSN deployment with the objective of increasing coverage and decreasing the number of sensor nodes while maintaining the network connectivity. Their solution is based on the ACO algorithm with a greedy migration scheme to enhance ants tours. Further, it adjusts the sensing and communication ranges dynamically to reduce coverage holes and boost network lifetime. In [122], the authors addressed the problem of maximizing WSN coverage with lower cost in a 3D environment. They applied the ACO algorithm at the first step, with a modified heuristic value to guide the selection at each iteration towards a distant point in order to get a sparsely deployed network. The second step of the approach is the removal of redundant sensors which enables the network to ensure the connectivity constraint with a minimized deployment cost. The same authors proposed a WSN deployment approach in a 2D environment in [123]. As for the 3D environment, this approach applies at the first step the ordinary ACO with the greedy migration scheme presented in [121] to expand coverage. Whereas the second step removes redundant sensors from the solution computed in the first phase.

c: ANT LION OPTIMIZER (ALO)

Ant lion optimizer is a recent nature-inspired metaheuristic proposed by Mirjalili in 2015 [125]. It mimics the operation of catching prey (usually ants) by ant lions. This process comprises five steps: random walk of ants, creating traps of antlions, entrapment of ants in traps, hunting ants, and rebuilding traps. An ant lion starts by building a cone-shaped trap in the sand, entering its bottom, and waiting for ants to fall into the pit. Ants move in the search space using random walks and are caught by antlions in order to ameliorate their fitness [125]. The entrapment of ants in traps is mathematically modeled by adapting the ratio of ants' random walks. Once the prey reaches the base of the pit, it will be caught by the predator (antlion). ALO assumes that the catching process occurs when the fitness of a hunted ant is better than its corresponding antlion; in this case, an antlion updates its position to its prey position. The following algorithm summarizes the main steps of ALO. ALO algorithm has been adapted to the problem of WSNs deployment [126], [127], [128], [129]. In [129], the authors combined the ALO algorithm with the virtual force to optimize the coverage rate and the moving distance of mobile sensor nodes. The authors improved the ALO algorithm using three methods. First, they proposed an enhanced random walk to restrict the moving distance of a sensor. Next, they introduced a dynamic adjustment of the boundary shrinkage factor to improve convergence speed. Finally, they dynamically reduced the number of antlions participating in the selection to avoid falling into the local optima. In addition, the authors used the virtual force algorithm to direct the movement of the sensors according to three factors: boundary repulsion, neighbor node, and the target point gravity. Another work in [127] used the ALO algorithm to position mobile sensors node in a 2D target area to maximize the network coverage. The authors combined the ALO algorithm with the Tabu search to avoid visiting the same solutions in future generations. Therefore, the fitness of the elite stagnated in local optima is enhanced. In addition, they adopted a normalized random walk using the min-max normalization to avoid generating new solutions outside the boundary of the zone of interest.

Algorithm 3 Pseudo Code of Ant Lion Optimizer

```
Initialize the population of ants and antlions randomly
Find the elite (the best antlion)
while termination condition is not satisfied do
    for each ant do
        Select an antlion using Roulette wheel
        Decrease the radius of ants random walk to mimic the
        sliding process of an ant inside the trap
        Create a random walk and normalize it to keep it in
        the research space
    end for
end while
```

Algorithm 4 Pseudo Code of ABC Algorithm

```
Initialization phase
while Termination condition is not satisfied do
    Employed bees Phase
    Update optimal solution
    Update best food sources
    Onlooker bees phase
    Scout bees phase
    Update optimal solution
end while
```

d: ARTIFICIAL bEE COLONY OPTIMIZATION (ABC)

Artificial Bee colony optimization is a swarm-based metaheuristic proposed by Karaboga in 2005 [130] to solve both unconstrained and constrained optimization problems. ABC has three control parameters: population size, maximum cycle number, and limit. It simulates the smart foraging behavior of honey bees, and its population is composed of three types of bees: employed bees, onlookers, and scouts. An Employed bee is associated with a single food source representing a potential solution to the given optimization problem. Its nectar amount corresponds to the fitness of its solution. The onlookers collaborate with employed bees to find a food source, and scouts are responsible for finding new food sources by exploiting the research space. The algorithm 4 represents the main steps of the ABC algorithm. The main phases of this algorithm are summarized as follows:

1) Initialization phase: In this phase, the food sources are initialized by the scout bees according to Eq. 21 [131]. Each food source corresponds to the solution vector that must be optimized.
$$x_{mi} = l_i + rand(0, 1) \times (u_i - l_i) \tag{21}$$
where x_m indicates the mth solution vector and i is the ith position within it, u_i and l_i are respectively the upper and the lower bound of the parameter x_{mi}.
2) Employed Bees phase: Each employed bee goes to the food source in its memory and searches for new food sources in the neighborhood. If the neighbor source's nectar amount (fitness) is higher than the current source, then the employed bee updates the source position. Otherwise, it keeps the previous one in its memory. The neighbor food sources can be generated using the following formula [131]:
$$v_{mi} = x_{mi} + \varphi_{mi}(x_{mi} - x_{ki}) \tag{22}$$
where x_k is a randomly selected food source and φ_{mi} is a random number within the range [−a, a].
3) Onlooker Bees phase: After the previous phase's end, the employed bees return to the hive and share their food source information (position and nectar amount) with the onlookers. The latter will then choose their food sources

according to a selection technique such as the wheel selection method after calculating the probability value p_m for each food source as follows:

$$p_m = \frac{fit_m(\vec{x_m})}{\sum_{m=1}^{SN} fit_m(\vec{x_m})} \tag{23}$$

where SN is the number of food sources and $fit_m(\vec{x_m}$ is the fitness value of the food source $\vec{x_m}$.

4) Scout Bees Phase: The employed bees whose solutions are abandoned will be converted into scouts. This latter will generate new random food sources using Eq. 21.

Various works based on the ABC algorithm have been published in the literature to handle the problem of finding optimum sensor locations while optimizing several objectives [49], [65], [107], [132], [133], [134]. Authors in [133] considered the energy-efficient dynamic deployment of homogeneous mobile sensor nodes in a 2D environment. The proposed solution is based on the ABC algorithm, where a food source represents a WSN deployment and aims to maximize coverage with reduced energy consumption. This latter is handled by a routing mechanism based on RSSI measurement to determine the shortest paths between sensors. The work presented in [65] illustrates a relay nodes deployment method to eliminate the communication holes caused by a random deployment of static sensor nodes in a 3D environment. The designed deployment system is based on the ABC algorithm, and it intends to boost the network lifespan under the cost constraint. Indeed, the first phase of the system is to build the network backbone by deploying a minimum number of relay nodes with the Minimum Spanning Tree algorithm. Then in the second phase, the ABC algorithm is used to optimize the objective function with the required network connectivity. In [132], the authors applied the multi-objective bee algorithm to minimize two objectives: none covered area and none connected sensors deployed in a 3D environment. The algorithm adopts the ranking mechanisms of NSGA-II and the levy flight random walk to avoid the local optima. Another WSN deployment approach based on the ABC algorithm was presented in [49]. The authors considered the coverage maximization objective and proposed two adjustments to accelerate the algorithm convergence. The first adjustment concerns the Onlooker bee phase, where the creation of a new individual is parameterized by two new factors: the neighbor factor and the dynamically decreased forgetting factor. The second adjustment is introducing a backward learning strategy in the scout phase.

e: GREY WOLF OPTIMIZER (GWO)

Grey wolf Optimizer was first proposed by Mirjalili et al in 2014 [135]. It is inspired by the leadership hierarchy and predation process of grey wolves in nature. The leadership hierarchy in the pack is divided into four types of grey wolves: alpha, beta, delta, and omega, and the dominance decreases from alpha to omega. To mathematically model it, the GWO algorithm considers the fittest solution as the α grey wolf, the second-best solution and the third-best solution as β grey wolf and δ grey wolf respectively, and the rest of the population is assumed to be ω grey wolves; therefore, the population will be guided by the first three best solutions. In addition, the GWO algorithm implements the main phases of the grey wolves hunting technique as follows:

- Encircling prey: The mathematical model of encircling prey is defined as follows [135]:

$$\vec{D} = |\vec{C}.\vec{X_p}(t) - \vec{X}(t)| \tag{24}$$
$$\vec{X}(t+1) = \vec{X_p}(t) - \vec{A}.\vec{D} \tag{25}$$

where $\vec{X_p}(t)$ indicates the position vector of the prey at iteration t, $\vec{A}$ and $\vec{D}$ are coefficient vectors, and $\vec{X}(t)$ is the current position of the grey wolf. $\vec{A}$ and $\vec{C}$ are calculated as follows [135]:

$$\vec{A} = 2.\vec{a}.\vec{r_1} - \vec{a} \tag{26}$$
$$\vec{C} = 2.\vec{r_2} \tag{27}$$

where $\vec{a}$ is linearly decreased from 2 to 0 and r1, r2 are random vectors in [0, 1].

- Hunting: GWO algorithm simulates the prey hunting behavior mathematically by updating the position of wolves according to α, β and δ wolves positions as follows [135]:

$$\vec{D_\alpha} = |\vec{C_1}.\vec{X_\alpha} - \vec{X}|, \quad \vec{D_\beta} = |\vec{C_2}.\vec{X_\beta} - \vec{X}|, \quad \vec{D_\delta} = |\vec{C_3}.\vec{X_\delta} - \vec{X}| \tag{28}$$
$$\vec{X_1} = \vec{X_\alpha} - \vec{A_1}.\vec{D_\alpha}, \quad \vec{X_2} = \vec{X_\beta} - \vec{A_2}.\vec{D_\beta}, \quad \vec{X_3} = \vec{X_\delta} - \vec{A_3}.\vec{D_\delta} \tag{29}$$
$$\vec{X}(t+1) = \frac{\vec{X_1} + \vec{X_2} + \vec{X_3}}{3} \tag{30}$$

In addition, the GWO algorithm simulates the wolves attack on the prey by updating the $\vec{a}$ parameter. The pseudo-code of GWO algorithm is depicted in algorithm 5. The GWO algorithm was applied in several works to deal with the sensors deployment [51], [136], [137], [138]. Authors in [51] developed an enhanced version of the GWO algorithm to deploy WSN in a 3D environment with the objective of coverage maximization under the connectivity constraint. For the first enhancement, the authors used the Tent map that generates chaotic research sequences, increasing population diversity and promoting algorithm exploration to escape the local optima. Another enhancement was the suggestion of a new position update strategy that splits the population equitably into an inner layer group to perform the inner layer encircle and an outer layer group to perform the outer layer encircle. The inner layer encircle focuses on the exploitation aspect of the algorithm and hence, impact the convergence speed. The outer layer encircle focuses on the exploitation aspect. The work in [137] focused on the coverage rate, the sensor nodes' distribution uniformity, and the average moving distance. They applied a Lévy-embedded Grey Wolf Optimization (LGWO) algorithm, which combines the GWO algorithm with the Lévy flight to enhance the searching mechanism and avoid the local optima. Additionally, the virtual

Algorithm 5 Pseudo Code of GWO Algorithm

```
Initialization of grey wolves
Initialization of parameters: a, A and C
Assess the fitness value for each grey wolf
X_α= the best candidate
X_β= the second best candidate
X_δ= the third best candidate
while Termination condition is not satisfied do
    for Each grey wolf do
        Update position
    end for
    Update parameters a, A and C
    Assess the fitness value for each grey wolf
    Update X_ω, X_β, X_δ
end while
Return X_α
```

force algorithm was introduced in sensor nodes' positions updating to maintain a connected network. Authors in [136] proposed another approach based on GWO to tackle the problem of WSN deployment with three objectives: coverage rate, connectivity, and network energy. The proposed approach called a behavior-based grey wolf optimizer (BGWO), simulates two wolf groups' natural behaviors, namely the lost wolf strategy and the mating strategy. The lost wolf strategy allows the wolf pack to get rid of wolves with low fitness, and it is used to prevent the algorithm from falling in the local optima. In contrast, the mating strategy applies genetic operators to produce new individuals. Also, all the male wolves compete to mate instead of prioritizing only α wolves. A GWO variant called GWO-EH was presented in [139], with the objective of optimizing the WSN coverage. GWO-EH reinforces the exploration and exploitation processes through the improvement of the position-updating equation of the leading wolves and the repositioning of the worst three wolves around the leading wolves respectively. Furthermore, the hunting mechanism was also adjusted to involve the α wolves, β wolves, and γ wolves in the research process according to their ranks in the leadership hierarchy.

f: BAT ALGORITHM (BA)

Bat algorithm is a swarm intelligence algorithm proposed by Yang and Gandomi [140]. It is based on the echolocation behavior of bats, which allows for the identification of prey and the avoidance of obstacles even in low-light conditions [140]. The echolocation behavior is the process of emitting signal pulses and receiving their reflected echoes from objects in the vicinity. It has three main features: a frequency that varies from f_{min} to f_{max}, an emission rate, and loudness. The features are modeled according to three rules [140]:

- All the bats use echolocation to measure distance and differentiate between food/prey and background obstacles.
- To find prey, bats fly arbitrary with velocity v_i at position x_i, with a constant frequency f_{min}, changing wavelength λ, and loudness A_0. Depending on the closeness of their target, they may dynamically modify the wavelength (or frequency) of their generated pulses as well as the rate of pulse emission r in the range of [0, 1].
- The loudness ranges from large positive value A_0 to minimum constant value A_{min}.

The bat position represents the solution to the problem, and it is updated according to the following equations:

$$f_i = f_{min} + (f_{max} - f_{min}) \times \beta. \tag{31}$$

$$v_i^t = v_i^(t-1) + (x_i^t - x_*) \times f - i \tag{32}$$

$$x_i^t = x_i^(t-1) + v_i^t \tag{33}$$

where f_{min} and f_{max} represent the minimum and maximum frequency, respectively. x_* is the position of the fittest individual in the population, and $\beta \in [0, 1]$ is a random vector generated from a uniform distribution. The BA algorithm was adapted in several approaches to solve the problem of WSN deployment [64], [141], [142]. Authors in [64] presented a smart BA (SBA) based solution for the WSN deployment problem in a 3D environment. In their method, the search behavior of bats is more intelligent than the one proposed in the original algorithm since it encompasses the decision theory and fuzzy logic techniques. The decision theory is used in the utility function to direct the search for artificial bats. This allows achieving a good exploration of the research space without stagnation in the local optima. Next, the direction utility value is used with other parameters to set the velocity and the frequency of bats by means of a fuzzy logic inference-based technique. In addition, the SBA algorithm is carried out in two stages. The first stage tries to position sensor nodes in a 3D grid while optimizing a weighted sum fitness function that combines the coverage rate and the deployment cost, further in the second stage, the relay nodes are positioned to optimize three objectives which are connectivity quality, the fault-tolerance quality and the number of deployed relay nodes.

g: FLOWER POLLINATION ALGORITHM (FPA)

Flower pollination algorithm is inspired by the pollination process of flowering plants and was proposed by Xin-She Yang in 2012 [143]. FPA assumes that each plant has a single flower, and each flower produces a unique pollen gamete that represents a solution to a given problem. In order to exchange information between flowers, FPA uses two mechanisms: global pollination and local pollination, and it switches between the two modes of pollination using a switch probability $p \in [0, 1]$. Global pollination simulates natural cross-pollination, where pollen is carried over long distances by pollinators (insects and birds). Therefore, distant flowers can exchange information with each other, and this process is represented mathematically as follows [143]:

$$X_i^{t+1} = X_i^t + L(X_i^t - g^*) \tag{34}$$

where X_i^t is the ith solution vector (pollen) at iteration $t+1$, g^* is the best solution at the iteration i, and parameter L is the step size which is the strength of the pollination. It obeys Levy distribution in order to simulate the movement of insects [143]. Local pollination allows the exchange of information between the nearby flowers through the wind. It is described mathematically as follows [143]:

$$X_i^{t+1} = X_i^t + \epsilon(X_j^t - X_k^t) \tag{35}$$

where X_i^{t+1} is the i-th solution at iteration $t+1$, X_i^t, X_j^t and X_k^t respectively represent the i-th, j-th and k-th flowers in the current iteration (t). The algorithm 6 represents the main steps of FPA. The work presented in [144] adapted the multi-objective FPA to solve the WSN deployment problem. Its fitness function considers both the coverage rate with a 2D binary sensing model and the energy consumption of the network. The initial population is randomly generated, and each individual represents a connected WSN where a sink node is positioned in the center of the target area. The proposed method is basically based on the classical steps of multi-objective FPA, and it outperforms, according to the simulation results, the PSO algorithm in both coverage and energy consumption. Wang et al. [145], conceived two optimization approaches based on FPA to deploy optimally heterogeneous sensors within a monitoring area with obstacles. The first approach improves the classical FPA and considers only the coverage rate under the connectivity constraint. The improved FPA uses a nonlinear convergence factor strategy to restrict the original scaling factor. Further, the Tent chaotic map is used in population initialization to boost the diversity of individuals and avoid the problem of iteration stagnation. The final improvement was applying a greedy crossover strategy after the local or the global pollination to boost the solution accuracy further. The second approach is based on a non-dominated sorting multi-objective FPA. It aims at optimizing the coverage rate, the minimum radiation overflow rate, and the WSN energy consumption rate. The first proposed improvement of this method is the use of external archive strategy and leader strategy in the global pollination phase to direct the search toward the current non-dominated solutions set. Moreover, The NSGA-II elite technique is used to preserve good solutions in the parent population so that the algorithm convergence can be boosted. At the same time, the computation of the degree of crowding is adjusted to avoid losing population diversity. According to the simulation results, both approaches provide good solutions with an optimized convergence performance.

Algorithm 6 Pseudo Code of Flower Pollination Algorithm

```
Initialization of flowers /pollen gametes
Find the best solution g*
Set the switch probability p ∈ [0, 1]
while Termination condition is not satisfied do
  for Each flower do
    if rand < p then
      Perform global pollination mode
    else
      Perform local pollination model
    end if
    Assess the fitness of the new flower
    if the new solution is better than the best solution then
      Update the best solution g*
    end if
  end for
end while
Return g*
```

B. HYBRID METAHEURISTIC

Despite their good performance in looking for near-optimum solutions in polynomial time, stand-alone metaheuristics still have limitations and drawbacks, such as premature convergence and low accuracy of solutions. This has motivated researchers to turn toward other optimization strategies based on the hybridization of algorithms [146]. Hybridization seeks to integrate two or more algorithms with complementary features to capitalize on and reap the benefits of their advantages [147]. The hybrid metaheuristics can be categorized into collaborative hybrids and integrative hybrids [148]. In collaborative hybrids, the combined algorithms work in multi-stage, sequentially, or in parallel as depicted in Fig. 9. For integrative hybrids, a subordinated algorithm is embedded in a master metaheuristic with a contributing rate between 10% to 20% [148]. Several hybrid-metaheuristics based approaches were designed to deal with the WSN deployment problem [149], [150], [151], [152], [153], [154], [155], [156], [157], [158], [159], [160]. In the solution presented in [149], the authors conceived two hybrid algorithms, namely Hybrid-MOEA/D-I and Hybrid-MOEA/D-II, to simultaneously optimize three conflicting objectives: coverage rate, energy consumption, and equilibrium of energy consumption while positioning the sensor nodes in the target area. The last objective intends to limit the amount of energy consumed by a subset of sensor nodes in the WNS. For Hybrid-MOEA/D-I, the authors combined the MOEA/D framework with the three reproduction operators of GA and the differential evolutionary algorithm (DE): selection, crossover, and mutation. These operators are randomly selected for each sub-problem to increase the population diversity. The Hybrid-MOEA/D-II was developed by combining the Hybrid-MOEA/D-I with the discrete binary particle swarm optimization (DBPSO) to schedule sensor nodes and hence, boost the network lifetime. Another hybrid metaheuristic-based approach is proposed by Mnasri et al. in [150] to investigate the problem of finding the optimal 3D locations for additional nodes to an already deployed WSN. The approach combines the NSGA-III with the ACO to redress the low selection pressure problem of NSGA-III while maximizing coverage and keeping the ACO from falling into

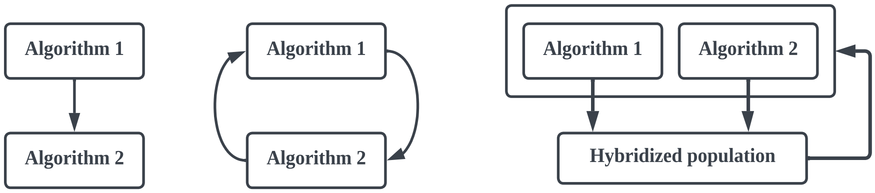


FIGURE 9. Collaborative framework of hybrid algorithm, depicting multi-stage, sequential, and parallel structures [148].

the local optima. ACO constructs the initial population of NSGA-III to produce only feasible solutions, and then for each iteration, the approach applies the classical steps of NSGA-III to create new solutions. These solutions are used in the next step to update the value of pheromones as a means to guide the search for fitter future solutions. A further hybrid method for WSN deployment is detailed in [151], it combines GA with binary ACO, which uses the binary coding of individuals. It also optimizes a mono objective fitness function that assesses the covered area and the number of working nodes. The initial population is randomly generated and enhanced using a repeated execution of genetic reproduction operators. Moreover, the new solutions are utilized to update information pheromones of the WSN. In the main loop, the algorithm executes the ants' traverse, updates the pheromone, and carries out the genetic crossover and mutation operators on the new solutions until the stopping criterion is met. In [161], the BA and the Grasshopper Optimization Algorithm (GOA) were hybridized to resolve the dynamic deployment problem of WSNs. The BA is known for its random behavior in both exploitation and exploitation phases. This reduces the algorithm's precision and convergence rate. To remedy these shortcomings, the authors applied the GOA algorithm in the exploitation phase to accurately exploit the neighborhood. GOA is a recent nature-inspired algorithm developed by Saremi et al. [162]. This algorithm mimics the behavior of grasshopper insect movement in searching for an optimal solution by aggregating the social interaction behavior, the gravity force factor, and the wind advection factor in the same mathematical formula that computes the grasshopper position. Consequently, the hybrid BA changes each forager location depending on its current position, the position of the best solution in the neighborhood, and the position of all other foragers in the related neighborhood. This guides the BA search process to a more accurate solution within a reasonable convergence time. Chen et al. [156] conceived a hybrid framework based on an evolutionary algorithm called memetic algorithm and a heuristic recursive algorithm, designed to ensure a permanent full coverage with an extending network lifespan. Each potential solution of the memetic algorithm contains disjoint sets of sensor nodes that are sequentially activated using a scheduling mechanism. The heuristic recursive algorithm is developed to cope with the coverage hole problem caused by node failure or energy exhaustion through the activation of other nodes in other sets. The authors performed real-world tests to evaluate their approach and compared it with other existing solutions through computer simulations. The results revealed that the hybrid framework outperformed other algorithms in terms of network lifetime and this is for variant experimental conditions. Another hybrid solution was proposed in [157]. It combined the GA and the Binary PSO to compute the optimal deployment scheme with maximized coverage and connectivity and minimized cost. The hybrid algorithm began by creating the initial population and evaluating the fitness of individuals, then for each iteration, the population is spitted into two groups, the first group represents the input of GA and it encompasses the individuals with the highest fitness, and the worst solutions are sent to the binary PSO. With this solution scheme, the authors aimed at creating new solutions with evolved performances using the GA operators, and exploring other directions in the search space using the Binary PSO. The last step consists of merging the two outputs of GA and Binary PSO into one population for the next iteration. In the work presented in [158], the authors developed a hybrid search for the optimal WSN deployment based on the PSO and Hooke–Jeeves search method. The PSO is used to perform the global search in the search space. If the global solution is not improved after a preset number of iterations, the hybrid solution applies the Hooke-Jeeves method to carry out a local search in the neighborhood of the global best solution to improve its coverage and ensure faster convergence.

El Khamlichi et al. [159], designed a hybrid approach based on gradient method and Simulated Annealing algorithm to deal with sensor nodes placement problem. The simulated Annealing algorithm is a local search metaheuristic that adopts the hill climbing moves to escape the local optima [163]. The hybrid approach has the objective of deploying the necessary number of sensor nodes to achieve at least 1-coverage and 1-connectivity, and it consists mainly of three major steps. The first step is to position the sensor nodes in the target area using a triangular grid deployment technique with a preset distance between every two sensors, then in the second step, the gradient method is applied to reposition sensors placed on the area boundary to improve the network coverage. Finally, the connectivity constraint is ensured by adding new sensor nodes to fill the gap between connected and isolated sensors. The work presented in [164] deals with the problem of wireless body area network design and routing. The proposed design takes into account the traffic uncertainty caused by the body sensors' variable rate data generation. Additionally, the authors have considered single-hop routing and modeled the installation of relay nodes as a binary linear program. The problem was solved using a hybrid algorithm that combines a heuristic using randomized algorithms and Approximate Nondeterministic Tree-Search (ANTS) and an exact binary linear program that represents a large-variable neighborhood search. The hybrid algorithm directs the process of variable fixing during the feasible solutions' construction using linear relaxations for the same problem. A total of 30 realistic instances were used in the experiments, and the results confirm that the proposed algorithm outperforms the CPLEX solver in terms of speed and solution quality. The same authors have developed a hybrid solution for designing a body WSN in [165]. They proposed an Integer Linear Programming heuristic based on deterministic and probabilistic variable fixing methods. Also, the body WSN design problem was modeled with a scenario-based min-max robust optimization model to consider the data uncertainty generated by the biosensors. The objective function represents the network energy consumption to be reduced and the constraints ensure the balance between the ongoing and outgoing flows.

Although the performance enhancement that the meta-heuristic hybridization could provide, it is still not sufficiently investigated by the research community working on WSNs deployment. Therefore, it is highly encouraged to consider hybridizing metaheuristics to address the WSNs deployment problem in order to accurately direct the search and avoid local optima.

The statistical data presented in Fig. 10 were collected from the research papers presented in Table 5 in order to clearly highlight the current research directions in the WSNs deployment problem, namely commonly used sensing and communication zone modelings, the environment dimension, and the obstacles heterogeneity.

As shown in Fig.10a, 68.4% of works employed the binary sensing model to estimate the network's coverage rate, compared to 31.6% of works that used probabilistic models. Furthermore, Fig.10b shows that about one-third of reported techniques do not include connectivity in their solutions, and only 29.8 % consider a probabilistic communication model. These results should encourage the research community to explore more these two aspects since they may result in a significant mismatch between theoretical and real-world WSNs performances. In addition and as previously explained, the environment dimension and the obstacles modeling are two critical factors influencing the WSNs deployment solution. According to Figs. 10c and 10d, more than two-thirds of examined papers consider the target area as a 2D plane with no obstacles. 12.3% consider homogeneous obstacles, which are mostly viewed as opaque objects, and only 10.5% consider obstacles with different impacts (signal attenuation) on the sensing and communication zones of the sensor nodes.

C. MACHINE LEARNING

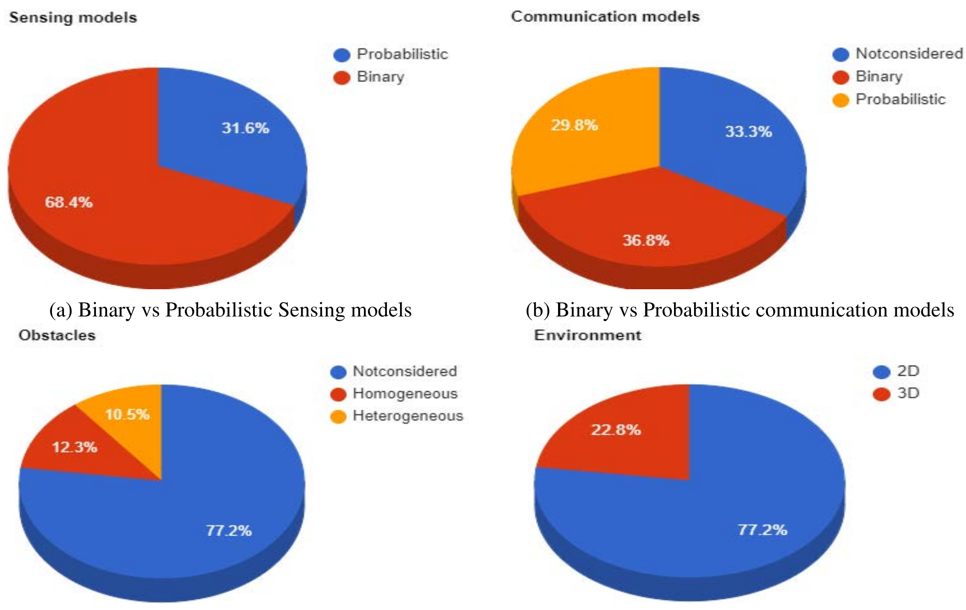
Machine Learning (ML) is a fundamental branch of AI that represents the intersection of computer science and statistics [175]. It enables computer systems to automatically learn from a huge amount of data and make predictions without being explicitly programmed for the task. The ML algorithms are classified into four categories based on the classification of the training data, these families are Supervised learning based on labeled data, Unsupervised learning based on unlabeled data, Semi-supervised learning based on a mixture of classified and unclassified data, and Reinforcement learning which does not require data. Several initiatives based on ML were proposed to deal with functional and nonfunctional aspects related to WSNs such as data aggregating, routing, localization, security, resource management, and sensors placement. [176], [177], [178], [179], [180]. Authors in [181] developed an environmental sensor deployment algorithm based on a multi-response Taguchi-guided k-means clustering embedded GA. The deployment algorithm considers the coverage, connectivity, network lifetime, fault tolerance, and HVAC airflow optimization objectives, and it is conducted in three main stages. The initial stage of the deployment strategy is to determine the sensor locations that will provide an optimized network lifespan with a low installation cost. In the second stage, the network connectivity and the deployment cost of the relay nodes are addressed. Finally, the third stage considers the development of the entire system at the physical, network, and application layers with the aim of minimizing the total number the deployed sensor and relay nodes, while preserving the network performance. The multi-response Taguchi method has been applied to identify the best values for the crossover rate, the mutation rate, the population size, and the number of clusters in k-means clustering. The k-means clustering is a machine learning method that aims at partitioning a set of observations into K clusters, with each observation belonging to the cluster with the closest centroid. It has been used in the deployment scheme to select the best cluster for the initial population with the best set of chromosomes in order to improve the convergence and computational time of the solution. The authors in [182] suggested a hybrid distributed approach for coverage hole healing based on game theory and Q-learning. Each mobile sensor node is depicted as a player that can compute its new position autonomously and in a decentralized manner. In their approach, the authors formulated the problem as a potential multiplayer game in which each player has to choose a combined action to improve simultaneously both coverage by reducing the overlapped zones and power consumption by adjusting the sensing radius and minimizing the motion energy. Further, the process of computing the payoff function is based on a multi-agent Q-learning algorithm since it involves the player's profile alongside the actions of its neighbors. The distributed payoff-based Q-learning algorithm is divided into two main phases. The first phase consists of selecting actions and updating states according to the exploration and exploitation processes and the second

TABLE 5. Comparison between WSN deployment related works.

Work	Year	Objectives	Constraints	Methods	Environment	obstacles	Limits
[77]	2017	Coverage Energy consumption	Ensure at least p% coverage (p is input data)	Evolutionary algorithm	2D	No obstacles	Homogeneous sensors Connectivity is not considered Binary sensing model
[166]	2020	Coverage Overlapped area Minimize the number of sensors	Connectivity	GA	2D	No obstacles	Homogeneous sensors Binary sensing model Binary communication model
[167]	2016	Coverage Overlapped area	-	GA	2D	No obstacles	Homogeneous sensors Connectivity not considered Binary sensing model
[46]	2020	Coverage Minimize the number of sensors	-	NSGA-II	3D	Homogeneous	Homogeneous sensors Connectivity is not considered
[168]	2020	Coverage, Cost, Localisation Connectivity, Energy consumption, Network lifetime Network utilization	-	εNSGA-II U-NSGA-III MOEAD/DD	3D	No obstacles	Homogeneous sensors
[52]	2020	Coverage, K-coverage Overlapping coverage, Cost	M-connectivity Limited budget	GA	2D	Heterogeneous	-
[53]	2020	Coverage, K-coverage Overlapping coverage, Cost	Connectivity	GA	2D	Heterogeneous	Binary sensing model
[54]	2017	Coverage Cost Connectivity Over coverage	-	GA NSGA-II	2D	Heterogeneous	Binary sensing coverage
[80]	2020	Coverage Cost	Connectivity	NSGA-II	2D	Heterogeneous	Homogeneous sensors
[81]	2016	Coverage Network lifetime	Connectivity	constrained Pareto-based multi-objective evolutionary approach	2D	No obstacles	Homogeneous sensors Binary sensing model Binary communication model
[82]	2016	Coverage Minimize the number of sensors	-	NSGA-II SPEA2 SMSEMOA MOEA/D	2D	No obstacles	Homogeneous sensors Binary sensing model
[169]	2019	Coverage Energy consumption	-	NSGA-III	2D	No obstacles	Homogeneous sensors Binary sensing model Connectivity not considered
[95]	2017	Energy consumption Average sensitivity area Network reliability	Coverage all target points	modified MOEA/D	2D	No obstacles	Homogeneous sensors Binary sensing model Connectivity not considered
[96]	2013	Coverage Energy consumption Network lifetime	Connectivity between sensors and sink node	MOEA/D-DE	2D	No obstacles	Homogeneous sensor nodes Binary communication model Binary sensing model
[98]	2011	Coverage Lifetime	K-connectivity	MOEA/D	2D	No obstacles	Homogeneous sensor nodes Binary sensing model
[97]	2010	Coverage Lifetime	Connectivity	MOEA/D	2D	No obstacles	Homogeneous sensors Binary sensing model
[107]	2020	Coverage Lifetime	Connectivity	BA PSO	2D	No obstacles	Homogeneous sensors Binary sensing model Binary communication range
[113]	2012	Coverage Network lifetime	Connectivity	Multi objective PSO	2D	No obstacles	Homogeneous sensors Binary communication model
[114]	2019	Connectivity rate Utilization of the network Coverage Quality of links Rate of localization	-	bird's accent-based many objective PSO	3D	No obstacles	Homogeneous sensors Binary sensing model
[47]	2018	Coverage Lifetime	Reliability	cooperative coevolutionary PSO comprehensive learning PSO	3D	Homogeneous	Binary communication model
[106]	2020	Coverage	Connectivity	Distributed PSO combined with 3D VF algorithm	3D	Heterogeneous	Homogeneous sensors Binary communication model
[108]	2020	Coverage Reduce moving distance of mobile nodes	-	PSO combined with VF algorithm	2D	No obstacles	Homogeneous sensors Connectivity is not considered Binary sensing model
[109]	2021	Coverage Sleep rate Coverage uniformity	-	Binary PSO	2D	No obstacles	Homogeneous sensor nodes Connectivity not considered Binary sensing model
[110]	2020	Coverage Network lifetime	-	cooperative PSO cooperative PSO using fuzzy logic	2D	No obstacles	Homogeneous sensor nodes Connectivity is not considered Binary sensing model
[111]	2020	Coverage	Energy consumption	quantum behaved PSO	2D	No obstacles	Homogeneous sensor nodes Connectivity is not considered Binary sensing model
[112]	2017	Coverage Reduce moving distance of mobile nodes	-	Multi swarm PSO	2D	No obstacles	Homogeneous sensor nodes Connectivity is not considered
[119]	2017	Cost	Meet a specified minimum level of reliability	ACO	2D	No obstacles	Homogeneous sensor nodes Binary sensing range Binary communication range

TABLE 5. ***(Continued.)*** **Comparison between WSN deployment related works.**

[124]	2016	Minimize the number of sensors	Coverage Connectivity	Dynamic ACO	2D	No obstacles	Homogeneous sensors Binary sensing model Binary communication model
[122]	2018	Cost	Coverage Connectivity	ACO	3D	No obstacles	Homogeneous sensor nodes Binary communication range Binary sensing range
[121]	2014	Cost	Coverage Connectivity	ACO	2D	No obstacles	Binary sensing model Binary communication model
[123]	2018	Cost	Coverage Connectivity	ACO	2D	No obstacles	Binary sensing model Binary communication model
[120]	2015	Minimize the number of sensors	Coverage Connectivity	ACO Culture algorithm	2D	No obstacles	Homogeneous sensor nodes Binary sensing model Binary communication model
[126]	2018	Coverage	-	ALO Algorithm	2D	No obstacles	Homogeneous sensor nodes Binary sensing model Connectivity is not considered
[132]	2019	Coverage Connectivity	-	multi-objective Levy flight bee algorithm	3D	No obstacles	Binary communication model Binary sensing model
[49]	2013	Coverage	-	modified ABC	2D	Homogeneous	Homogeneous sensors Connectivity is not considered Binary sensing model
[134]	2014	Coverage	-	improved ABC	2D	No obstacles	Homogeneous sensor nodes Connectivity is not considered Binary sensing model
[65]	2016	Network lifetime	Cost budget Connectivity	ABC	3D	No obstacles	Homogeneous sensor nodes
[51]	2020	Coverage	-	Enhanced GWO	3D	Homogeneous	Homogeneous sensor nodes Binary sensing model Connectivity is not considered
[136]	2020	Coverage Energy consumption	Connectivity	Behavior-based WO	2D	No obstacles	Homogeneous sensor nodes Binary communication model
[137]	2019	Coverage	Connectivity	Lévy-embedded GWO combined with VFA	2D	No obstacles	Homogeneous sensor nodes Binary communication model
[138]	2019	Coverage	Connectivity	WO	2D	Homogeneous	Homogeneous sensor nodes Binary sensing model Binary communication model
[64]	2018	Coverage Minimize the number of sensors Connectivity between Relay nodes Fault tolerance of relay nodes	-	Smart BA	3D	No obstacles	Homogeneous sensor nodes Binary sensing model Binary communication model
[141]	2021	Coverage	-	BA	2D	No obstacles	Homogeneous sensor nodes Connectivity is not considered Binary sensing model
[142]	2015	Coverage Energy consumption	Connectivity	multi-objective bat swarm optimization	2D	No obstacles	Homogeneous sensor nodes Binary sensing model Binary communication model
[145]	2019	Coverage Energy consumption Node radiation overflow rate	Connectivity	Improved FPA non-dominated sorting multi-objective FPA	2D	Hmogeneous	Binary sensing model Binary communication model
[144]	2017	Coverage Energy consumption	Connectivity	Multi-objective FPA	2D	No obstacles	Homogeneous sensor nodes Binary sensing model Binary communication model
[84]	2019	Coverage Reduce mobile nodes number Average mobile distance	-	CS algorithm	2D	No obstacles	Homogeneous sensors Binary sensing model Do not consider connectivity
[87]	2018	Coverage	-	CS algorithm Chaotic FPA	2D	No obstacles	Binary sensing model Connectivity is not considered
[88]	2020	K-coverage Network lifetime	-	CS algorithm	2D	No obstacles	Binary sensing model connectivity is not considered
[170]	2020	Coverage	-	Diversity Teams in Soccer League Competition Algorithm	2D	No obstacles	Homogeneous sensors Connectivity is not considered
[171]	2020	Coverage Cost	-	Harmony search	2D	No obstacles	Connectivity is not considered
[66]	2020	Coverage Cost	-	Harmony search	3D	No obstacles	Homogeneous sensor nodes Connectivity is not considered Binary sensing model
[172]	2016	Coverage	-	Search economics	2D	No obstacles	Homogeneous sensor nodes Connectivity is not considered Binary sensing model
[173]	2017	Cost	Coverage Connectivity	Integer linear programming	2D	-	Homogeneous sensor nodes Do not specify the propagation model used to assess connectivity
[174]	2019	Cost Network lifetime	Connectivity Ensure that a specific coverage rate is met	GA based algorithm	2D	No obstacles	Homogeneous sensors
[56]	2021	Coverage	-	Greedy algorithm	3D	No obstacles	Homogeneous sensors Connectivity is not considered
[67]	2014	Coverage	-	Gradient descent	3D	Homogeneous	Homogeneous sensors Connectivity is not considered



(a) Binary vs Probabilistic Sensing models
(b) Binary vs Probabilistic communication models
(c) Considered obstacles
(d) Considered environment

FIGURE 10. Distribution of sensing models, communication models, obstacles heterogeneity and environment dimension.

phase represents the learning phase which focuses on the other players' actions in order to react appropriately while repairing newly formed coverage holes. The simulation result shows that the distributed approach reached a better trade-off between the coverage and the energy consumption compared to other solutions proposed in the literature. Another work in [183] proposes a coverage hole detection and recovery approach based on a multi-intelligent agent-enabled reinforcement learning algorithm. In the first stage of the solution, the authors used the Sierpinski cluster-tree topology construction method to partition the sensor network into a set of unequal clusters. Then, they applied the multi-objective black widow optimization algorithm for the cluster head selection and a new Tsallis entropy-enabled Bayesian probability (TE2BP) algorithm for the dynamic scheduling of the sensor nodes. The goals of these previous phases are to ensure an energy-efficient transmission, minimize data loss, enhance energy consumption, and reduce the probability of coverage holes occurring. In the second stage of the solution, the virtual sector-based hole detection protocol is applied to detect the existing coverage holes in each cluster, then each coverage hole is healed using the multi-agent SARSA (State-Action-Reward-State-Action) algorithm. This algorithm takes as an input the coverage hole location and the list of mobile nodes around it and determines the optimal mobile node to heal the hole as an output. The factors that allow to SARSA algorithm to learn the environment are distance, node lifetime, and coverage level. The simulation tests show that the proposed algorithm outperforms existing solutions in the literature in terms of coverage rate, the number of dead nodes, average energy consumption, and throughput.

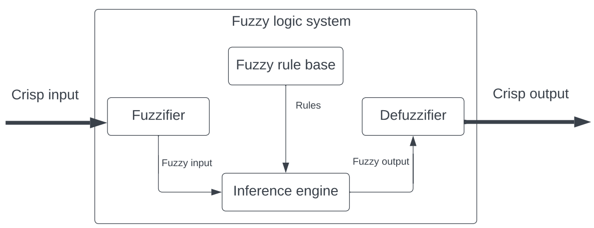


FIGURE 11. Fuzzy logic system architecture.

D. FUZZY LOGIC

Fuzzy logic was proposed by Zadeh [184] in 1965 to extend the Boolean logic, it enables several truth values to be handled by the same variable. The fuzzy logic system has four main components (see Fig. 11):

- Inference engine: It is in charge of applying the fuzzy rules to the fuzzy input in order to produce the fuzzy output.
- Fuzzy rule base: It contains the IF-THEN rules introduced by experts.
- Fuzzifier: It transforms the crisp values into fuzzy values.
- Defuzzifier: It maps the output of the fuzzy engine (fuzzy set) into crisp values.

Various research efforts have modeled the uncertainty aspect of WSN using Fuzzy logic, some of them have focused mainly on the deployment process [185], [186], [187], [188], [189]. Authors in [185], developed a sensor nodes deployment approach based on fuzzy logic. The proposed approach partitioned the area of interest into square subareas, each with its terrain profile and its needed degree of coverage.

Further, the authors regarded targets to be signal sources, yet, they considered that each subarea i has a PL_i and PL_{thi} depending on the types of obstacles within it. The PL and PL_{th} of each subarea are assumed to have three overlapping membership functions that encompass the entire input space and they are used to define the rule base for the fuzzy deployment. This latter is then applied to compute the required sensor number to be deployed for each subarea. Authors in [188], tackled the problem of coverage hole healing with a Fuzzy Logic and Shuffled Frog-Leaping approach. The proposed approach is carried out in four main phases: overlapped area estimation, sensor nodes scheduling, prediction of death time of sensor nodes, and computing the redeployment scheme of mobile sensors to heal the coverage holes. To begin, the authors assumed that the WSN is comprised of N randomly distributed sensors with different sensing and communication ranges and energy sources, and each sensor is conscious of its own location as well as the location of the base station. Further, each sensor can compute its overlapped coverage area with its neighbors based on a geometric distributed scheme. This information, along with the distance between a sensor node and the base station and residual energy, is then employed as an input parameter by the fuzzy scheduling method and results in 27 fuzzy rules. The last stage depicts the redeployment of mobile sensors to heal coverage holes based on the Shuffled Frog-Leaping algorithm, with each frog representing the mobile sensor nodes chosen to cover holes and their new locations. Simulation results indicate that the solution outperforms three existing approaches in the literature, in terms of coverage rate and energy consumption. Another self-healing coverage hole based on fuzzy logic was proposed in [189]. In this solution, each sensor checks its neighbors' regular basis to identify noisy and dead nodes and consequently the location of the coverage hole that results from the nonoperational neighbor node. Once the positions of the coverage holes are identified, the sensors applied FLS2 to select the appropriate mobile node for coverage healing based on its available energy, its euclidean distance to the uncovered area, and node redundancy. experimental results show that the proposed algorithm is able to enhance the coverage with an optimized energy consumption compare to other existing approaches in the literature.

VI. SIMULATION AND RESULTS

A. SYSTEM MODEL

This section presents the WSN deployment model used in the simulation tests. We assume that all the sensor nodes are static and have the same sensing ability in all directions.

1) NETWORK MODEL

We assume that the area of interest is a 2D free-obstacle plane of dimension $100m \times 100m$. We assume also that the sensor network comprises homogeneous sensor nodes with a sensing range of $8m$. Each algorithm attends to find the best position for each sensor node in order to optimize the network coverage. The number of sensor nodes for mono-objective algorithms is fixed at 45, allowing for a maximum coverage rate of 90%.

2) SENSING MODEL

For sake of simplicity, all the simulation tests adopt the binary sensing model described by Eq.1 in the section III-B.

3) OPTIMIZATION MODEL

1) Solution encoding and decision variables: For mono-objective algorithms, the deployment scheme is defined by the vector representation (see section III-A), each decision variable (x_i, y_i) fulfills the upper and lower bounds constraints, i.e: $0 \leq x_i \leq 100-R_s$, and $0 \leq y_i \leq 100-R_s$. The length of the array is equal to the number of the deployed sensor nodes (45).
   For multi-objective algorithms, the deployment scheme is defined by the grid representation (see section III-A). The grid has 100 rows and 100 columns. Each value c_i of the grid corresponds to 1 m^2 in the real environment. Also, each cell $c_{i,j}$ contains a binary value $V_{i,j}$, if $V_{i,j} = 1$, then the cell contains a sensor node, otherwise it is equal to 0.
2) Objective functions
   Coverage: as explained earlier in section III-B, the coverage refers to the ratio of the covered area by the WSN to the entire area of interest. In these simulation tests, we have used the Eq. 12 and Eq. 9 to compute coverage for mono-objective solutions and multi-objective solutions, respectively.
   Deployment cost: in the case of our tests, the cost function represents the number of the deployed sensor and it is computed as follows:
   $$\sum_{i=1,j=1}^{i=100,j=100} Vi,j$$
   where Vi,j refers to the cell at position (i,j) in the grid representation of the deployment solution.
3) Feasibility constraint: In the second part of the mono-objective simulation tests, we have considered the network's connectivity penalty in computing the score of a potential solution. The connectivity penalty refers to the number of isolated sensor nodes and it is equal to the difference between the number of the deployed sensor nodes and the number of sensor nodes belonging to the largest tree in the network. To compute it, we first established the connectivity matrix using the communication binary model (Eq. 17). Next, we applied the DFS algorithm [190] to generate a forest from the connectivity matrix, and then we maintained the largest tree in the forest to calculate the number of its nodes.

B. TESTED METAHEURISTICS

1) MOTIVATION OF THE SELECTION OF ALGORITHMS TO BE TESTED

The selection of these algorithms is based on the research presented in section V and summarized in Table 5. According to

TABLE 6. Time complexity and space complexity of the tested algorithms.

Algorithm	Algorithm complexity	parameters
GA	$O(K.N.M^2)$	K: number of iterations N: population size M: length of solution
PSO	$O(K.N.M^2)$	K: number of iterations N: population size M: length of solution
FPA	$O(K.N.M^2)$	K: number of iterations N: population size M: length of solution
ABC	$O(K.Max(NO, NE).MT.M^2)$	K: number of iterations NL: number of onlooker bees NE: number of employee bees MT: max trials M: length of solution
NSGA-II	$O(K.M.N^2)$	K: number of iterations M: number of objectives N: population size
MOEA/D	$O(K.T.N^2)$	K: number of iterations T: number of weight vectors N: population size

TABLE 7. Simulation parameters.

Algorithm	Parameter	Value
GA	Crossover probability	0.9
	Mutation Probability	0.015
	Selection technique	roulette wheel
PSO	c1 and c2	2
	Inetria weight	decreases from 0.9 to 0.4
FPA	switch probability	0.8
ABC	Max trials	100
	Selection technique of best food source	random
NSGA-II	Crossover probability	0.9
	Mutation probability	0.03
	Selection technique	Tournament
	Tournament size	2
	Tournament probability	0.9
	Population size	70
	Number of iterations	300
MOEA/D	Decomposition technique	Tchebycheff approach
	Selection technique in neighborhood	random
	Size of neighborhood	20
	λ_1 and λ_2	Tchebycheff approach
	Population size	80
	Number of iterations	300

this Table, the NSGA-II and the MOEA/D algorithms were used in the majority of the Pareto optimization-based techniques. As for single-objective approaches, we noticed that the most commonly used algorithms are the GA, which belongs to evolutionary algorithms, and PSO, which belongs to swarm optimization algorithms. Therefore, we chose to implement and test two multi-objective algorithms: NSGA-II and the MOEA/D, and the mono objective algorithms: GA and PSO, to which we have added ABC and FPA to enrich our comparison. All the algorithms were implemented in python, and simulation parameters are summarized in Table 7.

2) ANALYSIS OF TESTED ALGORITHMS

This section aims to measure the computational complexity (time complexity) of the chosen algorithms using the Big O notation. The intersection function in the fitness function is assumed to be an elementary operation with a complexity of $O(1)$. The complexity of algorithms is summarized in Table 6.

C. PROBLEM FORMULATION

For these simulation tests, the area of interest is depicted as a 2D plane without obstacles. The objective is to position the sensor nodes to reach maximum network coverage. We assume that the sensor network comprises 45 homogeneous sensor nodes with a sensing range of $8m$ for the mono-objective algorithms, and each algorithm attends to find the best coordinates (x_i, y_i) for each sensor node in an area of dimension $100m \times 100m$. The preset sensor number allows reaching a maximum coverage rate of 90%. To assess the coverage function, we use the binary sensing model (see Eq. 1) and calculate the union of detection zones of all sensor nodes without counting the duplicated overlapped area. For the multi-objective algorithms, we considered the cost function which seeks to reduce the number of deployed sensor nodes. Yet, the two conflicting objective functions to be optimized are coverage maximization and deployment cost minimization.

D. COMPARISON CRITERIA

For the mono objective algorithms, we focused on the following comparison criteria:

- The convergence of metaheuristics algorithms: We measured the evolution of the optimal solution over iterations for the GA, FPA, PSO, and ABC, The related results are depicted in Fig. 14.
- The diversity of metaheuristics algorithms: The diversity of the population means how much the individuals of a given generation (iteration) are different. It can be assessed in terms of individuals' fitness (behavior) or sensors dispersion (appearance) within the Target area. In our case, we have chosen to evaluate the population's diversity according to the fitness of their individuals. The diversity formula is as follows:

$$Pop_{diversity}(iter_i) = AVG[ABS((median(\vec{VF}) - \vec{VF})]$$

where $\vec{VF}$ is a vector containing the fitness value of all individuals with decreasing order, AVG function computes the average of a vector, ABS function computes the absolute value of each element in the vector and $median$ function computes the median of a vector. Each population has its diversity value and this criterion allows tracking the evolution of populations diversity over iterations, the corresponding result is shown in fig. 15 to 18

- The exploration and exploitation indicators of algorithms: The exploration and exploitation are two diversity-related indicators(related results are depicted in Figs. 19 to 22), we assume that the exploration indicator increases with the increase of diversity and the exploitation indicator decreases with the decrease of diversity. The exploration is computed as follows:

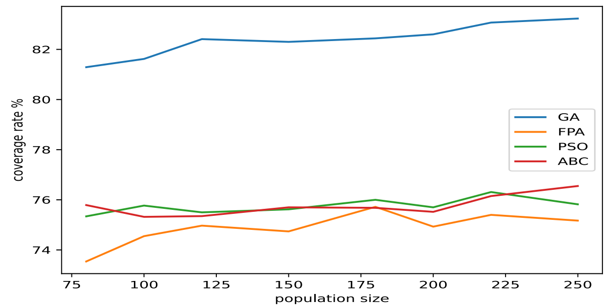
$$Pop_{exploration}(iter_i) = 100 \times \frac{Pop_{diversity}(iter_i)}{\max_{j\in[0,nb_{iter}]} Pop_{diversity}(iter_j)}$$

  Once the exploration is computed, we compute the exploitation indicator as follows:

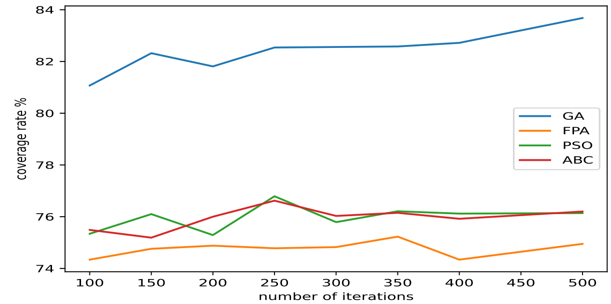
$$Pop_{exploitation}(iter_i) = 100 - Pop_{exploration}(iter_i)$$

  Before starting simulation tests, we have studied the variation of coverage rate and execution time as functions of population size and number of iterations as follows:
  - Impact of the population size on the coverage rate: To study the variation of the coverage rate according to the population size, we have set the number of iterations to 300 and varied the population size between 80, 100, 120, 150, 180, 200, 220 and 250. Then, for each case, we calculated the corresponding coverage rate for the GA, FPA, PSO and ABC. The corresponding results are shown in Fig. 12a.
  - Impact of the population size on the execution time: For this simulation test, we preserved the previous values of parameters, and for each population size, we calculated the corresponding execution time for the GA, FPA, PSO, and ABC. The test result is shown in Fig. 13a.
  - Impact of the number of iterations on the coverage rate: We set the population size to 150 and varied the number of iterations between 100, 150, 200, 250, 300, 350, 400, and 500. Then, for each case, we calculated the corresponding coverage rate for the GA, FPA, PSO, and ABC. The corresponding results are shown in Fig. 12b.
  - Impact of the number of iterations on the execution time: As with the prior test, we set the population size to 150 and varied the number of iterations between 100, 150, 200, 250, 300, 350, 400, and 500. Then, for each case we calculated the corresponding execution time for the GA, FPA, PSO, and ABC, The corresponding results are shown in Fig. 13b.

Fig. 12a and Fig. 12b present the variation of the coverage rate as a function of the population size and the number of iterations. According to Fig. 12a, Ga maintains a slight improvement of the coverage rate with the increase of the population size, FPA, PSO, and ABC, on the other hand, have fluctuated variations which means that the increase in population size does not necessarily improve the coverage. As for the first test, Fig. 12b shows that the GA outperforms PSO, ABC, and FPA. FPA has the worst performance with a maximum coverage rate of 75,23% reached at iteration 350. Both PSO and ABC reached their maximum coverage at iteration 250 with rates equal to 76,79% and 76.62%, respectively, and the maximum coverage rate of GA is 83.68% reached at iteration 500. Therefore, We can deduce that increasing the number of iterations does not always enhance the coverage rate of the four algorithms. Both Fig. 13a and Fig. 13b depict the evolution of the execution time as a function of population size and the number of iterations, respectively. GA has the

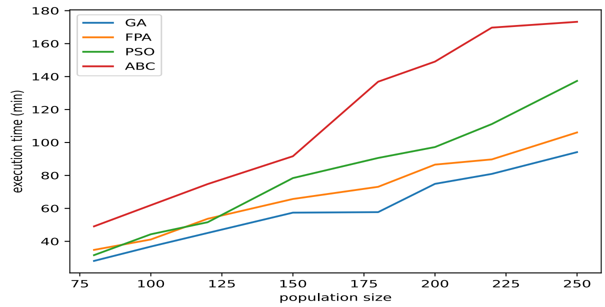


(a) Coverage rate vs population size

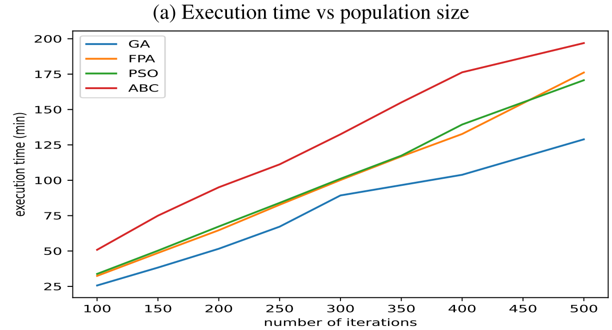


(b) Coverage rate vs number of iterations

FIGURE 12. Coverage rate variation as a function of population size and number of iterations.



(a) Execution time vs population size



(b) Execution time vs number of iterations

FIGURE 13. Execution time as a function of population size and number of iterations.

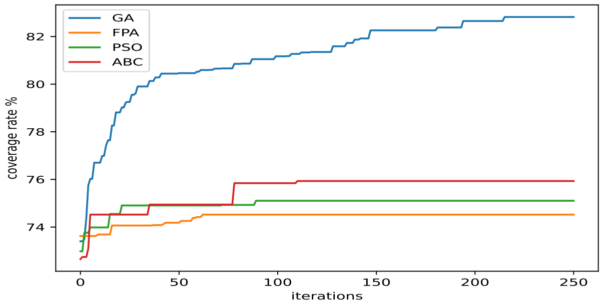


FIGURE 14. **Global convergence of GA, FPA, PSO, and ABC.**

fastest execution time in both cases, while ABC has the slowest. FPA and PSO have almost the same execution time in the second scenario, yet their two curves deviate when the population size increases. This gap can be linked to various reasons, including the values of each algorithm's parameters and its process for creating new generations. Based on the obtained curves depicted in Figs. 12a, 12b, 13a and 13b, we chose to set the population size and the number of iterations to 150 and 250 respectively, since these values allow for achieving acceptable coverage rates in relatively short execution time. Then we measured the global convergence for all algorithms (see Fig. 14) which refers to the best coverage rate researched at each iteration. As depicted in Fig. 14, GA outperforms PSO, FPA, and ABC with a maximal coverage rate of 82.82% against 75.10% for PSO, 74.52% for FPA and 75.93% for ABC. Furthermore, we observed a constant improvement in the coverage rate of GA vs a quick convergence of FPA, PSO, and ABC, which means that the three algorithms were in the local optima prior to iteration 130. We used the same settings and calculated the highest coverage rate of all algorithms while accounting for the connectivity penalty. PSO and ABC both attained 100% connectivity with coverage rates of 76.02% and 75.58% respectively. GA had a coverage rate of 82.57% versus 97.75% connectivity, while FPA had the lowest connectivity of 96.6% with a coverage rate of 75.3%.

Figs. 15 to 18 depict the diversity of GA, FPA, PSO and ABC, respectively. As mentioned earlier, diversity is considered as a score associated with a given population for a given iteration. It measures the average distance between the medium fitness of the population and the fitness of its individuals. A high diversity refers to a significant dissimilarity between two consecutive populations, and it is primarily due to the method of producing the next generation. For instance, the PSO algorithm updates at each iteration the position of particles by considering three different factors: the best global position, the best personal position, and the current position of a particle. Consequently, the diversity score of the new swarm might be considerably different from the previous one (see Fig. 17).

Similarly, the ABC algorithm updates the swarm using the exploration function of employee bees and the exploitation function of the onlooker bees. The two functions may lead to a very noticeable distinction between the new and the older food sources(see Fig. 18). Moreover, GA and FPA have a less destabilized diversity than PSO and ABC, with a slight fluctuation in the diversity curve of GA. This might be because both GA and FPA generation techniques of subsequent populations enable maintaining old individuals' properties, particularly for the FPA algorithm, where global pollination employs a small step size of the levy flight.

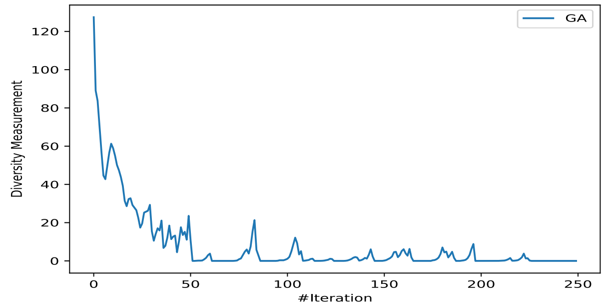


FIGURE 15. **GA diversity.**

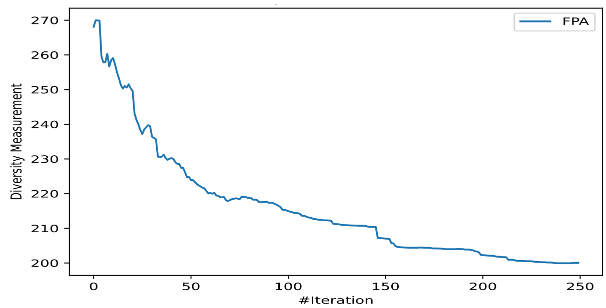


FIGURE 16. **FPA diversity.**

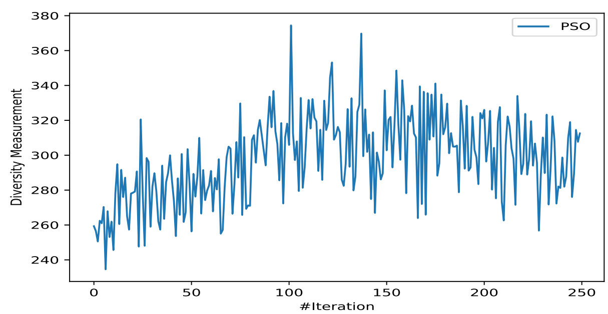


FIGURE 17. **PSO diversity.**

Figs. 19 to 22 illustrate the exploration and exploitation curves of GA, FPA, PSO, and ABC, respectively. They allow us to better understand the behavior of their populations as they progress over iterations. These plots indicate that the GA exploration curve drops from its greatest value, which corresponds to the maximum value of diversity, to a relatively low value, this is because GA combines the current population with newly created offspring and keeps the top Pop_{size}

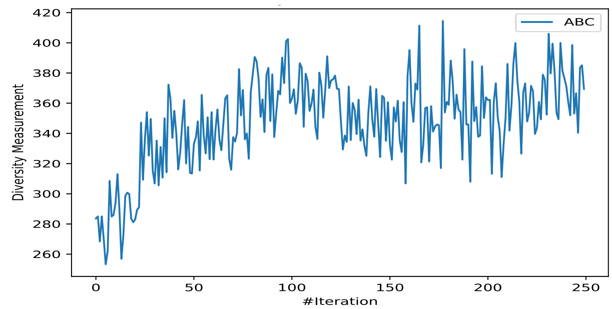


FIGURE 18. ABC diversity.

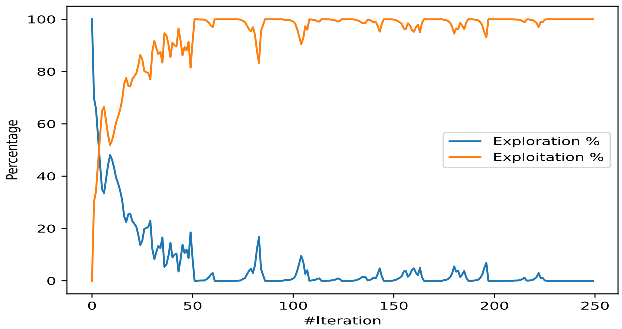


FIGURE 19. GA exploration and exploitation.

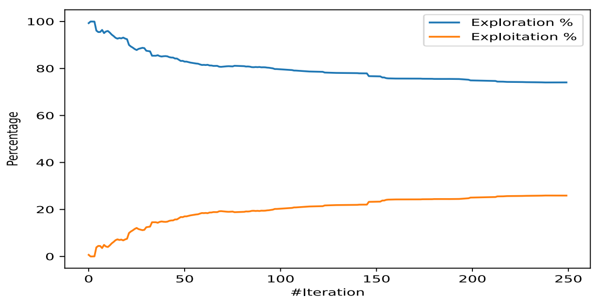


FIGURE 20. FPA exploration and exploitation.

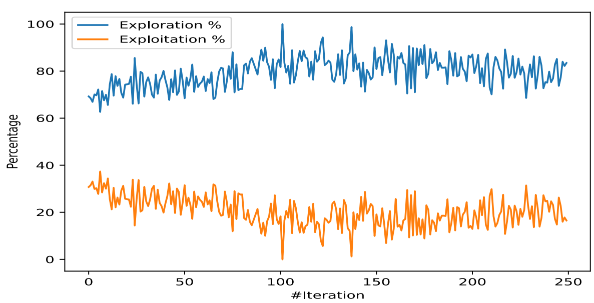


FIGURE 21. PSO exploration and exploitation.

individuals for the succeeding population. As a result, after a given number of iterations, we observe that the individuals start having close coverage rate values resulting in a relatively low diversity value, which explains why the GA exploitation

TABLE 8. comparison between GA, PSO, FPA, and ABC.

	Coverage rate	Execution time	Convergence	Diversity
GA	82.82%	Fast	Medium	Medium
PSO	75.10%	Medium	Fast	High
FPA	74.52%	Medium	Fast	Low
ABC	75.93%	Slow	Fast	High

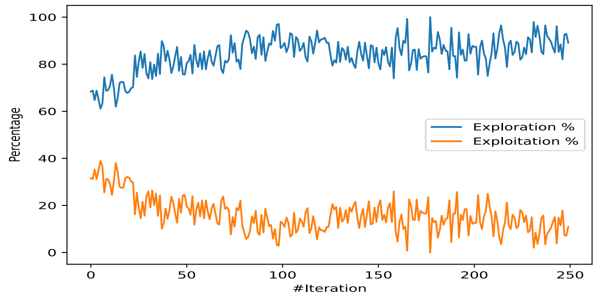


FIGURE 22. ABC exploration and exploitation.

curve is above the exploration curve. In contrast to GA, all FPA, PSO, and ABC maintain the exploration curve above the exploitation curve throughout iterations. This is due to the fact that the three algorithms do not retain full individuals from the prior population in the current one; instead, they employ other criteria to make new offspring. Hence, these latter do not have necessarily a close fitness value (coverage rate).

Based on the previous results, we deduce that

- GA outperforms ABC, PSO, and FPA in terms of coverage rate and execution time (see Table 8).
- FPA, PSO, and ABC lack mechanisms to avoid local optima.
- It is crucial to find a balance between the exploration and exploitation features in constructing a new population and to control the population diversity in order to improve the quality of the solutions.
- Considering the hybridization of metaheuristics could significantly improve the search in the search space and balance the exploration and exploitation features.

In the second part of these simulation tests, we aimed at analyzing the performance of NSGA-II and MOEA/D since they are both widely applied when dealing with multiple objectives in WSNs deployment. For that, we considered two objective functions to optimize: coverage rate and deployment cost, which refers to the number of deployed sensor nodes. The area of interest is a 2D plane without obstacles with a dimension of $100m \times 100m$.

Fig. 23 illustrates the Pareto fronts of three NSGA-II and MOEA/D executions. As we can see, the Pareto front of NSGA-II is above the Pareto front of MOEA/D; this means that the NSGA-II solutions completely dominate solutions proposed by MOEA/D. Moreover, the NSGA-II Pareto front proposes various solutions compared to MOEA/D, enabling the decision-maker to consider all possible scenarios and select the optimal compromise. Further, the NSGA-II

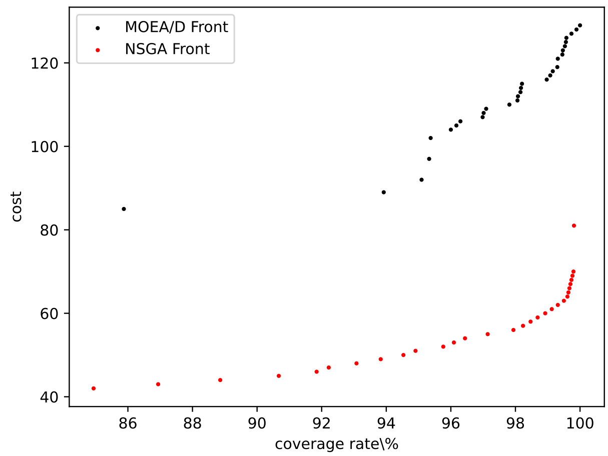


FIGURE 23. NSGA-II Pareto front vs MOEA/D Pareto front.

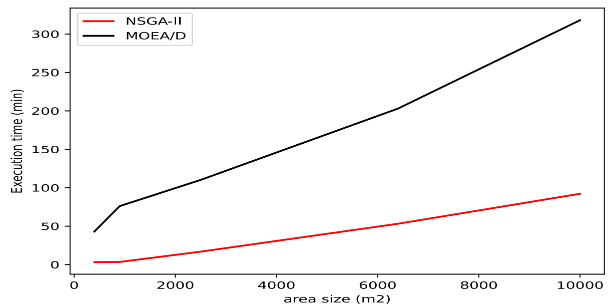


FIGURE 24. Scalability of NSGA-II and MOEA/D.

solutions have a coverage rate varying from 84% to 100% and deployment cost varying from 40 to 80 while the Pareto front of MOEA/D comprises one solution with a coverage rate of less than 94%, and the rest of the solutions have a coverage rate varying from 94% to 100% with deployment cost values upwards of 85. As a second test, we compared the NSGA-II and MOEA/D scalability by increasing the size of the target region and measuring the execution time in minutes for each example (results are shown in Fig.24). As we can observe, the NSGA-II takes substantially less execution time than the MOEA/D in all scenarios. Therefore, we can conclude that NSGA-II outperforms MOEA/D in both solutions quality and scalability.

VII. CHALLENGES AND OPEN ISSUES

Although the huge effort done by the research community to solve the problem of WSNs deployment, many open issues and challenges are still existing. Indeed, the nodes deployment problem has to be meticulously considered by including realistic parameters, such as the type of sensors (contact/noncontact, directional/ omnidirectional, the sensing technology), the sensors heterogeneity, the type of area of interest (indoor/outdoor, 2D/3D), and the impact of obstacles on the sensor network, in order to reach an optimal and a feasible deployment scheme.

A. SENSING RANGE MODELING

Coverage is the most crucial and critical factor in designing WSNs deployment solutions. Thus, it is essential to accurately assess it in order to reach reliable results and avoid any discrepancies between simulations and real-world network performance. As mentioned in section III-B, the coverage function depends mostly on the sensing models, however neither the deterministic nor the probabilistic models are rigorous enough to simulate the real sensing zone of a particular sensor type. This is due to the fact that each sensor type employs a sensing technology that can be based on one or more physical effects, including inductance, capacitance, magnetism, and electromagnetism; And because each physical effect might be disturbed differently by distance and other external factors. The probability of detection will also change in response to these effects. For that, future contributions for accurate modeling of sensing zone are highly encouraged, the proposed sensing models must consider the sensing technology as well as the distortion of the sensing zone caused by the existing obstacles.

B. REAL TARGET AREA MODELING

Most of the proposed solutions for the WSN deployment problem assume a rectangular 2D and free-obstacle area of interest without specifying the type of environment(indoor or outdoor). In fact, these very strong hypotheses are rarely met in reality since each target area has its specific form and dispersed obstructions, and each obstruction has a distinct impact on the sensing and communication zone of a sensor node. Furthermore, the height of obstacles is an important parameter that must be considered in defining the relevant positions where sensors could be placed. Yet, this information is not available when dealing with a 2D plane, consequently, the final result is incomplete and does not provide a full 3D location of each sensor. Moreover, most of the works dealing with a 3D environment with obstacles have adopted the line of sight technique to evaluate the impact of obstacles on the network coverage and connectivity [191], [192]. However, this technique is still not rigorous as it considers all obstacles as opaque objects.

C. SENSOR NETWORK HETEROGENEITY

A WSN may contain sensor nodes with various physical characteristics (ex: batteries) and embedded technologies (sensing and communication technologies). This heterogeneity must be considered while designing the deployment scheme by choosing the appropriate models to calculate the fitness function. The few articles that address the problem of heterogeneous WSN deployment consider sensors with different sensing ranges without specifying the technologies or the protocols used. It is therefore important that future contributions take into account all the factors influencing network heterogeneity, and design accurate models to simulate the real behavior of sensor nodes.

D. ENVIRONMENT DYNAMICITY

It is very probable that the target area will change over time. For example, in the case of buildings, separators can be updated or removed, and others of different forms and materials could be added. Thus, considering the environment dynamicity while designing the deployment scheme must be further explored to adjust current deployment and maintain the network connectivity and the optimal coverage rate. Therefore, we believe that using an appropriate data source to model the area of interest, whether it is outdoor or indoor, is highly recommended in order to provide the necessary information needed by the deployment algorithm to adjust the optimal solution.

E. COMPUTATIONAL TIME AND SEARCH SPACE COMPLEXITY OF METAHEURISTICS-BASED SOLUTIONS

The problem of WSN deployment has been proven to be an NP-hard combinatorial optimization problem [12]. The complexity of the problem depends on several factors including, the size and the type of the target area, the number of sensor nodes in the network, the network heterogeneity, and the number of objective functions to optimize. Most research efforts have applied metaheuristics to solve the problem, however, this category of algorithms is also computationally consuming, particularly when the search space expands. Therefore, reducing the complexity of the research space is a key step toward improving the computational efficiency of the solution. This can be done using a reliable data source describing the target area to exclude unsuitable deployment points and yet, minimize the number of sensors locations to test. Moreover, designing sophisticated metaheuristic operators to guide the search and balance the exploration and exploitation aspects will contribute to attaining better results in less time.

VIII. CONCLUSION

In this paper, we presented a general reference optimization model and surveyed AI-based approaches used in literature to optimize WSNs deployment. The general reference model details the decision variables and the feasibility constraints, as well as the mathematical modeling of the most salient objective function: coverage function, network lifetime, energy consumption, and cost function. Furthermore, it emphasizes the importance of environment modeling in the design of the deployment scheme and recapitulates the commonly used target area modeling reported in the literature. A variety of deployment solutions utilizing AI-based techniques, including evolutionary algorithms, swarm intelligence optimization algorithms, hybrid metaheuristics, fuzzy logic, and machine learning are reviewed and analyzed in this survey. In order to highlight current research directions in the field of WSN deployment, we presented some statistical information based on the studied research works. In addition, simulation experiments were conducted to evaluate the performance of commonly used algorithms for tackling the deployment problem of WSNs. They demonstrate that GA can achieve a higher coverage rate in a shorter execution time compared to PSO, ABC, and FPA, which all stagnated in the early stage. For multi-objective algorithms, simulation tests reveal that NSGA-II can provide a good trade-off between coverage rate and deployment cost and has better scalability compared with MOEA/D.

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Comput. Netw.*, vol. 38, no. 4, pp. 393–422, 2002.
[2] W. Y. Du, *Resistive, Capacitive, Inductive, and Magnetic Sensor Technologies*. Boca Raton, FL, USA: CRC Press, 2014.
[3] R. Elhabyan, W. Shi, and M. St-Hilaire, "Coverage protocols for wireless sensor networks: Review and future directions," *J. Commun. Netw.*, vol. 21, no. 1, pp. 45–60, Feb. 2019.
[4] S. Abdollahzadeh and N. J. Navimipour, "Deployment strategies in the wireless sensor network: A comprehensive review," *Comput. Commun.*, vols. 91–92, pp. 1–16, Oct. 2016.
[5] A. Srivastava and P. K. Mishra, "A survey on WSN issues with its heuristics and meta-heuristics solutions," *Wireless Pers. Commun.*, vol. 121, no. 1, pp. 745–814, Nov. 2021.
[6] 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, 1st Quart., 2017.
[7] A. Tripathi, H. P. Gupta, T. Dutta, R. Mishra, K. K. Shukla, and S. Jit, "Coverage and Connectivity in WSNs: A survey, research issues and challenges," *IEEE Access*, vol. 6, pp. 26971–26992, 2018.
[8] C. W. Tsai, P. W. Tsai, J. S. Pan, and H. C. Chao, "Metaheuristics for the deployment problem of WSN: A review," *Microprocess. Microsyst.*, vol. 39, no. 8, pp. 1305–1317, 2015.
[9] F. Aznoli and N. J. Navimipour, "Deployment strategies in the wireless sensor networks: Systematic literature review, classification, and current trends," *Wireless Pers. Commun.*, vol. 95, no. 2, pp. 819–846, 2017.
[10] A. Singh, S. Sharma, and J. Singh, "Nature-inspired algorithms for wireless sensor networks: A comprehensive survey," *Comput. Sci. Rev.*, vol. 39, Feb. 2021, Art. no. 100342.
[11] C. Zhu, C. Zheng, L. Shu, and G. Han, "A survey on coverage and connectivity issues in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 35, no. 2, pp. 619–632, Mar. 2012.
[12] D. S. Deif and Y. Gadallah, "Classification of wireless sensor networks deployment techniques," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 2, pp. 834–855, 2nd Quart., 2014.
[13] R. Priyadarshi, B. Gupta, and A. Anurag, "Deployment techniques in wireless sensor networks: A survey, classification, challenges, and future research issues," *J. Supercomput.*, vol. 76, pp. 7333–7373, Jan. 2020.
[14] 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. Netw.*, vol. 99, pp. 134–161, Apr. 2016.
[15] A. Hossain, S. Chakrabarti, and P. K. Biswas, "Impact of sensing model on wireless sensor network coverage," *IET Wireless Sensor Syst.*, vol. 2, no. 3, pp. 272–281, Sep. 2012.
[16] A. Shan, X. Xu, and Z. Cheng, "Target coverage in wireless sensor networks with probabilistic sensors," *Sensors*, vol. 16, no. 9, p. 1372, Aug. 2016.
[17] E. Amaldi, A. Capone, M. Cesana, and I. Filippini, "Design of wireless sensor networks for mobile target detection," *IEEE/ACM Trans. Netw.*, vol. 20, no. 3, pp. 784–797, Jun. 2012.
[18] M. Chaudhary and A. K. Pujari, "Q-coverage problem in wireless sensor networks," in *Proc. Int. Conf. Distrib. Comput. Netw.* Springer, 2009, pp. 325–330.

[19] M. Li, W. Cheng, K. Liu, Y. He, X. Li, and X. Liao, "Sweep coverage with mobile sensors," *IEEE Trans. Mobile Comput.*, vol. 10, no. 11, pp. 1534–1545, Nov. 2011.

[20] P. Huang, L. Xu, Z. Kang, C. Liu, and F. Lin, "GA-based sweep coverage scheme in WSN," in *Proc. 7th Int. Conf. Cloud Comput. Big Data (CCBD)*, Nov. 2016, pp. 254–259.

[21] Z. Nie and H. Du, "An approximation algorithm for general energy restricted sweep coverage problem," *Theor. Comput. Sci.*, vol. 864, pp. 70–79, Apr. 2021.

[22] B. Liu, O. Dousse, J. Wang, and A. Saipulla, "Strong barrier coverage of wireless sensor networks," in *Proc. 9th ACM Int. Symp. Mobile ad hoc Netw. Comput. (MobiHoc)*, 2008, pp. 411–420.

[23] J. Amutha, S. Sharma, and J. Nagar, "WSN strategies based on sensors, deployment, sensing models, coverage and energy efficiency: Review, approaches and open issues," *Wireless Pers. Commun.*, vol. 111, no. 2, pp. 1089–1115, Mar. 2020.

[24] J. Liang, M. Liu, and X. Kui, "A survey of coverage problems in wireless sensor networks," *Sensors Transducers*, vol. 163, no. 1, pp. 240–246, 2014.

[25] Y. Wang, S. Wu, Z. Chen, X. Gao, and G. Chen, "Coverage problem with uncertain properties in wireless sensor networks: A survey," *Comput. Netw.*, vol. 123, pp. 200–232, Aug. 2017.

[26] S. Kundu, N. Das, and D. Saha, "A realistic sensing model for event area estimation in wireless sensor networks," in *Progress in Advanced Computing and Intelligent Engineering*. Springer, 2021, pp. 244–256.

[27] S. Kumar and D. Lobiyal, "Sensing coverage prediction for wireless sensor networks in shadowed and multipath environment," *Sci. World J.*, vol. 2013, Oct. 2013, Art. no. 565419.

[28] J. Vidhya, P. Prasanna, M. Margarat, and S. Jayalakshmy, "Enhancing network coverage using sensing models in wireless sensor network," in *Proc. J. Phys., Conf.*, vol. 1717, 2021, Art. no. 012062.

[29] V. Akbarzadeh, C. Gagné, M. Parizeau, M. Argany, and M. A. Mostafavi, "Probabilistic sensing model for sensor placement optimization based on line-of-sight coverage," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 2, pp. 293–303, Feb. 2012.

[30] B. Wang, "Coverage problems in sensor networks: A survey," *ACM Comput. Surv.*, vol. 43, no. 4, pp. 1–53, 2011.

[31] H. T. T. Binh, N. T. M. Binh, N. H. Ngoc, D. T. H. Ly, and N. D. Nghia, "Efficient approximation approaches to minimal exposure path problem in probabilistic coverage model for wireless sensor networks," *Appl. Soft Comput.*, vol. 76, pp. 726–743, Mar. 2019.

[32] A. Hossain, P. K. Biswas, and S. Chakrabarti, "Sensing models and its impact on network coverage in wireless sensor network," in *Proc. IEEE Region 10 3rd Int. Conf. Ind. Inf. Syst.*, Dec. 2008, pp. 1–5.

[33] N. Rai and R. Daruwala, "Node density optimisation using composite probabilistic sensing model in wireless sensor networks," *IET Wireless Sensor Syst.*, vol. 9, no. 4, pp. 181–190, Aug. 2019.

[34] R. Soua and P. Minet, "A survey on energy efficient techniques in wireless sensor networks," in *Proc. 4th Joint IFIP Wireless Mobile Netw. Conf. (WMNC)*, Oct. 2011, pp. 1–9.

[35] I. Ben Arbi, F. Derbel, and F. Strakosch, "Forecasting methods to reduce energy consumption in WSN," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (I2MTC)*, May 2017, pp. 1–6.

[36] P. Maheshwari, A. K. Sharma, and K. Verma, "Energy efficient cluster based routing protocol for WSN using butterfly optimization algorithm and ant colony optimization," *Ad Hoc Netw.*, vol. 110, Jan. 2021, Art. no. 102317.

[37] R. Bhardwaj and D. Kumar, "MOFPL: Multi-objective fractional particle lion algorithm for the energy aware routing in the WSN," *Pervas. Mobile Comput.*, vol. 58, Aug. 2019, Art. no. 101029.

[38] A. Vinitha and M. S. S. Rukmini, "Secure and energy aware multi-hop routing protocol in WSN using Taylor-based hybrid optimization algorithm," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 5, pp. 1857–1868, May 2022.

[39] M. Jeske, V. Rosset, and M. C. V. Nascimento, "Determining the trade-offs between data delivery and energy consumption in large-scale WSNs by multi-objective evolutionary optimization," *Comput. Netw.*, vol. 179, Oct. 2020, Art. no. 107347.

[40] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Netw.*, vol. 7, no. 3, pp. 537–568, May 2009.

[41] R. Rajagopalan, C. K. Mohan, P. Varshney, and K. Mehrotra, "Multi-objective mobile agent routing in wireless sensor networks," in *Proc. IEEE Congr. Evol. Comput.*, vol. 2, Sep. 2005, pp. 1730–1737.

[42] G. Sati and S. Singh, "A review on outdoor propagation models in radio communication," *Int. J. Comput. Eng. Sci.*, vol. 4, no. 2, pp. 64–68, 2014.

[43] Z. Naseem, I. Nausheen, and Z. Mirza, "Propagation models for wireless communication system," *Int. J. Res. Eng. Technol.*, vol. 5, no. 1, pp. 237–242, 2018.

[44] M. Luo, "Indoor radio propagation modeling for system performance prediction," Ph.D. thesis, INSA de Lyon, Villeurbanne, France, 2013.

[45] J. Milanović, S. Rimac-Drlje, and I. Majerski, "Radio wave propagation mechanisms and empirical models for fixed wireless access systems," *Tehnički Vjesnik*, vol. 17, no. 1, pp. 43–53, 2010.

[46] A. Saad, M. R. Senouci, and O. Benyattou, "Toward a realistic approach for the deployment of 3D wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 21, no. 4, pp. 1508–1519, Apr. 2022.

[47] B. Cao, J. Zhao, Z. Lv, X. Liu, X. Kang, and S. Yang, "Deployment optimization for 3D industrial wireless sensor networks based on particle swarm optimizers with distributed parallelism," *J. Netw. Comput. Appl.*, vol. 103, pp. 225–238, Feb. 2018.

[48] B. Cao, J. Zhao, Z. Lv, and X. Liu, "3D terrain multiobjective deployment optimization of heterogeneous directional sensor networks in security monitoring," *IEEE Trans. Big Data*, vol. 5, no. 4, pp. 495–505, Dec. 2019.

[49] X. Yu, J. Zhang, J. Fan, and T. Zhang, "A faster convergence artificial bee colony algorithm in sensor deployment for wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 9, no. 10, Oct. 2013, Art. no. 497264.

[50] M. T. Kouakou, K. Yasumoto, S. Yamamoto, and M. Ito, "Cost-efficient sensor deployment in indoor space with obstacles," in *Proc. IEEE Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2012, pp. 1–9.

[51] Z. Wang and H. Xie, "Wireless sensor network deployment of 3D surface based on enhanced grey wolf optimizer," *IEEE Access*, vol. 8, pp. 57229–57251, 2020.

[52] S. E. Bouzid, Y. Seresstou, K. Raoof, M. N. Omri, M. Mbarki, and C. Dridi, "MOONGA: Multi-objective optimization of wireless network approach based on genetic algorithm," *IEEE Access*, vol. 8, pp. 105793–105814, 2020.

[53] S. E. Bouzid, Y. Serrestou, K. Raoof, M. Mbarki, M. N. Omri, and C. Dridi, "Wireless sensor network deployment optimisation based on coverage, connectivity and cost metrics," *Int. J. Sensor Netw.*, vol. 33, no. 4, pp. 224–238, 2020.

[54] M. A. Benatia, M. Sahnoun, D. Baudry, A. Louis, A. El-Hami, and B. Mazari, "Multi-objective WSN deployment using genetic algorithms under cost, coverage, and connectivity constraints," *Wireless Pers. Commun.*, vol. 94, no. 4, pp. 2739–2768, Jun. 2017.

[55] K. Zaimen, M.-E.-A. Brahmia, J.-F. Dollinger, L. Moalic, A. Abouaissa, and L. Idoumghar, "A overview on WSN deployment and a novel conceptual BIM-based approach in smart buildings," in *Proc. 7th Int. Conf. Internet Things: Syst., Manage. Secur. (IOTSMS)*, Dec. 2020, pp. 1–6.

[56] W. Fu, Y. Yang, G. Hong, and J. Hou, "WSN deployment strategy for real 3D terrain coverage based on greedy algorithm with DEM probability coverage model," *Electronics*, vol. 10, no. 16, p. 2028, Aug. 2021.

[57] N. T. Tam, V. D. Hoang, and H. T. T. Binh, "Multi-objective teaching–learning evolutionary algorithm for enhancing sensor network coverage and lifetime," *Eng. Appl. Artif. Intell.*, vol. 108, Feb. 2022, Art. no. 104554.

[58] M. Argany, F. Karimipour, F. Mafi, and A. Afghantoloee, "Optimization of wireless sensor networks deployment based on probabilistic sensing models in a complex environment," *J. Sensor Actuator Netw.*, vol. 7, no. 2, p. 20, May 2018.

[59] K. Zaimen, M.-E.-A. Brahmia, J.-F. Dollinger, L. Moalic, A. Abouaissa, and L. Idoumghar, "Coverage maximization in WSN deployment using particle swarm optimization with Voronoi diagram," in *Proc. Int. Conf. Model Data Eng.* Springer, 2021, pp. 88–100.

[60] A. Metiaf and Q. Wu, "Particle swarm optimization based deployment for WSN with the existence of obstacles," in *Proc. 5th Int. Conf. Control, Autom. Robot. (ICCAR)*, Apr. 2019, pp. 614–618.

[61] M. Abo-Zahhad, N. Sabor, S. Sasaki, and S. M. Ahmed, "A centralized immune-Voronoi deployment algorithm for coverage maximization and energy conservation in mobile wireless sensor networks," *Inf. Fusion*, vol. 30, pp. 36–51, Jul. 2016.

[62] N. Nasri, S. Mnasri, and T. Val, "3D node deployment strategies prediction in wireless sensors network," *Int. J. Electron.*, vol. 107, no. 5, pp. 808–838, May 2020.

[63] U. Mansoor and H. M. Ammari, "Coverage and connectivity in 3D wireless sensor networks," in *The Art of Wireless Sensor Networks*. Springer, 2014, pp. 273–324.

[64] C. K. Ng, C. H. Wu, W. H. Ip, and K. L. Yung, "A smart bat algorithm for wireless sensor network deployment in 3-D environment," *IEEE Commun. Lett.*, vol. 22, no. 10, pp. 2120–2123, Oct. 2018.

[65] H. A. Hashim, B. O. Ayinde, and M. A. Abido, "Optimal placement of relay nodes in wireless sensor network using artificial bee colony algorithm," *J. Netw. Comput. Appl.*, vol. 64, pp. 239–248, Apr. 2016.

[66] D. Sharma and V. Gupta, "Modeling 3D WSN to maximize coverage using harmony search scheme," in *Advanced Engineering Optimization Through Intelligent Techniques*. Springer, 2020, pp. 95–111.

[67] V. Akbarzadeh, J.-C. Lévesque, C. Gagné, and M. Parizeau, "Efficient sensor placement optimization using gradient descent and probabilistic coverage," *Sensors*, vol. 14, no. 8, pp. 15525–15552, Aug. 2014.

[68] K. Murota, "Linear programming," in *Computer Vision: A Reference Guide*. Springer, 2020, pp. 1–7.

[69] J. Rust, "Dynamic programming," in *The New Palgrave Dictionary of Economics*, vol. 1. 2008, p. 8.

[70] D. R. Morrison, S. H. Jacobson, J. J. Sauppe, and E. C. Sewell, "Branch-and-bound algorithms: A survey of recent advances in searching, branching, and pruning," *Discrete Optim.*, vol. 19, pp. 79–102, Feb. 2016.

[71] L. Jourdan, M. Basseur, and E.-G. Talbi, "Hybridizing exact methods and metaheuristics: A taxonomy," *Eur. J. Oper. Res.*, vol. 199, no. 3, pp. 620–629, Dec. 2009.

[72] M. Abdel-Basset, L. Abdel-Fatah, and A. K. Sangaiah, "Metaheuristic algorithms: A comprehensive review," in *Computational Intelligence for Multimedia Big Data on the Cloud With Engineering Applications*. 2018, pp. 185–231.

[73] M. A. Lones, "Mitigating metaphors: A comprehensible guide to recent nature-inspired algorithms," *Social Netw. Comput. Sci.*, vol. 1, no. 1, pp. 1–12, Jan. 2020.

[74] K. Sörensen, "Metaheuristics—The metaphor exposed," *Int. Trans. Oper. Res.*, vol. 22, no. 1, pp. 3–18, Jan. 2015.

[75] A. J. Al-Mousawi, "Evolutionary intelligence in wireless sensor network: Routing, clustering, localization and coverage," *Wireless Netw.*, vol. 26, no. 8, pp. 5595–5621, Nov. 2020.

[76] A. Slowik and H. Kwasnicka, "Evolutionary algorithms and their applications to engineering problems," *Neural Comput. Appl.*, vol. 32, no. 16, pp. 12363–12379, Aug. 2020.

[77] K. Li, Z. Wen, and Z. Wang, "Optimised placement of wireless sensor networks by evolutionary algorithm," *Int. J. Comput. Sci. Eng.*, vol. 15, nos. 1–2, pp. 74–86, Jan. 2017.

[78] H. T. T. Binh, N. T. Hanh, L. Van Quan, N. D. Nghia, and N. Dey, "Metaheuristics for maximization of obstacles constrained area coverage in heterogeneous wireless sensor networks," *Appl. Soft Comput.*, vol. 86, Jan. 2020, Art. no. 105939.

[79] D. Zhang and J. Zhang, "Multi-species evolutionary algorithm for wireless visual sensor networks coverage optimization with changeable field of views," *Appl. Soft Comput.*, vol. 96, Nov. 2020, Art. no. 106680.

[80] W. M. Dahmane, M.-E.-A. Brahmia, J.-F. Dollinger, and S. Ouchani, "A BIM-based framework for an optimal WSN deployment in smart building," in *Proc. 11th Int. Conf. Netw. Future (NoF)*, Oct. 2020, pp. 110–114.

[81] M. Khalesian and M. R. Delavar, "Wireless sensors deployment optimization using a constrained Pareto-based multi-objective evolutionary approach," *Eng. Appl. Artif. Intell.*, vol. 53, pp. 126–139, Aug. 2016.

[82] A. Ndam Njoya, W. Abdou, A. Dipanda, and E. Tonye, "Optimization of sensor deployment using multi-objective evolutionary algorithms," *J. Reliable Intell. Environments*, vol. 2, no. 4, pp. 209–220, Dec. 2016.

[83] F. Domingo-Perez, J. Lazaro-Galilea, I. Bravo, A. Gardel, and D. Rodriguez, "Optimization of the coverage and accuracy of an indoor positioning system with a variable number of sensors," *Sensors*, vol. 16, no. 6, p. 934, Jun. 2016.

[84] T. Xiang, H. Wang, and Y. Shi, "Hybrid WSN node deployment optimization strategy based on CS algorithm," in *Proc. IEEE 3rd Inf. Technol., Netw., Electron. Autom. Control Conf. (ITNEC)*, Mar. 2019, pp. 621–625.

[85] S. B. Chaudhry, V. C. Hung, R. K. Guha, and K. O. Stanley, "Pareto-based evolutionary computational approach for wireless sensor placement," *Eng. Appl. Artif. Intell.*, vol. 24, no. 3, pp. 409–425, Apr. 2011.

[86] S. Singh, A. S. Nandan, G. Sikka, A. Malik, and P. K. Singh, "Genetic algorithm-based data controlling method using IoT-enabled WSN in power grid," *Soft Comput.*, to be published.

[87] H. T. T. Binh, N. T. Hanh, L. Van Quan, and N. Dey, "Improved cuckoo search and chaotic flower pollination optimization algorithm for maximizing area coverage in wireless sensor networks," *Neural Comput. Appl.*, vol. 30, no. 7, pp. 2305–2317, 2018.

[88] D. Arivudainambi, R. Pavithra, and P. Kalyani, "Cuckoo search algorithm for target coverage and sensor scheduling with adjustable sensing range in wireless sensor network," *J. Discrete Math. Sci. Cryptogr.*, vol. 24, no. 4, pp. 975–996, 2020.

[89] E. O. Rangel, D. G. Costa, and A. Loula, "On redundant coverage maximization in wireless visual sensor networks: Evolutionary algorithms for multi-objective optimization," *Appl. Soft Comput.*, vol. 82, Sep. 2019, Art. no. 105578.

[90] J. R. Srivastava and T. S. B. Sudarshan, "Energy-efficient cache node placement using genetic algorithm in wireless sensor networks," *Soft Comput.*, vol. 19, no. 11, pp. 3145–3158, Nov. 2015.

[91] R. Kosar and C. Ersoy, "Sink placement on a 3D terrain for border surveillance in wireless sensor networks," *Eng. Appl. Artif. Intell.*, vol. 25, no. 1, pp. 82–93, Feb. 2012.

[92] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2000.

[93] J. M. Lanza-Gutierrez and J. A. Gomez-Pulido, "Studying the multiobjective variable neighbourhood search algorithm when solving the relay node placement problem in wireless sensor networks," *Soft Comput.*, vol. 20, no. 1, pp. 67–86, Jan. 2016.

[94] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[95] K. Wei, "Node deployment for wireless sensor networks based on improved multi-objective evolutionary algorithm," *Int. J. Internet Protocol Technol.*, vol. 10, no. 3, pp. 189–195, 2017.

[96] S. Sengupta, S. Das, M. D. Nasir, and B. K. Panigrahi, "Multi-objective node deployment in WSNs: In search of an optimal trade-off among coverage, lifetime, energy consumption, and connectivity," *Eng. Appl. Artif. Intell.*, vol. 26, no. 1, pp. 405–416, 2013.

[97] A. Konstantinidis, K. Yang, Q. Zhang, and D. Zeinalipour-Yazti, "A multi-objective evolutionary algorithm for the deployment and power assignment problem in wireless sensor networks," *Comput. Netw.*, vol. 54, no. 6, pp. 960–976, Apr. 2010.

[98] A. Konstantinidis and K. Yang, "Multi-objective K-connected deployment and power assignment in WSNs using a problem-specific constrained evolutionary algorithm based on decomposition," *Comput. Commun.*, vol. 34, no. 1, pp. 83–98, Jan. 2011.

[99] J. M. Lanza-Gutierrez and J. A. Gomez-Pulido, "Assuming multiobjective metaheuristics to solve a three-objective optimisation problem for relay node deployment in wireless sensor networks," *Appl. Soft Comput.*, vol. 30, pp. 675–687, May 2015.

[100] B. A. Attea, M. N. Abbas, M. Al-Ani, and S. Özdemir, "Bio-inspired multi-objective algorithms for connected set K-covers problem in wireless sensor networks," *Soft Comput.*, vol. 23, no. 22, pp. 11699–11728, Nov. 2019.

[101] X.-S. Yang and S. Deb, "Cuckoo search via Lévy flights," in *Proc. World Congr. Nature Biologically Inspired Comput. (NaBIC)*, Dec. 2009, pp. 210–214.

[102] A. Chakraborty and A. K. Kar, "Swarm intelligence: A review of algorithms," in *Nature-Inspired Computing and Optimization*. 2017, pp. 475–494.

[103] M. N. Ab Wahab, S. Nefti-Meziani, and A. Atyabi, "A comprehensive review of swarm optimization algorithms," *PLoS ONE*, vol. 10, no. 5, May 2015, Art. no. e0122827.

[104] L. Brezočnik, I. Fister, and V. Podgorelec, "Swarm intelligence algorithms for feature selection: A review," *Appl. Sci.*, vol. 8, no. 9, p. 1521, Sep. 2018.

[105] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw. (ICNN)*, vol. 4, Nov./Dec. 1995, pp. 1942–1948.

[106] Y. Du, "Method for the optimal sensor deployment of WSNs in 3D terrain based on the DPSOVF algorithm," *IEEE Access*, vol. 8, pp. 140806–140821, 2020.

[107] M. Shakeri, A. Sadeghi-Niaraki, S.-M. Choi, and S. M. R. Islam, "Performance analysis of IoT-based health and environment WSN deployment," *Sensors*, vol. 20, no. 20, p. 5923, Oct. 2020.

[108] W. Qi, H. Yu, G. Fan, L. Chen, and X. Wen, "WSN coverage optimization based on two-stage PSO," in *Proc. Int. Conf. Collaborative Comput., Netw., Appl. Worksharing*. Springer, 2020, pp. 19–35.

[109] Y. Li and J. Cao, "WSN node optimal deployment algorithm based on adaptive binary particle swarm optimization," *ASP Trans. Internet Things*, vol. 1, no. 1, pp. 1–8, May 2021.

[110] R. Yarinezhad and S. N. Hashemi, "A sensor deployment approach for target coverage problem in wireless sensor networks," *J. Ambient Intell. Humanized Comput.*, to be published.
[111] J.-L. Xie, C.-R. Li, and G.-Y. Xu, "Optimization and energy-saving deployment of nodes in wireless sensor network," *J. Comput.*, vol. 31, no. 3, pp. 142–153, 2020.
[112] 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," *Natural Comput.*, vol. 16, no. 1, pp. 5–13, Mar. 2017.
[113] P. M. Pradhan and G. Panda, "Connectivity constrained wireless sensor deployment using multiobjective evolutionary algorithms and fuzzy decision making," *Ad Hoc Netw.*, vol. 10, no. 6, pp. 1134–1145, Aug. 2012.
[114] S. Mnasri, N. Nasri, A. van den Bossche, and T. Val, "A new multi-agent particle swarm algorithm based on birds accents for the 3D indoor deployment problem," *ISA Trans.*, vol. 91, pp. 262–280, Aug. 2019.
[115] A. M. Jubair, R. Hassan, A. H. M. Aman, and H. Sallehudin, "Social class particle swarm optimization for variable-length wireless sensor network deployment," *Appl. Soft Comput.*, vol. 113, Dec. 2021, Art. no. 107926.
[116] M. Dorigo and G. D. Caro, "Ant colony optimization: A new meta-heuristic," in *Proc. Congr. Evol. Comput.*, vol. 2, Jul. 1999, pp. 1470–1477.
[117] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, Nov. 2006.
[118] M. A. Adnan, M. A. Razzaque, I. Ahmed, and I. F. Isnin, "Bio-mimic optimization strategies in wireless sensor networks: A survey," *Sensors*, vol. 14, no. 1, pp. 299–345, 2014.
[119] D. S. Deif and Y. Gadallah, "An ant colony optimization approach for the deployment of reliable wireless sensor networks," *IEEE Access*, vol. 5, pp. 10744–10756, 2017.
[120] X. Sun, Y. Zhang, X. Ren, and K. Chen, "Optimization deployment of wireless sensor networks based on culture–ant colony algorithm," *Appl. Math. Comput.*, vol. 250, pp. 58–70, Jan. 2015.
[121] X. Liu and D. He, "Ant colony optimization with greedy migration mechanism for node deployment in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 39, pp. 310–318, Mar. 2014.
[122] T. Qasim, M. Zia, Q.-A. Minhas, N. Bhatti, K. Saleem, T. Qasim, and H. Mahmood, "An ant colony optimization based approach for minimum cost coverage on 3-D grid in wireless sensor networks," *IEEE Commun. Lett.*, vol. 22, no. 6, pp. 1140–1143, Jun. 2018.
[123] T. Qasim, A. Mujahid, N. A. Bhatti, and M. Mushtaq, "ACO-Discreet: An efficient node deployment approach in wireless sensor networks," in *Information Technology—New Generations*. Springer, 2018, pp. 43–48.
[124] H. Su, G. Wang, X. Sun, Z. Zhang, and Y. Zhou, "Optimal node deployment strategy for wireless sensor networks based on dynamic ant colony algorithm," *Int. J. Embedded Syst.*, vol. 8, nos. 2–3, pp. 258–265, 2016.
[125] S. Mirjalili, "The ant lion optimizer," *Adv. Eng. Softw.*, vol. 83, pp. 80–98, May 2015.
[126] W. Liu, S. Yang, S. Sun, and S. Wei, "A node deployment optimization method of WSN based on ant-lion optimization algorithm," in *Proc. IEEE 4th Int. Symp. Wireless Syst. Int. Conf. Intell. Data Acquisition Adv. Comput. Syst. (IDAACS-SWS)*, Sep. 2018, pp. 88–92.
[127] H. Deghbouch and F. Debbat, "Dynamic deployment of wireless sensor networks using hybrid ant lion optimizer and Tabu search," in *Proc. 4th Int. Symp. Informat. Appl. (ISIA)*, Dec. 2020, pp. 1–5.
[128] M. A. Syed, M. Md, and R. Syed, "Optimal sensor deployment using ant lion optimization," in *Advances in Decision Sciences, Image Processing, Security and Computer Vision*. Springer, 2020, pp. 460–469.
[129] Y. Yao, Y. Li, D. Xie, S. Hu, C. Wang, and Y. Li, "Coverage enhancement strategy for WSNs based on virtual force-directed ant lion optimization algorithm," *IEEE Sensors J.*, vol. 21, no. 17, pp. 19611–19622, Sep. 2021.
[130] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Comput. Sci. Fac., Tech. Rep. TR06, Erciyes Univ., Kayseri, Turkey, 2005.
[131] D. Karaboga, "Artificial bee colony algorithm," *Scholarpedia*, vol. 5, no. 3, p. 6915, 2010.
[132] N. Hajizadeh, R. Javidan, P. Shamsinejad, and R. Akbari, "Node deployment in wireless sensor networks using the new multi-objective Lévy flight bee algorithm," *IET Wireless Sensor Syst.*, vol. 10, no. 2, pp. 78–87, 2019.
[133] V. M. Valsalan, "Dynamic deployment of wireless sensor networks using enhanced artificial bee colony algorithm," *Int. J. Sci. Res.*, vol. 2, no. 4, pp. 99–103, 2013.
[134] P. Y. He and M. Y. Jiang, "Dynamic deployment of wireless sensor networks by an improved artificial bee colony algorithm," in *Applied Mechanics and Materials*, vol. 511. Trans Tech Publ, 2014, pp. 862–866.
[135] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.
[136] Y. Qiao, H.-Y. Hsu, and J.-S. Pan, "Behavior-based grey wolf optimizer for wireless sensor network deployment problem," *Int. J. Ad Hoc Ubiquitous Comput. (IJAHUC)*, vol. 32, nos. 1–2, 2020.
[137] S. Wang, X. Yang, X. Wang, and Z. Qian, "A virtual force algorithm-Lévy-embedded grey wolf optimization algorithm for wireless sensor network coverage optimization," *Sensors*, vol. 19, no. 12, p. 2735, 2019.
[138] Z. Wang, H. Xie, Z. Hu, D. Li, J. Wang, and W. Liang, "Node coverage optimization algorithm for wireless sensor networks based on improved grey wolf optimizer," *J. Algorithms Comput. Technol.*, vol. 13, Jan. 2019, Art. no. 174830261988949.
[139] Z. Miao, X. Yuan, F. Zhou, X. Qiu, Y. Song, and K. Chen, "Grey wolf optimizer with an enhanced hierarchy and its application to the wireless sensor network coverage optimization problem," *Appl. Soft Comput.*, vol. 96, Nov. 2020, Art. no. 106602.
[140] X.-S. Yang and A. H. Gandomi, "Bat algorithm: A novel approach for global engineering optimization," *Eng. Comput.*, vol. 29, no. 5, pp. 464–483, 2012.
[141] S. S. Mohar, S. Goyal, and R. Kaur, "Optimized sensor nodes deployment in wireless sensor network using bat algorithm," *Wireless Pers. Commun.*, vol. 116, no. 4, pp. 2835–2853, Feb. 2021.
[142] M. Sharawi, E. Emary, I. A. Saroit, and H. El-Mahdy, "WSN's energy-aware coverage preserving optimization model based on multi-objective bat algorithm," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, May 2015, pp. 472–479.
[143] X.-S. Yang, "Flower pollination algorithm for global optimization," in *Proc. Int. Conf. Unconventional Comput. Natural Comput.* Springer, 2012, pp. 240–249.
[144] F. Hajjej, R. Ejbali, and M. Zaied, "Sensor node deployment approach in wireless sensor network based on multi-objective flower pollination algorithm," *Sensors Transducers*, vol. 213, no. 6, pp. 1–8, 2017.
[145] Z. Wang, H. Xie, D. He, and S. Chan, "Wireless sensor network deployment optimization based on two flower pollination algorithms," *IEEE Access*, vol. 7, pp. 180590–180608, 2019.
[146] C. Blum, J. Puchinger, G. R. Raidl, and A. Roli, "Hybrid metaheuristics in combinatorial optimization: A survey," *J. Appl. Soft Comput.*, vol. 11, no. 6, pp. 4135–4151, 2011.
[147] C. Blum and A. Roli, "Hybrid metaheuristics: An introduction," in *Hybrid Metaheuristics*. Springer, 2008, pp. 1–30.
[148] T. Ting, X.-S. Yang, S. Cheng, and K. Huang, "Hybrid metaheuristic algorithms: Past, present, and future," in *Recent Advances in Swarm Intelligence and Evolutionary Computation*. 2015, pp. 71–83.
[149] Y. Xu, O. Ding, R. Qu, and K. Li, "Hybrid multi-objective evolutionary algorithms based on decomposition for wireless sensor network coverage optimization," *Appl. Soft Comput.*, vol. 68, pp. 268–282, Jul. 2018.
[150] S. Mnasri, N. Nasri, A. V. D. Bossche, and T. Val, "A hybrid ant-genetic algorithm to solve a real deployment problem: A case study with experimental validation," in *Proc. Int. Conf. Ad-Hoc Netw. Wireless*. Springer, 2017, pp. 367–381.
[151] J. Tian, M. Gao, and G. Ge, "Wireless sensor network node optimal coverage based on improved genetic algorithm and binary ant colony algorithm," *EURASIP J. Wireless Commun. Netw.*, vol. 2016, no. 1, pp. 1–11, Dec. 2016.
[152] S. Mnasri, A. V. D. Bossche, N. Nasri, and T. Val, "The 3D redeployment of nodes in wireless sensor networks with real testbed prototyping," in *Proc. Int. Conf. Ad-Hoc Netw. Wireless*. Springer, 2017, pp. 18–24.
[153] M. Yuvaraja, R. Sabitha, and S. Karthik, "Hybrid PSO-bat algorithm with fuzzy logic based routing technique for delay constrained lifetime enhancement in wireless sensor networks," *J. Internet Technol.*, vol. 21, no. 2, pp. 479–487, 2020.
[154] D. Ma and Q. Duan, "A hybrid-strategy-improved butterfly optimization algorithm applied to the node coverage problem of wireless sensor networks," *Math. Biosci. Eng.*, vol. 19, no. 4, pp. 3928–3952, 2022.
[155] A. Boualem, M. Ayaida, and C. De Runz, "Hybrid model approach for wireless sensor networks coverage improvement," in *Proc. 8th Int. Conf. Wireless Netw. Mobile Commun. (WINCOM)*, Oct. 2020, pp. 1–6.
[156] C.-P. Chen, S. C. Mukhopadhyay, C.-L. Chuang, T.-S. Lin, M.-S. Liao, Y.-C. Wang, and J.-A. Jiang, "A hybrid memetic framework for coverage optimization in wireless sensor networks," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2309–2322, Oct. 2015.

[157] T. Holanda, T. Almeida, P. C. M. Teixeira, A. P. D. S. P. Rodrigues, and R. Lima, "A hybrid algorithm for deployment of sensors with coverage and connectivity constraints," *Int. J. Adv. Eng. Res. Sci.*, vol. 6, no. 3, pp. 13–19, 2019.
[158] T. S. Panag and J. S. Dhillon, "Maximal coverage hybrid search algorithm for deployment in wireless sensor networks," *Wireless Netw.*, vol. 25, no. 2, pp. 637–652, Feb. 2019.
[159] Y. El Khamlichi, A. Tahiri, A. Abtoy, I. Medina-Bulo, and F. Palomo-Lozano, "A hybrid algorithm for optimal wireless sensor network deployment with the minimum number of sensor nodes," *Algorithms*, vol. 10, no. 3, p. 80, 2017.
[160] X. Fan, Z. Zhang, X. Lin, and H. Wang, "Coverage hole elimination based on sensor intelligent redeployment in WSN," in *Proc. 4th Annu. IEEE Int. Conf. Cyber Technol. Autom., Control Intell.*, Jun. 2014, pp. 336–339.
[161] H. Deghbouch and F. Debbat, "A hybrid bees algorithm with grasshopper optimization algorithm for optimal deployment of wireless sensor networks," *Inteligencia Artif.*, vol. 24, no. 67, pp. 18–35, 2021.
[162] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: Theory and application," *Adv. Eng. Softw.*, vol. 105, pp. 30–47, Mar. 2017.
[163] D. Henderson, S. H. Jacobson, and A. W. Johnson, "The theory and practice of simulated annealing," in *Handbook of Metaheuristics*. Springer, 2003, pp. 287–319.
[164] F. D'Andreagiovanni and A. Nardin, "Towards the fast and robust optimal design of wireless body area networks," *Appl. Soft Comput.*, vol. 37, pp. 971–982, Dec. 2015.
[165] F. D'andreagiovanni, A. Nardin, and E. Natalizio, "A fast ILP-based heuristic for the robust design of body wireless sensor networks," in *Proc. Eur. Conf. Appl. Evol. Comput.* Springer, 2017, pp. 234–250.
[166] H. ZainEldin, M. Badawy, M. Elhosseini, H. Arafat, and A. Abraham, "An improved dynamic deployment technique based-on genetic algorithm (IDDT-GA) for maximizing coverage in wireless sensor networks," *J. Ambient Intell. Humanized Comput.*, vol. 11, pp. 4177–4194, Jan. 2020.
[167] O. Zorlu and O. K. Sahingoz, "Increasing the coverage of homogeneous wireless sensor network by genetic algorithm based deployment," in *Proc. 6th Int. Conf. Digit. Inf. Commun. Technol. Appl. (DICTAP)*, Jul. 2016, pp. 109–114.
[168] S. Mnasri, N. Nasri, A. van den Bossche, and T. Val, "Improved many-objective optimization algorithms for the 3D indoor deployment problem," *Arabian J. Sci. Eng.*, vol. 44, no. 4, pp. 3883–3904, Apr. 2019.
[169] A. Metiaf and Q. Wu, "Wireless sensor network deployment optimization using reference-point-based non-dominated sorting approach (NSGA-III)," in *Proc. J. Phys., Conf.*, vol. 1284, 2019, Art. no. 012063.
[170] Y. Qiao, T.-K. Dao, J.-S. Pan, S.-C. Chu, and T.-T. Nguyen, "Diversity teams in soccer league competition algorithm for wireless sensor network deployment problem," *Symmetry*, vol. 12, no. 3, p. 445, Mar. 2020.
[171] B. Al-Fuhaidi, A. M. Mohsen, A. Ghazi, and W. M. Yousef, "An efficient deployment model for maximizing coverage of heterogeneous wireless sensor network based on harmony search algorithm," *J. Sensors*, vol. 2020, Nov. 2020, Art. no. 8818826.
[172] C.-W. Tsai, "An effective WSN deployment algorithm via search economics," *Comput. Netw.*, vol. 101, pp. 178–191, Jun. 2016.
[173] A. Boubrima, W. Bechkit, and H. Rivano, "Optimal WSN deployment models for air pollution monitoring," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 2723–2735, May 2017.
[174] M. R. Senouci and A. Mellouk, "A robust uncertainty-aware cluster-based deployment approach for WSNs: Coverage, connectivity, and lifespan," *J. Netw. Comput. Appl.*, vol. 146, Nov. 2019, Art. no. 102414.
[175] M. Mohammed, M. B. Khan, and E. B. M. Bashier, *Machine Learning: Algorithms and Applications*. Boca Raton, FL, USA: CRC Press, 2016.
[176] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 1996–2018, 4th Quart., 2014.
[177] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2224–2287, 3rd Quart., 2019.
[178] L. Ye, S. Roy, and S. Sundaram, "Resilient sensor placement for Kalman filtering in networked systems: Complexity and algorithms," *IEEE Trans. Control Netw. Syst.*, vol. 7, no. 4, pp. 1870–1881, Dec. 2020.
[179] V. Tzoumas, A. Jadbabaie, and G. J. Pappas, "Sensor placement for optimal Kalman filtering: Fundamental limits, submodularity, and algorithms," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2016, pp. 191–196.
[180] M. W. M. Seah, C.-K. Tham, V. Srinivasan, and A. Xin, "Achieving coverage through distributed reinforcement learning in wireless sensor networks," in *Proc. 3rd Int. Conf. Intell. Sensors, Sensor Netw. Inf.*, 2007, pp. 425–430.
[181] Y. P. Tsang, K. L. Choy, C. H. Wu, and G. T. S. Ho, "Multi-objective mapping method for 3D environmental sensor network deployment," *IEEE Commun. Lett.*, vol. 23, no. 7, pp. 1231–1235, Jul. 2019.
[182] F. Hajjej, M. Hamdi, R. Ejbali, and M. Zaied, "A distributed coverage hole recovery approach based on reinforcement learning for wireless sensor networks," *Ad Hoc Netw.*, vol. 101, Apr. 2020, Art. no. 102082.
[183] L. O. Philco, L. Marrone, and E. Estupiñan, "MiA-CODER: A multi-intelligent agent-enabled reinforcement learning for accurate coverage hole detection and recovery in unequal cluster-tree-based QoSensing WSN," *Appl. Sci.*, vol. 11, no. 23, p. 11134, Nov. 2021.
[184] L. A. Zadeh, "Fuzzy logic," *Computer*, vol. 21, no. 4, pp. 83–93, Apr. 1988.
[185] L. Zhao and Q. Liang, "Fuzzy deployment for wireless sensor networks," in *Proc. IEEE Int. Conf. Comput. Intell. Homeland Secur. Pers. Saf. (CIHSPS)*, Apr. 2005, pp. 79–83.
[186] M. Maksimović, V. Vujović, and V. Milošević, "Fuzzy logic and wireless sensor networks—A survey," *J. Intell. Fuzzy Syst.*, vol. 27, no. 2, pp. 877–890, 2014.
[187] H. Shu and Q. Liang, "Fuzzy optimization for distributed sensor deployment," in *Proc. IEEE Wireless Commun. Netw. Conf.*, vol. 3, Mar. 2005, pp. 1903–1908.
[188] A. M. Rahmani, S. Ali, M. S. Yousefpoor, E. Yousefpoor, R. A. Naqvi, K. Siddique, and M. Hosseinzadeh, "An area coverage scheme based on fuzzy logic and shuffled frog-leaping algorithm (SFLA) in heterogeneous wireless sensor networks," *Mathematics*, vol. 9, no. 18, p. 2251, Sep. 2021.
[189] D. Izadi, J. Abawajy, and S. Ghanavati, "An alternative node deployment scheme for WSNs," *IEEE Sensors J.*, vol. 15, no. 2, pp. 667–675, Feb. 2015.
[190] R. Tarjan, "Depth-first search and linear graph algorithms," *SIAM J. Comput.*, vol. 1, no. 2, pp. 146–160, 1972.
[191] B. Cao, J. Zhao, P. Yang, P. Yang, X. Liu, and Y. Zhang, "3-D deployment optimization for heterogeneous wireless directional sensor networks on smart city," *IEEE Trans. Ind. Informat.*, vol. 15, no. 3, pp. 1798–1808, Mar. 2019.
[192] B. Cao, X. Kang, J. Zhao, P. Yang, Z. Lv, and X. Liu, "Differential evolution-based 3-D directional wireless sensor network deployment optimization," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3594–3605, Oct. 2018.

KHAOULA ZAIMEN received the degree in computer science engineering with computer systems specialization from the École Nationale Supérieure d'Informatique (ESI), Algeria, in 2020. She is currently pursuing the Ph.D. degree with the LINEACT Research Laboratory, CESI Engineering School, and the IRIMAS Research Laboratory, University of Haute-Alsace (UHA). She also works on its Ph.D. thesis which is about optimizing the deployment of wireless sensor networks (WSN) in smart buildings based on the BIM database.

MOHAMED-EL-AMINE BRAHMIA (Member, IEEE) received the degree in engineering from the University of Guelma, Algeria, in 2006, the master's degree from the University of Versailles, France, in 2008, and the Ph.D. degree from the University of Haute-Alsace, France, in 2012. He has been an Associate Professor at the CESI Engineering School, Strasbourg, France, since 2013. His research interests include the Internet of Things, WSN, routing, blockchain, and IA as well as modeling and solving optimization problems related to smart cities, smart buildings, and security architectures. He is also the Chair of many symposiums, including SSCC and SIAS.

LAURENT MOALIC received the Ph.D. degree in computer science from the University of Franche Comté, in 2013. From 2004 to 2017, he worked as a Research Engineer at the University of Technology of Belfort-Montbéliard, Belfort, France. Since 2017, he has been an Associate Professor at the University of Haute-Alsace, Mulhouse, France. His research is carried out at the IRIMAS Institute. His research focuses on optimization algorithms, especially based on metaheuristic approaches, such as genetic algorithms, local searches, and especially hybrid methods. He is particularly interested in solving NP-hard combinatorial problems. His research focuses theoretical problems, such as graph coloring or geometric optimization, as well as on real-world problems, most often based on human mobility modeling. His research is notably used to optimize the deployment of electric vehicle fleets, and more generally mobility services.

ABDELHAFID ABOUAISSA received the M.S. degree from the University of Franche-Comte, Besançon, France, in 1996, the Ph.D. degree from the University of Technology of Belfort-Montbéliard, France, in January 2000, and the Accreditation to supervise research from the University of Haute-Alsace, France, in 2017. He has been a Full Professor of computer science at the University of Haute-Alsace, since 2019. He is the author/coauthor of five patents and 180 published international journals and conferences. His research interests include security, resource management in WSN, the IoT, 5G networks, e-health, blockchain, and IA.

LHASSANE IDOUMGHAR (Member, IEEE) received the Ph.D. degree from the University of Henri Poincaré, Nancy 1, France, in 2003, and the Accreditation to supervise research from the University of Haute-Alsace, Mulhouse, France, in 2012. From 2000 to 2003, he worked with the TDF-C2R Broadcasting and Wireless Research Center. He has been a Full Professor at the University of Haute-Alsace, since 2015. Until 2017, he was the Associate Director of the Laboratoire de Mathématiques, Informatique et Application (LMIA Laboratory), University of Haute-Alsace. He was elected as the Deputy Director of the Institut de Recherche en Informatique, Mathématique, Automatique et Signal (IRIMAS Institute), and the Director of the Computer Science Research Department, IRIMAS Institute, from January 2018 to December 2019. Since January 2020, he has been the Director of the IRIMAS Institute. He is currently the Head of the Optimization by MEtaheuristics, alGorithms and modelization (OMEGA) Research Team. His research interests include artificial intelligence, evolutionary algorithm, massively parallel and distributed metaheuristics, and optimization and uncertain optimization by hybrid metaheuristics.