Hindawi
Wireless Communications and Mobile Computing
Volume 2020, Article ID 8833767, 11 pages
https://doi.org/10.1155/2020/8833767

Research Article

Bodacious-Instance Coverage Mechanism for Wireless Sensor Network

Shahzad Ashraf,[1] Omar Alfandi,[2] Arshad Ahmad,[3] Asad Masood Khattak,[2] Bashir Hayat,[4] Kyong Hoon Kim,[5] and Ayaz Ullah[6]

[1]*College of Internet of Things Engineering, Hohai University, Changzhou Jiangsu, China*
[2]*College of Technological Innovation at Zayed University, Abu Dhabi, UAE*
[3]*Department of IT & Computer Science, Pak-Austria Fachhochschule: Institute of Applied Sciences and Technology, Mang Khanpur Road, Haripur 22620, Pakistan*
[4]*Institute of Management Sciences, Peshawar, Pakistan*
[5]*School of Computer Science & Engineering, Kyungpook National University, Daegu 41566, Republic of Korea*
[6]*Department of Computer Science, University of Swabi, Anbar 25000, Pakistan*

Correspondence should be addressed to Bashir Hayat; bashir.hayat@imsciences.edu.pk

Received 25 July 2020; Revised 22 September 2020; Accepted 29 October 2020; Published 28 November 2020

Academic Editor: Farman Ullah

Due to unavoidable environmental factors, wireless sensor networks are facing numerous tribulations regarding network coverage. These arose due to the uncouth deployment of the sensor nodes in the wireless coverage area that ultimately degrades the performance and confines the coverage range. In order to enhance the network coverage range, an instance (node) redeployment-based Bodacious-instance Coverage Mechanism (BiCM) is proposed. The proposed mechanism creates new instance positions in the coverage area. It operates in two stages; in the first stage, it locates the intended instance position through the Dissimilitude Enhancement Scheme (DES) and moves the instance to a new position, while the second stage is called the depuration, when the moving distance between the initial and intended instance positions is sagaciously reduced. Further, the variations of various parameters of BiCM such as loudness, pulse emission rate, maximum frequency, grid points, and sensing radius have been explored, and the optimized parameters are identified. The performance metric has been meticulously analyzed through simulation results and is compared with the state-of-the-art Fruit Fly Optimization Algorithm (FOA) and, one step above, the tuned BiCM algorithm in terms of mean coverage rate, computation time, and standard deviation. The coverage range curve for various numbers of iterations and sensor nodes is also presented for the tuned Bodacious-instance Coverage Mechanism (tuned BiCM), BiCM, and FOA. The performance metrics generated by the simulation have vouched for the effectiveness of tuned BiCM as it achieved more coverage range than BiCM and FOA.

1. Introduction

Wireless sensor networks (WSNs) have been widely considered as one of the most important technologies for the twenty-first century. The sensor nodes are deployed to observe the surrounding events for some phenomenon of interest and thereby process the sensed data and transmit it. These sensor nodes are typically smaller in size with inbuilt microcontrollers and radio transceivers. The fundamental issue in observing such an environment is the area coverage that reflects how well the region is being monitored. Coverage is usually defined as a measure of how well and how long the sensors are able to observe the physical space. The quality of coverage in static sensors is significantly affected by the initial deployment location of the sensor nodes [1]. Unfortunately, sensor deployment cannot be performed manually in most applications, for instance, the deployment in disaster areas, harsh environments, and toxic regions. Thus, sensors are usually deployed by scattering them from an aircraft; however, the actual landing position cannot be uniform due to the existence of obstacles like buildings, trees, and wind causing some areas of the sensing region to be denser than

others. Therefore, even if a large number of redundant nodes are deployed, the desired level of coverage still cannot be achieved [2]. Therefore, it is essential to make use of sagacious sensors that can move iteratively to a better location and can achieve the substantial coverage. In order to address the sensing coverage area, it is important to understand the attributes of the sensor node mobility control mechanism. Indeed, the sensor nodes have two types of mobility control attributes, i.e., centralized and distributed. For the centralized attribute, the bunch of nodes is centrally monitored by a sink node that overhears the sensing data from neighboring nodes, while in distributed networks, the sensors are self-controlled [3].

All sensor nodes have limited sensing and communication abilities which make the sensor nodes unable to obtain the entire network information. Due to that, sensors are being deployed randomly and allowed to move and communicate with respective neighbors by exchanging information among them. Miniaturized robotics has overcome some hurdles regarding sensor mobility. Thereby, mobile sensors have the same sensing capability as static sensors and can move freely to correct locations for providing the required coverage [4], but on the other hand, it is not a cost-effective solution. Considering all aforementioned challenges, we were motivated to design a sagacious sensor node deployment strategy which should enhance the coverage area by consuming the confine energy metrics. Considering the pattern of a hybrid sensor network [5], which has the dual mechanism of mobile and static sensors, we have proposed a Bodacious-instance Coverage Mechanism (BiCM) for wireless sensor networks. For this purpose, a BiCM algorithm has been designed which focuses on how to redeploy the sensor nodes to improve the network coverage area in the hybrid WSN environment. It is indeed a cost-effective solution for improving the coverage of unevenly deployed sensor nodes.

Initially, the proposed algorithm presages where the sensor nodes should be moved to while incurring the trivial moving cost. This will only result in a confined moving cost including the accumulated moving distance, total number of moves, and communication rounds. This algorithm can maintain a balance between coverage and resource consumption during the node redeployment process. The BiCM functions in two stages: In the first stage, the intended target positions of the instance (sensor node) are being computed through the Dissimilitude Enhancement Scheme (DES) [6]. The second stage is called the depuration [7], where the instance moving distance is sagaciously reduced; thereby, the final positions are attainable.

The strenuous contributions in regard to the objective of this study are given below.

(1) The proposed BiCM algorithm tends to overcome related issues with the network coverage range by shifting already deployed sensor nodes from previous to new positions

(2) In some cases, it makes substitutions of nodes to adjust the coverage hole

(3) The unnecessary sensor movement is also being monitored to reduce the movement distance between nodes which prevents the wastage of the energy resource

(4) The simulation results generated through MATLAB have vouched for the succulent performance of BiCM and tuned BiCM when compared with previous work such as FOA

(5) The proposed mechanism accomplished the operation in two junctures: During the first juncture, the intended target positions of the sensor node are computed through the Dissimilitude Enhancement Scheme (DES). The second juncture is referred to as depuration, where the moving distance between nodes is sagaciously reduced; thereby, the target positions are achieved

The rest of the findings are structured as follows: The previous work has been rummaged out in Section 2 and the proposed methodology has been explained in Section 3, while Section 4 renders the output performance and the discussion. Finally, overall achievements have been summarized in the form of a conclusion in Section 5.

2. Literature Review

Usually, the sensor nodes are deployed to cover the area between distinct boundaries; however, selection of the most suitable area has remained an ever present challenge. In order to achieve the sufficient coverage area, the distributed deployment strategy is commonly used to improve the coverage interest by moving the sensor nodes from one location to another. For this purpose, the distributed movement algorithms [8] are being used wherein the coverage area is allocated in multiple segments. If any sensor node was unable to detect the event happenings within the deployed segment, no other sensor node can detect it. Eventually, the monitoring of each segment area for the coverage gap (hole) [9] and calculation of a new instance location are the prime liabilities of the deployed sensor node.

All distributed movement algorithms are facing numerous tribulations regarding new instance calculations within the segment area while relocating the new location. No researcher could ever address overcoming the instance reallocation challenge in a hybrid environment. Therefore, no wireless network having coverage holes can successfully carry out its monitoring operation [10]. The researcher tried to incorporate more iterations in their designed model to address the new allocation issue, but it drastically increases the implications and causes higher energy consumption [11].

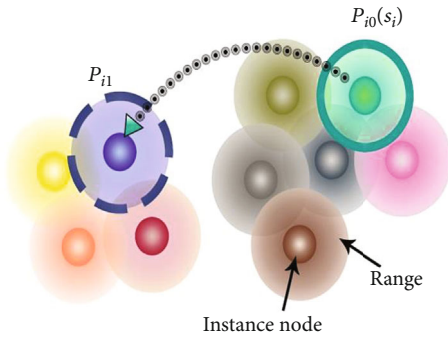
To some extent, numerous researchers have made substantial contributions to avoid such issues, for example, the motion capability of sensor nodes with relocation ability and dealing with sensor failure have been identified by Zhang and Fok [12]; they suggested a two-phase sensor relocation solution. The redundant sensors are first identified and then relocated to the target location. They proposed a grid-quorum solution to locate the closest redundant sensor and then use the cascaded movement to relocate the redundant sensors. In fact, the suggested model could not control the exorbitant energy drainage, and thereby, the entire network might die after the few transmission rounds. On the other

Table 1: Comparative analysis among various algorithms with the proposed BiCM.

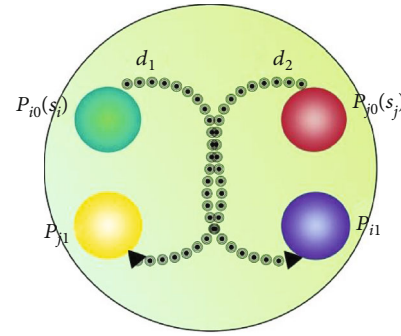
Algorithm	Working ground	Expediency	Impairments	Comparison with proposed BiCM
Genetic algorithm (GA)	Stochastic search methodology through generic system: within a population, it impels the recombination and mutation.	It is faster and has the ability to find the best quality solution in trivial time, possesses parallel capabilities, and easily discovers the global optimum.	It never guarantees an optimal solution. It is hard to choose parameters like number of generations and population size. It is expensive.	It functions in a hybrid environment and ensures relocation of the intended instance position within the coverage area; therefore, energy consumption remains confined.
Particle swarm optimization (PSO)	Inspired by bird flocking and fish schooling: the particles move in a multidimensional search space, and the single intersection of all dimensions forms a particle.	It can overcome the unconstrained minimization issue. Providing the derivative-free technique, it is less sensitive and less dependent on a set of initial points. It can generate high-quality solutions.	It can easily fall into the local optimum in high-dimensional space and has a low convergence rate in the iterative process. It is difficult to adopt the best topology.	At the beginning, it rummages where the sensor nodes should be moved; therefore, local minima can easily be avoided.
Tabu search (TS)	It works on the principle of adaptive memory and responsive exploration.	It has simple implementation and provides robust solution for complex issues.	It vanishes in a local minimum, requires large computing time, and cannot give an upper bound for the computation time	Within a trivial period, it maintains the network coverage range.
Bacterial foraging algorithm (BFA)	It works on search and optimal foraging decision-making capabilities; problems and movement take place either in clockwise or counterclockwise direction.	It is used for unconstrained numerical optimization, having dual movement, i.e., swimming and tumbling called chemotaxis.	It has a weak ability to perceive the environment and is vulnerable to perception of the local extreme; it is hard to deal with complex optimization problems.	As it operates in two stages, thereupon, no vulnerabilities can slow down the performance, and each stage performs independently.
Ant colony optimization (ACO)	Based on social behaviour of the insects, the optimization process is initialized by random solutions.	It allows rapid discovery of good solutions with guaranteed convergence.	It has dependent sequences of random decisions, a complicated theoretical analysis, and uncertain time to convergence.	The depuration technique in second stage reduces the moving distance, and there exists no uncertainty.
Harmony search (HS)	It is based on musical instrument harmony and is a process for better harmony movement.	No setting value is required; it can deal with discrete and continuous variables and can ignore the local optima.	It encounters a high-dimensional multimodal issue, causes unproductive iterations, and has poor local search.	Due to the hybrid environment, the local search is free of being followed by factors; thus, there are no impeaching hurdles.
Artificial bee colony (ABC)	Search optimization consists of three essential components: employed and unemployed foraging bees and food sources.	It minimizes the expense of deploying nodes inside the monitoring region, deals with local solution, and has broad applicability and complex functions.	It has a low process and a higher number of objective function evaluations; number of dimensions might change.	It maintains the network dimension by reducing the moving distance between instance nodes.
Jenga-inspired optimization algorithm (JOA)	Based on greedy fast convergence, it selects the minimum cost node subset through the roulette method and is a bridge between the optimal solution and a short computation time.	It addresses the energy-efficient coverage issues, having stochastic approach to conduct random exploration; if a sensor node cannot cover an area, the other node will avail of the chance.	The detection probability decreases exponentially as the distance becomes greater.	It has shrewd control over the moving distance; therefore, no uncouth movement can degrade the overall communication.

hand, Storn and Price [13] tried to address the coverage and load balancing issues by minimizing the moving distance and argued for a centralized movement solution, based on the Hungarian method. However, the centralized movement technique revealed that those sensor nodes already have appropriate positions when impelled to leave the position creating energy holes.

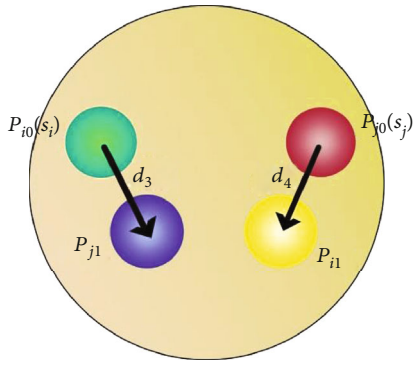
Wang et al. [14] proposed three different distributed movement-assisted sensor deployment algorithms, VEC, VOR, and Minimax, to improve the total area coverage.



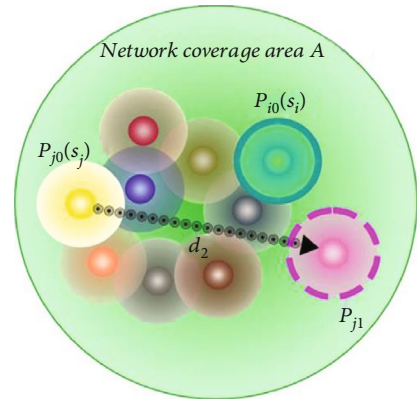
(a) Instance sensor node movement



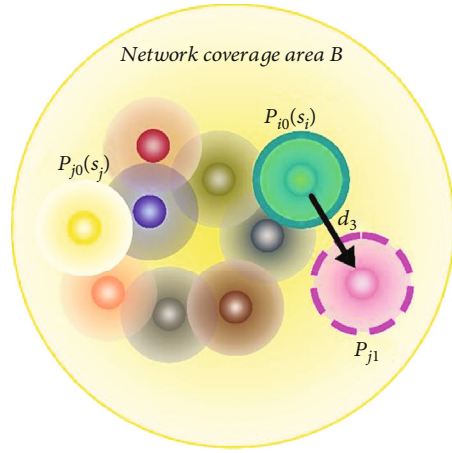
(b) Instance sensor node movement toward intended position



(c) Instance sensor node movement achieving intended position



(d) Moving distance reduction in coverage range A



(e) Moving distance reduction achieved in coverage range B

Figure 1: Instance sensor node movements.

Thereby, they used the Voronoi diagram to partition the monitoring area into n convex polygons where every polygon enclosed one sensor node only. This method utilizes the local polygon information [15], to calculate the new instance location to move the sensor node. The VEC approach uses virtual force between two nodes to push them away from each other at a certain distance. Minimax and VOR algorithms are greedy and try to fix the largest coverage hole by moving the sensor node towards the farthest polygon vertex. The nodes approaching the polygon do not need to move towards the farthest vertex. As a result, this movement may not reduce the coverage hole but might increase the complications.

The identification of a new instance location and its relative computation has been calculated through four local displacement conditions by Mahboubi and Aghdam [16], taking into account the circles having a centered position within the respective polygons. Some centers might lie out of the polygon, and thereby, sensor nodes locating around those circles may not have movement. Consequently, this issue demands more rounds to overcome the coverage tribulation. The more the rounds it demands, the more the resources are being consumed; as a result, the sensor nodes will cause the network to confine the lifespan before the specified time.

In order to increase the coverage rate of sensor nodes, various researchers have proposed different optimization

Table 2: Simulation parameters for BiCM.

Parameter identifiers	Values
Deployment area	$60 \times 60\,m^2$
Number of sensor nodes	60
Grid point	$0.4\,m * 0.4\,m$
Group size	20
Sensing radius	5 m
Maximum iterations	25
Loudness	0.5
Pulse emission rate	0.5
f_{min}	0
f_{max}	2

techniques. A sensing and perception-based Fruit Fly Optimization Algorithm (FOA) [17] was applied by Das et al. to address the position issue of the sensor node which is aimed at enhancing the coverage matter in ideal and obstacle environments. As the fruit flies can reach the food source by using their smell and vision organs, initially, they use osphresis organs to find all kinds of scents in the air. Then, they fly toward the food. When they get close to the food, they use their vision organs to get closer. Similar action is adopted for relocating the sensor positions. Despite its advantages, there are critical issues, for instance, the first pointing location remains poor. Further, the algorithm significantly traps into the local optimum, and the update strategy is limited.

In pursuit of a better coverage technique, a majority of scholars have tried to use intelligent algorithms, like Genetic Algorithm (GA) [18] and Particle Swarm Optimization (PSO) [19], to solve the issue. Though the Fruit Fly Optimization Algorithm is more simple and practicable than GA and PSO, but due to unavoidable limitations, the researchers are still exerting their efforts to develop a shrewder algorithm. Keeping the coverage phenomenon at a high level, Huang et al. [20] introduced a Multiworking Set Alternate Coverage (MWSAC) mechanism that claims to achieve a continuous partial coverage range. The author has achieved a maximum number of working sets by applying a distributed algorithm. The sleep and awakening mechanisms of nodes are adopted which separate the number of active and inactive nodes and keep them synchronous from time to time. Through this method, the nodes appear to work in shifts because the workload has been greatly reduced and the consumption of energy becomes trivial. The authors have however not addressed the false detection occurring in multiworking wireless sensor networks. Table 1 exhibits various comparisons among such algorithms and shows a significant improvement by the proposed algorithm.

3. Coverage Model

A coverage model explains the possible coverage range by the sensor nodes in a coverage area [21]. All sensor nodes have various coverage ranges characterized by area [22], where these sensors are being deployed, the accuracy, the environment factors, and the resolution. The coverage area depends on various factors such as the signal strength generated from the source, distance between the sensor node and the source, and the rate of attenuation in propagation [23]. For example, for an acoustic sensor network establishing the coverage range to detect the mobile vehicles, the sensor nearer to a vehicle can detect higher acoustic signal strength than the one farther away from the vehicle due to signal attenuation, and as a result, there is higher confidence of detecting vehicles [24].

3.1. Problem Formulation. For the proposed coverage model, a two-dimensional coverage area [25] has been considered. Further, the coverage area is divided into various segments each having unit size. When n number of sensor nodes have been deployed in the targeted area m, a full couplet of the sensor node can be defined as given in

$$S = \{S_1, S_2, \cdots..S_n\}. \tag{1}$$

The position of the i^{th} node is defined as $S_i = (x_i, y_i)$ where $i = (1, 2, \cdots n)$. The coverage range of sensor S_i can be expressed as a circle centered at its coordinates (x_i, y_i) with the radius of the sensing range R_s. Let E_i, be a random variable for an event where a sensor node S_i covers an area of segment $A(xA, yA)$. The presage factor for event E_i can be written as $P\{E_i\}$ which is equal to the coverage presage, i.e., $P(S_i, xA, yA)$. Thereupon, the happening of a presage event can be defined by the discrete coverage model expressed in

$$P(S_i, xA, yA) = \begin{cases} 1, d(S_i, xA, yA), & \leq R_s, \\ 0, & \text{other case.} \end{cases} \tag{2}$$

The Euclidean distance [26] of the i^{th} sensor node from segment area $A(x, y)$ can be computed by

$$P(S_i, xA, yA) = \sqrt{(x - x_i)^2 + (y - y_i)^2}. \tag{3}$$

All coverage pints within the coverage range are measured as unity covered by the particular sensor, whereas the points outside of this coverage range are regarded as 0. The shrewd objective of the coverage optimization issue is to provide a sufficient coverage range (CR) [27], by using less number of sensor nodes. The CR is used to estimate the performance of the sensor network. Generally, it is assumed that the segment area point can be covered by any sensor node only once.

3.2. BiCM Model. At present, among all optimization algorithms, the DES is considered as the fastest optimization scheme; therefore, we found it sagacious and were motivated to take full advantage of it for our proposed BiCM algorithm. Thus, the coverage range tribulations in WSN are being resolved by redeployment of sensor nodes through DES strategies, and therefore, the stages of the BiCM design model are explained one by one.

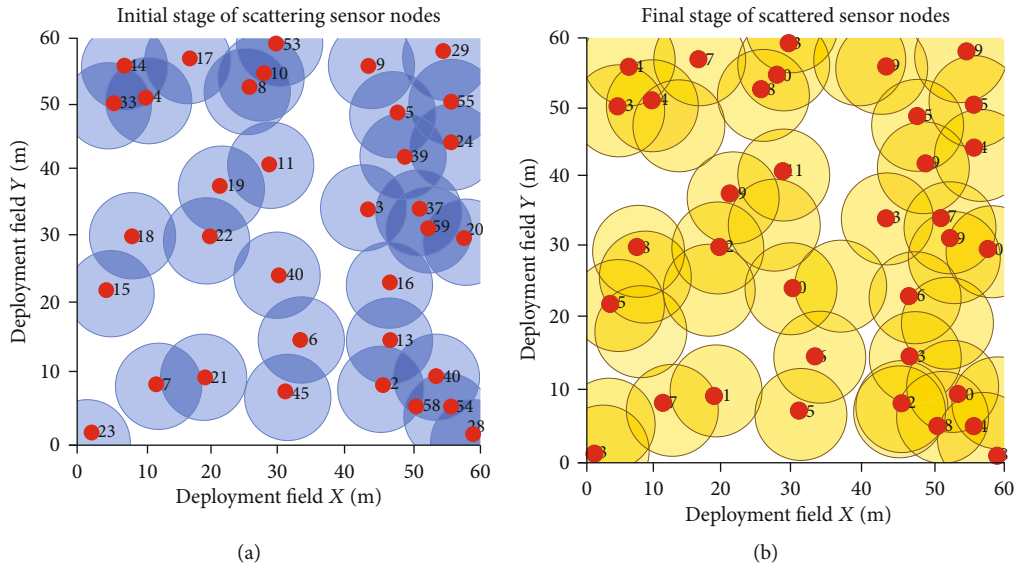


FIGURE 2: (a) The initial and (b) the final FOA sensor node deployment.

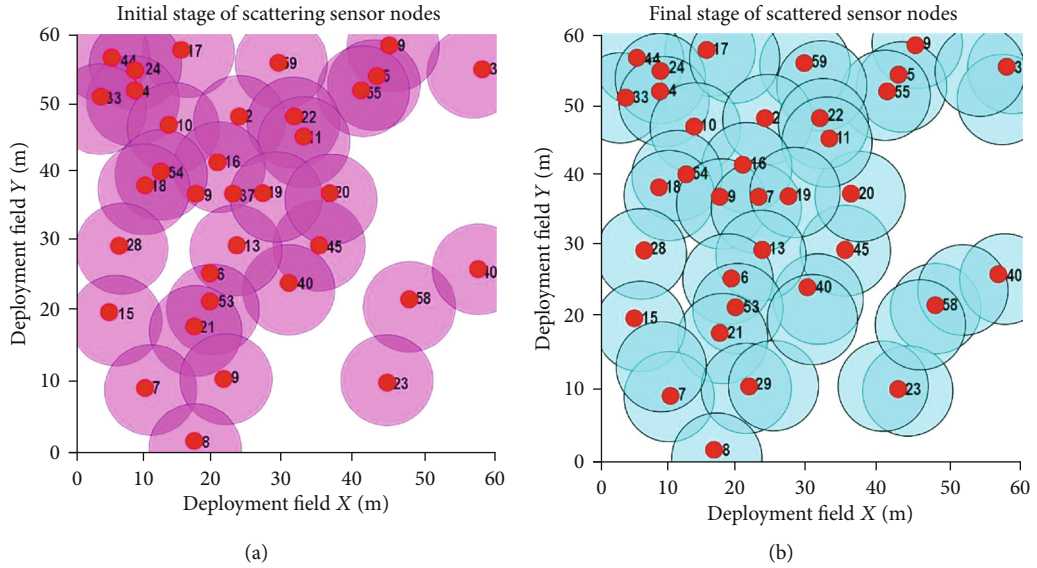


FIGURE 3: (a) The initial and (b) the final deployment of sensor nodes by BiCM.

3.2.1. Stage 1: Locating Intended Target Positions of the Instance. The Bodacious-instance Coverage Mechanism (BiCM) is an investigative search technique that utilizes the shrewd coverage mechanism. It exploits the instance of potential solutions and individuals, to probe the search range. It initializes the parameters while addressing the coverage area issue as depicted in

$$X_i = (x_{i1}, \cdots, x_{ii}, \cdots, x_{iD}), \tag{4}$$

considering $1 \leq i$, as the area range and $x_{ii} \in [\text{ai, bi}]$, where ai and bi denote the lower and upper bounds of the i^{th} node, respectively, and D represents the diameter of the sensor range accompanied with surrounding positions [28]. After every transmission round t, the corresponding reallocation round presages the new expected position of the bodacious instance node which is expressed as

$$V_i(t+1) = X_{\text{bodacious}} + F(X_{r2}(t) - X_{r3}(t)) + F(X_{r4}(t) - X_{r5}(t)). \tag{5}$$

The $X_{\text{bodacious}}$ indicates the appropriate position of the instance while r represents the transmission round and F points to a scaling factor that is a distance control parameter between the initial and the new instance position. To increase the sensing range, the position parameter $V_i(t+1)$

Table 3: Influence of pulse emission rate on coverage rate.

Pulse emission rate (r)	Initial coverage rate (%)	Final coverage rate (%)
0.1	0.8	0.8929
0.2	0.8124	0.905
0.3	0.787	0.9077
0.4	0.8281	0.9041
0.5	0.8097	0.908
0.6	0.8202	0.9025
0.7	0.8208	0.9218
0.8	0.8167	0.9108
0.9	0.8537	0.9354
1	0.8314	0.9153

Table 4: Effect of loudness on coverage rate.

Loudness, A_o (db)	Initial coverage rate (%)	Final coverage rate (%)
0.1	0.8052	0.8931
0.2	0.8375	0.9291
0.3	0.8491	0.9056
0.4	0.8281	0.9107
0.5	0.8276	0.9167
0.6	0.828	0.9219
0.7	0.8273	0.9048
0.8	0.8308	0.9259
0.9	0.8343	0.9281
1	0.8169	0.9179

incorporates the value of predicted instance $X_i(t)$, thereby yielding a temporal position $Q_i\,(t+1)$ as expressed in

$$Q_{i,j}\,(t+1) = \left\{ V_{i,j}(t+1), \text{if (rand } [0,1] < \text{CR or } j = J_{\text{rand}})X_i, j(t), \text{for other case.} \right. \tag{6}$$

The rand (0,1) represents a uniformly distributed random positions, while j_{rand} exhibits randomly predicted positions within the range $[1, D]$. The CR came up as a fractional control parameter $\epsilon[0, 1]$, which shows the inherited characters of previous instance position.

Proceeding towards the final position, the temporal position $Q_i(t+1)$ is being compared with predicted instance $X_i\,(t)$. The newly generated position that possessed a greater fitness metric among the rest of the positions is our intended position of the instance given in

$$X_i(t+1) = \begin{cases} Q_i(t+1), & \text{if } (f(Q_i(t+1)) \geq f(X_i(t))), \\ X_i(t), & \text{other case,} \end{cases} \tag{7}$$

Here, $f(X)$ represents the intended target position of the instance. In fact, the sensor network performs the virtual movement, and as long as it achieves the intended position of the instance sensor in accordance to the Equation (7), physical displacement has been performed accordingly.

Table 5: Effect of $f_{\max}$ on coverage rate.

$f_{\max}$ (f)	Initial coverage rate (%)	Final coverage rate (%)
0.1	0.8492	0.8698
0.2	0.819	0.8433
0.3	0.8135	0.8359
0.4	0.8115	0.8327
0.5	0.831	0.8602
0.6	0.8186	0.8507
0.7	0.8196	0.8414
0.8	0.8211	0.8417
0.9	0.8499	0.8712
1	0.8369	0.8549
1.1	0.8298	0.8888
1.2	0.822	0.9053
1.3	0.8134	0.9331
1.4	0.7965	0.898
1.5	0.8116	0.91
1.6	0.8367	0.9279
1.7	0.8145	0.9169
1.8	0.8267	0.9132
1.9	0.8296	0.9147
2	0.8127	0.9078

3.2.2. Stage 2: Depuration Process. The depuration process is performed to reduce the moving distance of the instance. This will reduce the number of instances (sensor nodes) that need to move, as well as reduce the average moving distance; however, it does not affect the network coverage. The moving distance reduction strategy can be understood as the following: consider the initial positions of an i^{th} instance node s_i is $P_{i0}(x_{i0}, y_{i0})$ and the j^{th} instance node s_j have $P_{j0}(x_{j0}, y_{j0})$. The length of the distance is defined as $d_1 = \overline{|p_{i0}p_{i1}|}$ and $d_2 = \overline{|p_{j0}p_{j1}|}$ and so on. The BiCM algorithm searches the new intended positions of all instance nodes in the coverage area and systematically reduces the number of instance nodes that are needed to be moved. The instance-sensing range may even fully overlap with other instance nodes [29]; these nodes are called redundant nodes and are illustrated in Figure 1(a). The instance sensor node s_i displaces from p_{i0} to p_{i1}; thereby, the coverage rate $R_{\text{area}}(S)$ shows that no substantial change has been recorded which confirms that no movement is required by the s_i instance node. Therefore, the substantial instance nodes can be removed from the queue which eventually decreases the distance.

The position of the instance nodes is being updated by changing the distance position of s_i and s_j that is $d_1 + d_2$ before and after the displacement has been occurred, and it will be updated to $d_3 + d_4$ accordingly as given in

Table 6: Influence of grid points on coverage rate.

Grid points (m ∗ m)	Initial coverage rate (%)	Final coverage rate (%)
0.1 ∗ 0.1	0.8306	0.9203
0.2 ∗ 0.2	0.7975	0.9006
0.3 ∗ 0.3	0.8006	0.9106
0.4 ∗ 0.4	0.8342	0.9132
0.5 ∗ 0.5	0.8012	0.9056
0.6 ∗ 0.6	0.8451	0.9341
0.7 ∗ 0.7	0.8052	0.9125
0.8 ∗ 0.8	0.8135	0.9181
0.9 ∗ 0.9	0.8142	0.9200
1 ∗ 1	0.8240	0.9212

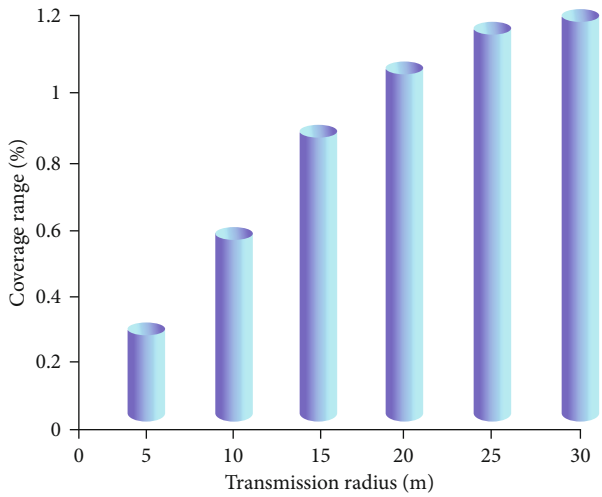


Figure 4: Coverage rate for varying sensing radii of sensor nodes by BiCM.

Figure 1(b). It is worth mentioning that $d_1 + d_2 > d_3 + d_4$; therefore, achieving the intended positions, the moving distance of s_i and s_j can be confined but no change will occur in the coverage area, but the coverage area distance rate will be extended. The instance nodes that are eager to update their moving position will be substituted with the moving position of the nodes which are stationary and do not require further movement. This step can prevent the instance nodes from making unnecessary and longer movement. In this case, the instance node does not possess sufficient energy while reaching the intended position; thereby, other surrounding nodes will surrogate the liability. We should consider Figure 1(c), where instance node s_i does not plan to leave its position while at the same time instance node s_j is eager to shift its position from P_{j0} to P_{j1}. Therefore, the instance node s_i is displaced from P_{i0} to P_{j1} but s_j remains in hiatus. The coverage range $B \geq A$ and $3 < d_2$, instead of sensor node s_j, and the algorithm smartly shifts the instance node s_i to the intended new position of node s_j while keeping the s_j node stationary. This change will not affect the coverage range of the network and does not impel the rest of the instance nodes to move in the queue. Eventually, an average moving distance of the instance node is reduced which enhances the coverage area distance rate. This moving distance reduction is illustrated in Figures 1(d) and 1(e).

4. Simulation Results and Discussion

In order to validate the efficiency of node deployment based on BiCM, the simulation trials are conducted using MATALAB R2016a [30]. The performance among BiCM, tuned BiCM, and FOA is carried out using the simulation setup parameters given in Table 2. To observe the performance of the aforementioned algorithms, nearabout 60 sensor nodes were deployed randomly in the monitoring area of size 60 $\times$ 60 m^2. To demonstrate the performances of FOA, BiCM, and tuned BiCM, the initial and final node deployments are presented in Figures 2 and 3.

These Figures 2 and 3 signify the initial and final node deployments after executing the FOA and BiCM algorithms. Thereupon, it can be clearly understood that node deployment based on BiCM has minimum redundancy and is most uniform compared to node deployment by the FOA mechanism. Table 3 signifies the influence of pulse emission rate (r) on the coverage of sensor nodes. The value of r changes from 0.1 to 1 whereas the value of other instance mechanism parameters such as loudness, maximum frequency, and sensing radius is kept constant to 0.5, 2, and 5, respectively. To beat the effect of arbitrariness [31], the instance mechanism is simulated 50 times, and greatest value of coverage is picked every time. The maximum value of coverage after performing BiCM is attained as 93.54% at a pulse emission rate of 0.9. As instances move towards their respective target (grid points), they emit a greater number of pulses [32]; therefore, the pulse emission rate will be high when sensor nodes move close to the grid points [33]. Thereupon, the value of the pulse emission rate is kept at 0.9. Further, to see the effect of the loudness parameter of the instance mechanism on the coverage rate of sensor nodes, the value of loudness (A_o) is varied from 0.1 to 1 while the pulse emission rate (r) is set to 0.9 and the value of other parameters is 0.5; the sensing radius (r_s) is fixed at 5 meters. Table 4 shows the variations of loudness and initial and final coverage rates of nodes after implementing BiCM. The BiCM is run 50 times, and the best value of the initial and final coverage rates is selected. The coverage rate after executing BiCM is obtained as the highest at about 93.1% at the 0.2 value of loudness. When sensor nodes (instance) get near to the grid point, the intensity of emitted pulses is low; therefore, the loudness parameter should be kept low [34]. Thereupon, the value of the loudness parameter is fixed at 0.2.

In addition to this, Table 5 demonstrates the effect of maximum frequency ($f_{\max}$) [35], on coverage; its value has been changed from 0.1 to 2. The constraints of the instance mechanism for instance pulse emission rate, loudness, and sensing radius are kept constant to 0.9, 0.2, and 5, respectively. For each variation of maximum frequency, the instance mechanism has been executed 50 times and supreme values of coverage before and after the execution

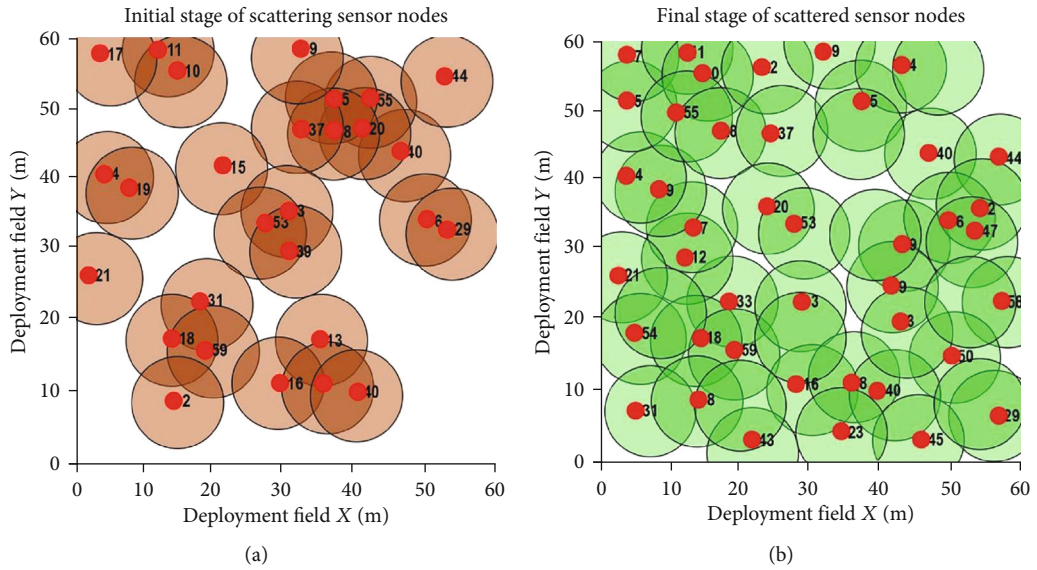


Figure 5: (a) Initial deployment of sensor nodes for tuned BiCM; (b) final deployment of sensor nodes by tuned BiCM.

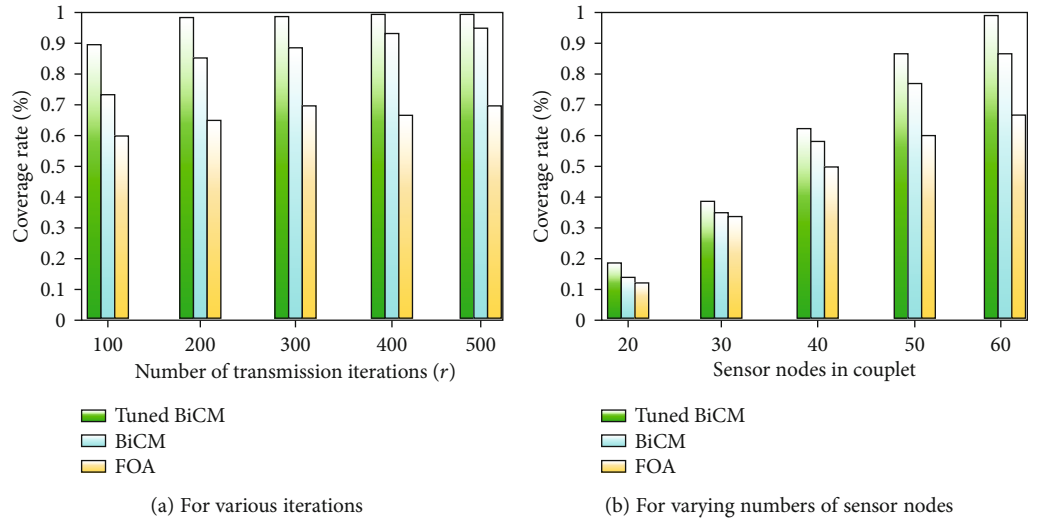


Figure 6: Coverage rate analysis by FOA, BiCM, and tuned BiCM.

Table 7: Deployment results for FOA, BiCM and Tuned BiCM.

Algorithms	FOA		BiCM		Tuned BiCM	
Parameters	Initial results	Final results after execution	Initial results	Final results after execution	Initial results	Final results after execution
Average coverage rate	75.56%	85.16%	82.72%	91.91%	91.54%	98.29%
Standard deviation	0.0286	0.0251	0.0187	0.0126	0.0126	0.0055
Best coverage value	78.92%	87.49%	87.10%	94.30%	93.45%	99.46%
Worst coverage value	68.40%	78.20%	79.38%	90.02%	89.55%	97.31%

of the instance mechanism have been chosen. The best value of coverage after implementing BiCM is 93.31% when $f_{\max}$ is 1.3. Thus, the value of $f_{\max}$ is set to 1.3. To observe the impact of grid points on the coverage rate of nodes, the value of the grid point has varied from 0.1 m ∗ 0.1 m to 1 m ∗ 1 m. The various simulation factors such as pulse emission rate,

Table 8: Comparison of computation time of BiCM, FOA, and tuned BiCM.

Algorithms	FOA	BiCM	Tuned BiCM
Computation time (s)	0.28	0.019	0.016

maximum frequency, sensing radius, and loudness are kept constant at 0.9, 1.3, 5, and 0.2, respectively. In Table 6, every value of grid point BiCM runs 50 times and the uppermost values of the coverage rate have been taken. The highest value of the coverage rate at about 93% is obtained after running the BiCM when grid points were set to 0.6 m ∗ 0.6 m. Further, the sensing radius is varied from 1 m to 10 m. Figure 4 signifies the variations of the coverage rate after applying BiCM w.r.t. changes in the sensing radius of the node. The parameters of BiCM, for example, grid points, loudness, pulse emission rate, and maximum frequency, are set as 0.6 m ∗ 0.6 m, 0.2, 0.9, and 1.3, respectively. It is clear from Figure 4, as the sensing radius has increased, that the coverage rate of sensor nodes is also increased, and its value is 100% when the sensing radius is increased beyond 7 m. But there is a trade-off between the sensing radius and cost: while the sensing radius of the node is increased, the cost of sensor nodes also increased.

The tuned values of various constraints of BiCM such as loudness, maximum frequency, sensing radius, pulse emission rate, and grid points are 0.2, 1.3, 6, 0.9, and 0.6 m ∗ 0.6 m, respectively. To validate the performance of node deployment based on BiCM after setting the above constraint values, the initial and final node deployments after executing the tuned BiCM are shown in Figure 5. Thereupon, it can be obviously seen that node deployment based on tuned BiCM has the lowest redundancy compared with BiCM and FOA. To further demonstrate the effectiveness of tuned BiCM, the coverage rates for the tuned BiCM, BiCM, and FOA for various iterations are shown in Figure 6. The iterations are varied from 0 to 500. The convergence speed of the tuned BiCM is more compared to FOA. The tuned BiCM converged around 150 iterations, whereas FOA converges around 350 iterations due to exploitation characteristics of the instances.

The tuned BiCM has achieved a higher coverage rate at about 99.46% compared to 93.37% and 88.33% of BiCM and FOA, respectively. In order to overwhelm the effect of randomness of tuned BiCM, instance mechanism optimization and Fruit Fly Optimization Algorithms are run 15 times. The deployment results in terms of average coverage rate, standard deviation, and best and worst coverage values for tuned BiCM and FOA are represented in Table 7. It can be obviously seen from Table 7 that tuned BiCM has achieved the average coverage rate of about 98.29% compared to 91.91% and 85.16% of BiCM and the Fruit Fly Optimization Algorithm. Further, the standard deviation for node deployment based on tuned BiCM is lowest, so tuned BiCM is more stable compared to FOA and BiCM. The best and worst coverage values for tuned BiCM are 99.46% and 97.31% compared to 94.30% and 90.02% and 87.49% and 78.20% for the BiCM- and FOA-based node deployments, respectively.

Further, the comparison of tuned BiCM, BiCM, and FOA in terms of computation time is represented in Table 8. The computation time for tuned BiCM is less, i.e., 0.016 seconds, compared to 0.019 seconds and 0.28 seconds for BiCM and FOA, respectively. The tuned BiCM and BiCM converge at 25 iterations whereas FOA converged at 500 iterations; therefore, the speeds of tuned BiCM and BiCM are more and converge faster at an earlier stage because of their exploitation feature compared to the Fruit Fly Optimization Algorithm.

5. Conclusion

In order to enhance the coverage rate of the sensor nodes, an innovative sensor deployment technique based on Bodacious-instance Coverage Mechanism (BiCM) has been purposed that accomplished the desired goal with limited energy consumption. The analysis of various factors of BiCM such as loudness, grid points, emission rate and radius of nodes, and frequency has been identified, and shrewd values of the above parameters are discovered. Node deployment based on tuned BiCM and BiCM shows that both algorithms converge at an earlier stage compared to the Fruit Fly Optimization Algorithm. The simulation results demonstrate that tuned BiCM has attained a mean coverage rate of about 98.29% which is higher compared to FOA and BiCM. Further, various simulations have been done by varying the number of sensor nodes and iterations, and a coverage rate curve is plotted for tuned BiCM, BiCM, and FOA. The comparison of the computation time is also represented in this paper. Tuned BiCM has a high coverage rate and less computation time compared to FOA and BiCM. In the future, the various evolutionary optimization algorithms can be applied to the node deployment problem to increase the coverage rate of sensor nodes.

Data Availability

The data to support the findings of this study is available inside the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by Zayed University Research Fund # R19046.

References

[1] M. Abazeed, N. Faisal, S. Zubair, and A. Ali, "Routing protocols for wireless multimedia sensor network: a survey," *Journal of Sensors*, vol. 2013, 11 pages, 2013.

[2] S. Ashraf, M. Gao, Z. Chen, S. Kamran, and Z. Raza, "Efficient node monitoring mechanism in WSN using ContikiMAC protocol," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 11, 2017.

[3] F. Ait Aoudia, M. Gautier, M. Magno, O. Berder, and L. Benini, "A generic framework for modeling MAC protocols in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 3, pp. 1489–1500, 2017.

[4] S. Ashraf, M. Gao, Z. Mingchen, T. Ahmed, A. Raza, and H. Naeem, "USPF: underwater shrewd packet flooding mechanism through surrogate holding time," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 9625974, 12 pages, 2020.

[5] M. Li, X. Du, X. Liu, and C. Li, "Shortest path routing protocol based on the vertical angle for underwater acoustic networks," *Journal of Sensors*, vol. 2019, Article ID 9145675, 14 pages, 2019.

[6] S. Ashraf, T. Ahmed, A. Raza, and H. Naeem, "Design of shrewd underwater routing synergy using porous energy shells," *Smart Cities*, vol. 3, no. 1, pp. 74–92, 2020.

[7] M. S. Aliyu, A. H. Abdullah, H. Chizari, T. Sabbah, and A. Altameem, "Coverage enhancement algorithms for distributed mobile sensors deployment in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 12, no. 3, Article ID 9169236, 2016.

[8] S. Ashraf, Z. Aslam, A. Yahya, and A. Tahir, "Underwater routing protocols analysis of intrepid link selection mechanism, challenges and strategies," *International Journal of Scientific Research in Computer Science and Engineering*, vol. 8, no. 2, pp. 1–9, 2020.

[9] S. Ashraf and T. Ahmed, "Machine learning shrewd approach for an imbalanced dataset conversion samples," *Journal of Engineering and Technology (JET)*, vol. 11, no. 1, 2020.

[10] S. Balsamo, A. Marin, and E. Vicario, Eds., *New Frontiers in Quantitative Methods in Informatics: 7th Workshop, InfQ 2017, Venice, Italy, December 4, 2017, Revised Selected Papers*, Springer, New York, NY, 1st edition edition, 2018.

[11] J. Wang, Y. Gao, C. Zhou, R. Simon Sherratt, and L. Wang, "Optimal coverage multi-path scheduling scheme with multiple mobile sinks for WSNs," *Computers, Materials & Continua*, vol. 62, no. 2, pp. 695–711, 2020.

[12] Q. Zhang and M. Fok, "A two-phase coverage-enhancing algorithm for hybrid wireless sensor networks," *Sensors*, vol. 17, no. 12, p. 117, 2017.

[13] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[14] G. Wang, G. Cao, and T. F. la Porta, "Movement-assisted sensor deployment," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 640–652, 2006.

[15] J. Wang, Y. Yang, T. Wang, R. S. Sherratt, and J. Zhang, "Big Data Service Architecture: A Survey," *Journal of Internet Technology*, vol. 21, no. 2, 2020.

[16] H. Mahboubi and A. G. Aghdam, "Distributed deployment algorithms for coverage improvement in a network of wireless mobile sensors: relocation by virtual force," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 4, pp. 736–748, 2017.

[17] S. Das, A. Biswas, S. Dasgupta, and A. Abraham, "Bacterial foraging optimization algorithm: theoretical foundations, analysis, and applications," in *Foundations of Computational Intelligence Volume 3: Global Optimization*, A. Abraham, A.-E. Hassanien, P. Siarry, and A. Engelbrecht, Eds., pp. 23–55, Springer, Berlin, Heidelberg, 2009.

[18] H. Stringer, *Behavior of variable-length genetic algorithms under random selection*, University of Central Florida, 2007.

[19] L. Sun, X. Song, and T. Chen, "An improved convergence particle swarm optimization algorithm with random sampling of control parameters," *Journal of Control Science and Engineering*, vol. 2019, Article ID 7478498, 11 pages, 2019.

[20] M. Huang, A. Liu, M. Zhao, and T. Wang, "Multi working sets alternate covering scheme for continuous partial coverage in WSNs," *Peer-to-Peer Networking and Applications*, vol. 12, no. 3, pp. 553–567, 2019.

[21] S. Ashraf, Z. A. Arfeen, M. A. Khan, and T. Ahmed, "SLM-OJ: surrogate learning mechanism during outbreak juncture," *International Journal for Modern Trends in Science and Technology*, vol. 6, no. 5, pp. 162–167, 2020.

[22] J. Wang, X. Gu, W. Liu, A. K. Sangaiah, and H.-J. Kim, "An empower Hamilton loop based data collection algorithm with mobile agent for WSNs," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, 2019.

[23] S. Goyal and M. S. Patterh, "Flower pollination algorithm based localization of wireless sensor network," in *2015 2nd International Conference on Recent Advances in Engineering & Computational Sciences (RAECS)*, Chandigarh, India, Dec. 2015.

[24] J. Wang, Y. Gao, W. Liu, and W. Wu and Se-Jung Lim, "An asynchronous clustering and mobile data gathering schema based on timer mechanism in wireless sensor networks," *Computers, Materials & Continua*, vol. 58, no. 3, pp. 711–725, 2019.

[25] S. Ashraf, D. Muhammad, and Z. Aslam, "Analyzing challenging aspects of IPv6 over IPv4," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 6, no. 1, pp. 54–67, 2020.

[26] "How to calculate Euclidean distance," May 2020, https://sciencing.com/how-to-calculate-euclidean-distance-12751761.html.

[27] Yourim Yoon and Yong-Hyuk Kim, "An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks," *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1473–1483, 2013.

[28] S. Ashraf, D. Muhammad, M. Shuaeeb, and Z. Aslam, "Development of shrewd cosmetology model through fuzzy logic," *Journal of Research in Engineering and Applied Sciences*, vol. 5, no. 3, pp. 93–99, 2020.

[29] S. Ashraf, A. Raza, Z. Aslam, H. Naeem, and T. Ahmed, "Underwater resurrection routing synergy using astucious energy pods," *Journal of Robotics and Control (JRC)*, vol. 1, no. 5, 2020.

[30] J. Zhang, Y. Lei, C. Chen, and F. Lin, "Directional probability perceived nodes deployment based on particle swarm optimization," *International Journal of Distributed Sensor Networks*, vol. 12, no. 4, Article ID 2046392, 2016.

[31] A. Shahzad and A. Tauqeer, "Dual-nature biometric recognition epitome," *Trends in Computer Science and Information Technology*, vol. 5, no. 1, pp. 8–14, 2020.

[32] S. Ashraf, T. Ahmed, S. Saleem, and Z. Aslam, "Diverging mysterious in green supply chain management," *Oriental journal of computer science and technology*, vol. 13, no. 1, pp. 22–28, 2020.

[33] J. E. Franklin and R. J. Urick, "A binary detection model for at-sea sonar prediction," *The Journal of the Acoustical Society of America*, vol. 66, no. S1, p. S15, 1979.

[34] S. Ashraf, S. Saleem, A. H. Chohan, Z. Aslam, and A. Raza, "Challenging strategic trends in green supply chain management," *Journal of Research in Engineering and Applied Sciences*, vol. 5, no. 2, pp. 71–74, 2020.

[35] S. Ashraf, A. Ahmad, A. Yahya, T. Ahmed, and 3 Dow University of Health Sciences Karachi Pakistan, "Underwater routing protocols: analysis of link selection challenges," *AIMS Electronics and Electrical Engineering*, vol. 4, no. 3, pp. 234–248, 2020.