Path Optimization for Mobile Sensors to Monitor Coverage Holes in Wireless Sensor Networks

Maher Rebai(1)* and Jinane Charafeddine(2) and Taha Houda(3) and Wafaa Shakir(4)

(1),(2)De Vinci Higher Education
Av. Léonard de Vinci, 92400 Courbevoie, France

(3)Price Mohammad Bin Fahd University
617, Al Jawharah, Khobar, Dhahran 34754, Saudi Arabia

(4)Al-Furat Al-Awsat Technical University
babylon-najaf street, Najaf, Najaf Governorate, 54003, Irak

Abstract. Efficient path planning for mobile sensors is crucial in Wireless Sensor Networks (WSNs) to ensure optimal monitoring of coverage holes while considering real-world constraints. This work addresses the problem of determining an optimal trajectory for mobile sensor ensuring optimal monitoring of coverage holes while efficiently navigating through the sensing field. We introduce a novel resolution approach, based on solving two Binary Integer Linear Programming (BILP) models. The first model determines the minimum set of stop locations needed to achieve full coverage of the sensing area, while the second model selects these locations and computes the shortest possible tour that connects them. The performance of the proposed approach is thoroughly evaluated through comparative experiments against both exact and heuristic methods from the literature. The obtained results confirm that the proposed approach outperforms the recent existing methods.

Keywords: Wireless Sensor Network (WSN), Linear programming.

1 Introduction

Wireless Sensor Networks (WSNs) have become a cornerstone technology in various domains, including environmental monitoring, disaster management, smart cities, and the rapidly expanding Internet of Things (IoT) [1]. These networks are composed of spatially distributed sensor nodes that collect and transmit data to a central sink for analysis. However, due to sensor failures, energy depletion, or suboptimal initial deployment, certain areas may remain uncovered—commonly referred to as coverage holes—resulting in monitoring gaps and reduced network efficiency. Ensuring full and reliable coverage is crucial for the effectiveness of WSN applications, making the detection and resolution of coverage holes a significant challenge.

A practical solution to address this issue is the deployment of mobile sensor nodes, which can dynamically adjust their positions to cover uncovered regions and restore network integrity. However, determining the optimal movement strategy for these mobile sensors is complex, as it requires balancing maximum coverage, minimal travel distance, and energy efficiency. Inefficient path planning can lead to excessive energy consumption and increased latency, which may degrade the network's overall performance.

David C. Wyld et al. (Eds): CMLA, CSEIT, NLPD, GRAPH-HOC, SPM, NeTCoM, NCS, ArIT, DMSE – 2025
pp. 65-75, 2025. CS & IT- CSCP 2025 DOI: 10.5121/csit.2025.151306

In this work, we propose a novel approach based on two Binary Integer Linear Programming (BILP) models to efficiently guide mobile sensor node in covering coverage holes within the sensing field. The proposed method is evaluated through simulations across different network configurations, accounting for various node densities and sensing ranges.

The main contributions of this study are as follows: (i) We introduce an optimal resolution method based on a two Binary Integer Linear Programming (BILP) models, without restrictive assumptions about mobility patterns, to determine the shortest trajectory for mobile sensors.
(ii) Our approach effectively handles both regular and irregular sensing areas.
(iii) We provide extensive numerical and experimental results to assess the effectiveness and performance of our solution compared to recent state-of-the-art approaches.

The remainder of this paper is structured as follows: Section 2 reviews related work on coverage hole detection and mobile sensor deployment strategies. Section 3 presents the problem definition and key assumptions. Section 4 details the two Binary Integer Linear Programming (BILP) models. Section 5 evaluates the performance of our approach through extensive simulations. Finally, Section 6 concludes the paper and outlines future research directions.

2 Related work

The integration of mobile sensors in Wireless Sensor Networks (WSNs) has been widely explored as a solution to enhance event detection in uncovered areas [2],[3] and mitigate the risk of missed events due to sensor failures or imperfect initial deployment of static sensors.

Research on mobility in WSNs generally falls into two categories: (i) employing mobile sensor nodes to traverse the monitored area and cover uncovered regions, and (ii) using a mobile base station (sink) to move within the network and collect data from static sensors [4],[5]. As our study focuses on the first category, we limit our discussion to works addressing the deployment of mobile sensors for coverage restoration.

One of the primary challenges in mitigating coverage holes is detecting uncovered areas and optimally deploying mobile sensors to restore full coverage. Various approaches have been proposed to address this problem. The Zoom algorithm [6] applies a divide-and-conquer strategy to identify coverage holes and plan sensor movements, ensuring maximum coverage while avoiding obstacles. An enhanced version [4] introduces a collision-avoidance mechanism, allowing mobile nodes to share movement data and prevent redundant coverage.

Path optimization for mobile sensors has also been investigated using game theory models, particularly in security applications like intrusion detection [2]. In such scenarios, a security sensor must strategically navigate the field to detect an intruder attempting to remain hidden. Another approach employs a dynamic auction-based protocol [7], where static sensors detect coverage holes using Voronoi diagrams and send bids to mobile sensors, which then move towards the most critical gaps. A variant of this approach [8] replaces Voronoi diagrams with the Zoom algorithm to reduce computational complexity while requiring global network knowledge.
Other strategies leverage potential field methods [9], where mobile nodes adjust their positions autonomously based on inter-node distances and coverage constraints. While effective in unknown environments, this method may leave residual gaps in case of node failures. A

similar idea is applied in [10],[11], where Voronoi diagrams help mobile nodes reposition cooperatively to maintain network connectivity while covering uncovered regions.
Several studies focus on optimizing the number and placement of mobile sensors to ensure coverage restoration. For instance, [12] applies the Harmony Search Algorithm to estimate the minimum number of mobile sensors needed to cover blind zones left by randomly deployed static sensors. Similarly, [13] employs a Genetic Algorithm to determine the optimal number of mobile sensors required to efficiently resolve coverage gaps.
Trajectory optimization plays a crucial role in reducing sensor movement and energy consumption. In [14], the problem is modeled as a grid-based optimization challenge using 0-1 Integer Programming (IP) models, without restrictive mobility pattern assumptions, to determine the shortest trajectory for mobile sensors. The authors prove that the problem is NP-hard and propose two heuristic algorithms along with a hybrid approach to approximate the optimal solution. In [15], the authors introduce three Mixed Integer Linear Programming (MILP) models for optimizing mobile sensor routing. These models aim to achieve the required coverage level within a given time constraint. The first formulation maximizes area coverage, while the second and third prioritize minimizing travel time. Comparative analysis against heuristic methods demonstrates the effectiveness of these models in achieving high-quality solutions within reasonable computation times.

In this study, we aim to develop an exact optimization approach for mobile sensor trajectory planning to monitor coverage holes in WSNs. Our approach integrates two Binary Integer Linear Programming (BILP) models to efficiently solve problem instances while reducing computational time. We compare our approach's results with the recent models proposed in [14] as they adress the same problem.

3 Problem description and assumptions

We model the sensing area as an $(M \times N)$ grid $(width \times length)$ discretized into $M \times N$ points where each adjacent point is separated by a distance of one measurement unit. Each grid position (i, j) can host one sensor. Each sensor has a sensing range R_{cov} and can monitor all grid positions (k, t) that satisfy:

$$(i-k)^2 + (j-t)^2 \leq R_{cov}^2$$

The sensing area contains a set of active sensors, but due to sensor failures, suboptimal initial deployment, or sensor energy depletion, some regions remain unmonitored. Our objective is to determine the optimal grid positions—called critical grid positions—where a mobile sensor should stop to sense or change direction to dynamically cover all uncovered regions while minimizing travel distance.
In Fig. 1, gray points represent covered cells, while white points indicate uncovered areas. The stars mark critical stopping points where the mobile sensor must adjust its trajectory to ensure full coverage while navigating through the sensing area. The sensor's movement path is depicted by red stars. In this study, we assume that the mobility pattern of the mobile sensor can follow predefined trajectories, optimizing coverage efficiency based on prior knowledge of the area.

4 Resolution approach

The proposed BILP models are founded on representing the $M \times N$ grid as a graph $G = (V, E)$, where each grid position (i, j) is mapped to a vertex v, indexed as $v = i \times M + j$, with M denoting the grid's width. An edge (p, q) is included in the graph if the mobile

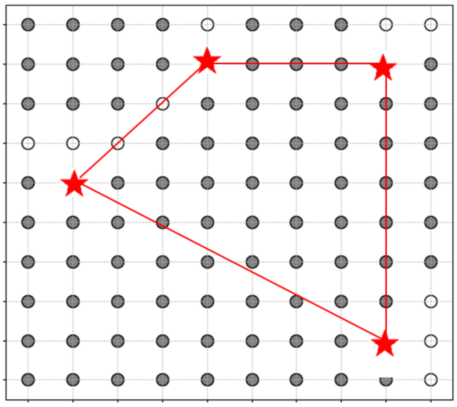

Fig. 1. Example for dynamically complete grid coverage by a mobile sensor with $R_{cov} = 1.5$

sensor can visit vertex p and then moves directly to vertex q. Furthermore, we define $\theta(j)$ as the set of vertices capable of covering vertex j, and γ as the set of currently uncovered vertices within the graph.

To address this problem, we first consider the selection of a minimal set of locations from which the uncovered vertices can be covered, and then construct the shortest cycle that visits these locations. This leads to the following proposition, which underpins our optimization strategy:

Proposition. The shortest cycle that allows a mobile sensor to cover all vertices in γ must visit a minimum-size subset of vertices.

Proof. Let $\gamma \subseteq V$ be the set of uncovered vertices in the graph, and let $\theta(j) \subseteq V$ denote the set of vertices from which vertex $j \in \gamma$ can be covered. Let $S \subseteq V$ represent the set of vertices visited by the mobile sensor to ensure coverage of γ.

The objective is to construct a cycle of minimum length that visits a subset of vertices such that each vertex in γ is covered by at least one visited vertex. To minimize the path length, the cycle must visit the fewest possible vertices, as visiting additional vertices increases the total length of the tour.

Assume by contradiction that the shortest cycle does not correspond to a minimum-size subset of visited vertices covering γ, i.e., there exists a cycle covering γ that includes more vertices than necessary. In that case, we can construct an alternative cycle that only visits the minimal set of vertices needed to cover γ, which would result in a shorter tour — contradicting the assumption that the original cycle was the shortest.

Conclusion. The shortest cycle covering γ must visit a minimum-size subset of vertices that collectively ensure full coverage. ■

Minimizing mobile sensor stop vertices

To minimize the number of stop locations for the mobile sensor, we reformulate the problem as a target location coverage problem, where the goal is to identify the minimum number of sensor placements required to cover a predefined set of target points. By treating the uncovered vertices as the set of targets to be covered, the solution to this problem provides the smallest subset of vertices from which complete coverage can be achieved. This subset directly determines the minimal set of positions the mobile sensor must visit to form its tour.

To this end, we use the BILP model proposed in [16] to solve the mobile sensor stop vertices minimization problem:

$$x_i = \begin{cases} 1, & \text{if a sensor is assigned to vertex } i, \\ 0, & \text{otherwise.} \end{cases}$$

$$Minimize \sum_{i \in \gamma} x_i \tag{1}$$

Subject to

$$\sum_{i \in \theta(j)} x_i \geq 1, \qquad \forall j \in \gamma \tag{2}$$

$$x_i \in \{0, 1\} \tag{3}$$

The objective function (1) minimizes the number of critical grid positions by reducing the number of selected vertices in the graph that collectively ensure complete coverage. The set of constraints (2) ensures that each uncovered vertex $j \in \gamma$ is covered by at least one vertex in the set $\theta(j)$, representing the positions from which j can be sensed. Finally, constraints (3) impose the binary nature of the decision variables x_i. This BILP model includes $M \times N$ decision variables and $|\gamma|$ coverage constraints, excluding the binary constraints.

Minimizing mobile sensor trajectory

we describe in this subsection the BILP model designed to solve the mobile sensor trajectory minimization problem.
Let d_{ij} denotes the Euclidian distance between vertex i and vertex j and the parameter nb corresponds to the total number of positions selected as stop locations for the mobile sensor, as computed by the BILP model described earlier.
the BILP model uses the following decision variables:

$$x_i = \begin{cases} 1, & \text{if the mobile sensor should stop at vertex } i, \\ 0, & \text{otherwise.} \end{cases}$$

$$y_{ij} = \begin{cases} 1, & \text{if an edge is created from vertex } i \text{ to vertex } j, \\ 0, & \text{otherwise.} \end{cases}$$

The BILP model can be formulated as follows:

$$Minimize \sum_{i \in \gamma} \sum_{\substack{j=1 \\ i \neq j}}^{M \times N} \sqrt{d_{ij}} \cdot y_{ij} \tag{4}$$

Subject to:

$$\sum_{i \in \theta(j)} x_i \geq 1, \qquad \forall j \in \gamma \tag{5}$$

$$y_{ij} \leq x_i, \qquad \forall i \in \gamma, j \in \{1, \ldots, M \times N\}, i \neq j, \forall nb > 2 \tag{6}$$

$$y_{ij} \leq x_j, \qquad \forall i \in \gamma, j \in \{1, \ldots, M \times N\}, i \neq j, \forall nb > 2 \tag{7}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{M \times N} y_{ij} = 2x_i, \qquad \forall i \in \{1, \ldots, M \times N\}, \forall nb > 2 \tag{8}$$

$$x_i \leq \sum_{\substack{t=1 \\ t \neq i}}^{M \times N} y_{it}, \qquad \forall i \in \{1, \ldots, M \times N\}, \forall nb > 2 \quad (9)$$

$$y_{ij} + y_{ji} \leq 1, \qquad \forall i, j \in \{1, \ldots, M \times N\}, i \neq j, \forall nb > 2 \quad (10)$$

$$y_{ij} \leq \sum_{\substack{t=1 \\ t \neq i, t \neq j}}^{M \times N} y_{jt}, \qquad \forall i, j \in \{1, \ldots, M \times N\}, i \neq j, \forall nb > 2 \quad (11)$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{M \times N} y_{ij} \leq 1, \qquad \forall i \in \gamma, \forall nb > 2 \quad (12)$$

$$y_{ij} \leq \sum_{\substack{t=1 \\ t \neq i, t \neq j}}^{M \times N} y_{ti}, \qquad \forall i, j \in \{1, \ldots, M \times N\}, i \neq j, \forall nb > 2 \quad (13)$$

$$\sum_{\substack{i \in \gamma \\ j \neq i}} y_{ij} \leq 1, \qquad \forall j \in \{1, \ldots, M \times N\}, \forall nb > 2 \quad (14)$$

$$\sum_{i \in \gamma} x_i = \text{nb} \quad (15)$$

$$\sum_{\substack{j \in \theta_k(i) \\ j \neq i}} y_{ij} \geq k - 1, \qquad \forall i \in \gamma, \forall k \in \{1, \ldots, nb\}, \forall nb > 2 \quad (16)$$

$$x_i + x_j \leq y_{ij} + y_{ji} \qquad \forall i, j \in \{1, \ldots, M \times N\}, i \neq j, \forall nb = 2 \quad (17)$$

$$x_i, y_{ij} \in \{0, 1\} \quad (18)$$

In this model, the objective function (4) seeks to minimize the total travel distance of the mobile sensor by reducing the cumulative length of the edges forming its trajectory. Constraints (5) ensure that each uncovered vertex j is covered at least once during the mobile sensor's tour. Constraints (6) and (7) guarantee that an edge (i, j) can only be selected if both vertices i and j are included in the tour. Constraint (8) enforces that each selected vertex is connected to exactly two other vertices, thereby forming a cycle. Constraint (9) ensures that each selected vertex has at least one incident edge. Constraint (10) defines the directionality of the arcs. Constraints (11) and (12) ensure that each arc in the tour has exactly one outgoing edge, while constraints (13) and (14) ensure that each arc has exactly one incoming edge. Constraint (15) defines the total number of edges composing the mobile sensor's cycle. Constraint (16) eliminates the formation of subtours. Constraint (17) ensures that the model still yields a valid solution when the optimal tour consists of a single edge. The remaining constraints define the binary nature of the decision variables.

5 experimental evaluation

In this section, we conduct numerical experiments on various grid sizes to assess the performance of the proposed approach. The benchmark instances used in our study are inspired by those introduced in [14].

We consider the random deployment of static sensors over a grid of size $N \times M$, where $N \in \{7, 10, 13\}$, $M \in \{7, 10, 13\}$, and sensor density levels $\alpha \in \{0.1, 0.2, 0.3\}$. Specifically, for a grid of size $N \times M$, a total of $\alpha(M \times N)^2$ sensors are randomly deployed. As the value of α decreases, the number of coverage holes in the grid tends to increase. Although the distribution of sensors can influence the presence of coverage holes, instances with lower α are generally more challenging. To ensure statistical reliability, we report average results over 30 randomly generated instances per configuration. Each instance is generated by distributing the sensors randomly over the grid, avoiding any fixed deployment pattern that might bias the results. In our simulations, we consider that the mobile sensor has a sensing range $R_{\text{cov}} \in \{1.5, 2, 2.5\}$. The proposed approach is implemented in Python using the OR-Tools optimization library [17], and all experiments are conducted on a server equipped with an *Intel(R) Xeon(R) CPU E5649 @ 2.53GHz*, running Python 3.9.5. The time limit for solving each instance is set to one hour (3600 seconds).

able 1 reports the average computational time for each combination of coverage range, sensor density α, and grid size.

		Grid size (N, M)					
α	R_{cov}	(7,7)	(7,10)	(7,13)	(10,10)	(10,13)	(13,13)
0.1	1.5	90,32	461,12	895,38	2925,94	2083,95	>3600
	2	35,16	261,96	465,41	1124,45	1105,5	>3600
	2.5	19,71	73,63	171,08	429,82	657	2544,87
0.2	1.5	14,19	169,71	765,52	642,79	882,36	2589,49
	2	1,57	47,16	29,33	456,3	419,69	1907,63
	2.5	10,22	0,16	0,27	0,42	1,52	510,18
0.3	1.5	5,67	12,39	327,06	223,7	258,55	1109,48
	2	0,18	0,24	59,16	0,51	151,14	2,81
	2.5	0,02	0,13	0,28	0,71	0,89	1,06

Table 1. Average computational time (in seconds) to solve an instance.

As shown in Table 1, higher values of R_{cov} are associated with reduced average computation times. This is due to the enlarged coverage capacity, which expands the feasible region and thus simplifies the search for optimal solutions, despite the added complexity from additional decision variables.

Table 2 illustrates the proportion of problem instances successfully solved to optimality using the BILP models. While the developed method performs well on small to medium grid sizes, the percentage of optimally solved cases significantly drops as the grid size increases, especially for configurations such as 10×10 and 13×13. We propose, in the following, to compare the results of our proposed approach with those of the exact and the heuristic approach in [14] for the regular grid sizes 7×7 and 10×10 and 13×13. The [14]'s proposed Mixed Integer Linear Programming (MILP) model will be referred

Grid size	(7,7)	(7,10)	(7,13)	(10,10)	(10,13)	(13,13)
% of solved instances within 1h	100	100	100	96	100	75

Table 2. Percentage (%) of optimally solved instances within 1h computational time.

to as Cav_MILP while [14]'s proposed heuristic will be referred to as Cav_Heuristic. Our developed approach will be denoted by BILP_model.
We first comapre the results of our BILP models with those of [14]'s MILP model. Comparison between the two exact approaches will be done on the percentage (%) of optimally solved instances within the prefixed computational time period.
As shown in Figure 2, our approach consistently outperforms the exact method from

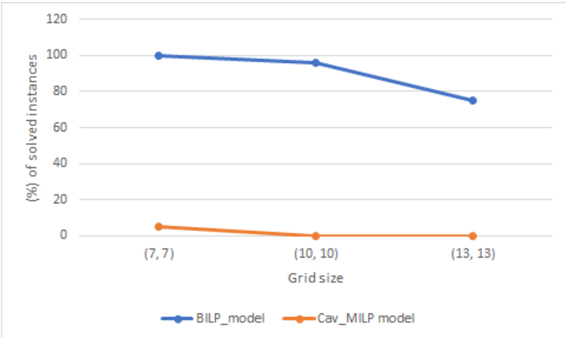

Fig. 2. (%) of optimally solved instances by the exacts methods.

the literature across all tested grid sizes. This improvement is mainly due to the reduced complexity of our BILP model, which involves significantly fewer decision variables and constraints compared to the MILP formulation in [14]. Specifically, our model uses $M \times N$ decision variables x_i and $(M \times N)^2 - (M \times N)$ variables y_{ij}, totaling $(M \times N)^2$ decision variables. In contrast, the Cav_MILP model includes $(M \times N)^2 - (M \times N)$ variables y_{ij}, $\gamma \times (M \times N)$ variables x_{ij}, and $M \times N$ variables u_i, resulting in $(M \times N)^2 + \gamma(M \times N)$ decision variables. Regarding constraints, our model requires $\gamma(1+4MN)+4M^2N^2-2MN+1+nb$ constraints, whereas the MILP model from [14] needs $5M^2N^2+2\gamma MN$ constraints. These reductions in model size explain the better scalability and computational efficiency of our approach.

In the following, we compare the performance of our BILP model with the heuristic method proposed in [14], focusing on two key aspects: solution quality and computational efficiency.

Figure3 illustrates the percentage of instances in which the BILP model outperforms the heuristic method proposed in[14] (Cav_Heuristic) with respect to solution quality. The results indicate that the superiority of the BILP model becomes more pronounced as the grid size increases, with a growing percentage of instances where it yields better solutions.

For the analysis of average computational time, we refer the reader to Figure 4. Figure 4 shows that the difference in computational time between the BILP model and the heuristic approach is relatively small for smaller instances, as these are typically solved to optimality quickly. However, as the instance size increases, the gap becomes significantly

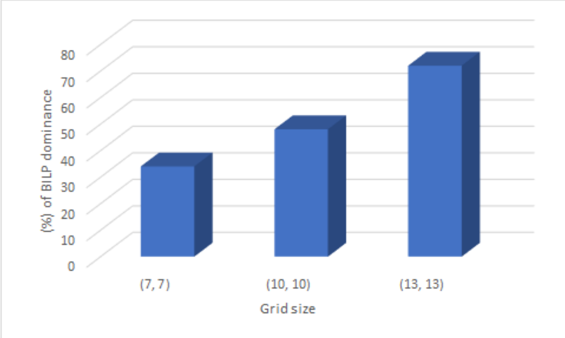

Fig. 3. Percentage of dominance of the solutions from the two-phase method.

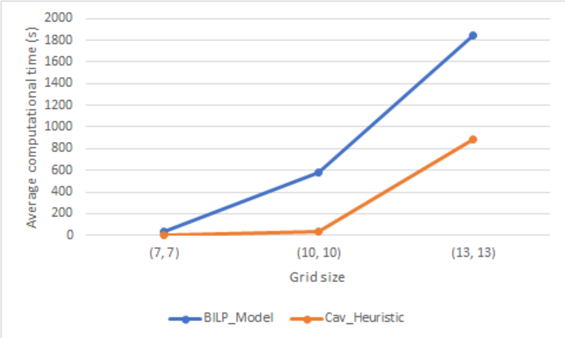

Fig. 4. Comparison of average computational time for each grid size.

more noticeable. Additionally, the computational time gap is influenced by the sensing range R_{cov}; specifically, as R_{cov} increases, the gap tends to decrease.

Based on the insights from Figures 3 and 4, we conclude that while the proposed BILP model consistently delivers superior solution quality compared to the heuristic method from the literature, it does so at the cost of increased computational time.

6 Conclusion

In this paper, we introduced a novel approach to solve the optimal path planning problem for a mobile sensor assigned to monitor coverage holes in Wireless Sensor Networks. Our methodology is structured in two main phases: first, we formulate a Binary Integer Linear Programming (BILP) model to identify the minimum set of sensor positions required to fully cover the sensing area. Then, we apply a second BILP model to determine the optimal sequence of stop points for the mobile sensor, ensuring coverage with the shortest possible tour.

Comparative experimental results confirm the effectiveness of the proposed approach, showing that it consistently outperforms recent methods from the literature in terms of solution quality.

As future work, we intend to strengthen the proposed BILP models by integrating additional valid inequalities to reduce computational time and enable the resolution of larger-scale instances. Furthermore, we plan to improve the linear relaxation of the BILP formulations and incorporate them into a branch-and-bound framework for enhanced optimization performance.

References

1. Nguyen, T.H., Huynh, T.T.B., Vu, Q.T., Nguyen, P.T. and Huynh C.P. (2023) 'Node placement optimization under Q-Coverage and Q-Connectivity constraints in wireless sensor networks', *Journal of Network and Computer Applications*, Vol.212, pp.103–578
2. Benyuan Liu , Olivier Dousse , Philippe Nain , Don Towsley , Dynamic coverage of mobile sensor networks, IEEE Trans. Parallel Distrib. Syst. 24 (2) (2013) 301–311 .
3. Li Zhu , Chunxiao Fan , Huarui Wu , Zhigang Wen , Coverage Optimization Algo- rithm of Wireless Sensor Network Based on Mobile Nodes, Int. J. Online Eng. (iJOE) 12 (08) (2016) 45–50 .
4. Theofanis P. Lambrou , Christos G. Panayiotou , Santiago Felici , Baltasar Beferull , Exploiting mobility for efficient coverage in sparse wireless sensor networks, Wirel. Pers. Commun. 54 (1) (2010) 187–201 .
5. Jun Luo , J-P. Hubaux , Joint mobility and routing for lifetime elongation in wire- less sensor networks, in: INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, 3, IEEE, 2005, pp. 1735–1746 .
6. Theofanis P. Lambrou , Christos G. Panayiotou , Collaborative area monitoring using wireless sensor networks with stationary and mobile nodes, EURASIP J. Adv. Signal Process. (2009) (2009) 7 .
7. Guiling Wang , Guohong Cao , Piotr Berman , Thomas F. La Porta , Bidding proto- cols for deploying mobile sensors, IEEE Trans. Mob. Comput. 5 (2007) 563–576 .
8. Massimo Vecchio , Roberto López-Valcarce , Improving area coverage of wireless sensor networks via controllable mobile nodes: a greedy approach, J. Netw. Comput. Appl. 48 (2015) 1–13 .
9. Andrew Howard , Maja J. Matari ´c , Gaurav S. Sukhatme , Mobile sensor net- work deployment using potential fields: A distributed, scalable solution to the area coverage problem, in: Distributed Autonomous Robotic Systems 5, Tokyo, Springer, 2002, pp. 299–308 . 66–78 .
10. Wei Fang , Xinhong Song , Xiaojun Wu , Jun Sun , Mengqi Hu , Novel efficient de- ployment schemes for sensor coverage in mobile wireless sensor networks, Inf. Fusion 41 (2018) 25–36 .
11. Chenxi Qiu , Haiying Shen , Kang Chen , An energy-efficient and distributed co- operation mechanism for k-coverage hole detection and healing in WSNs, IEEE Trans. Mob. Comput. 17 (6) (2018) 1247–1259 .
12. A. Mohsen, W. Aljoby, K. Alenezi, A. Alenezi, A robust harmony search algorithm based markov model for node deployment in hybrid wireless sensor networks, GEOMATE J. 11 (27) (2016) 2747–2754.
13. O. Banimelhem, M. Mowafi, W. Aljoby, Genetic algorithm based node deployment in hybrid wireless sensor networks, Commun. Netw. 5 (2013) 273–279.
14. Cansu Cav, Aysegül Altin-Kayhan, Coverage hole optimization with a mobile sensor in wireless sensor networks for smart grid, Ad Hoc Networks 140 (2023) 103039
15. Christopher Zygowski, Arunita Jaekel, Optimal path planning strategies for monitoring coverage holes in Wireless Sensor Networks, Ad Hoc Networks 96 (2020) 101990
16. M Rebai, H Snoussi, F Hnaien, L Khoukhi, Sensor deployment optimization methods to achieve both coverage and connectivity in wireless sensor networks *Computers and Operations Research* 59, 11-21 (2015)
17. L. Perron and V. Furnon. OR-Tools. *Google*, https://developers.google.com/optimization/ (2023).

Authors

M. Rebai Maher Rebai obtained his Ph.D. in Optimization and System Safety from the University of Technology of Troyes. Since 2015, he has been a computer science teacher and researcher at De Vinci Higher Education in Paris La Défense. His research focuses on combinatorial optimization, scheduling, wireless sensor network optimization, as well as Fog and Edge computing.

J. Charafeddine is a Research Professor at Devinci Higher Education, Paris La Défense. Her research focuses on Biomechanics and Artificial Intelligence applied to Medical Computer Vision. She earned her Ph.D. in Motion Science and Control of Mechatronic Systems from Paris-Saclay University, Orsay, France, in 2021. Previously, she worked as a Lecturer in the Department of Mechatronics and Digital Systems Engineering at Université Paris-Saclay and the engineering school “Institut des Sciences et Techniques des Yvelines” (ISTY). She was also affiliated with the Laboratoire d’Ingénierie des Systèmes de Versailles (LISV) as a Researcher.

T. Houda Taha Houda is an Assistant Professor of Computer Engineering at Prince Mohammad Bin Fahd University, Saudi Arabia. He received his Ph.D. in Robotics and Artificial Intelligence from Paris-Saclay University, France. His research interests include artificial intelligence, robotics, medical image analysis, human activity recognition, and immersive technologies such as virtual and augmented reality. Dr. Houda has published in several peer-reviewed journals and international conferences, and he is actively involved in interdisciplinary research projects, particularly in healthcare and smart systems.

W.M. R. SHAKIR received the Ph.D. degree in communications and electronics engineering from the University of Technology, Baghdad, Iraq, in 2011. She has been a Faculty Member with the Technical Institute of Babylon, Al-Furat Al-Awsat Technical University, Babil, Iraq, since 2005. She was a Research Fellow with The Pennsylvania State University, PA, USA, a Postdoctoral Fellow with The University of Queensland, QA, Australia. Since 2022, she is an Associate Research Fellow with Paris-Saclay University, France. She has been an Associate Professor of wireless communications engineering with the Department of Computer Networks and Softwares, at Al-Furat Al-Awsat Technical University since 2011. Her current research interests include artificial intelligence, modeling, design, and performance analysis of wireless communication systems.

© 2025 By AIRCC Publishing Corporation. This article is published under the Creative Commons Attribution (CC BY) license.