Iran Journal of Computer Science (2021) 4:171–183
https://doi.org/10.1007/s42044-020-00075-x

ORIGINAL ARTICLE

NRSM: node redeployment shrewd mechanism for wireless sensor network

Shahzad Ashraf[1] · Tauqeer Ahmed[1] · Sehrish Saleem[2]

Received: 11 May 2020 / Accepted: 5 October 2020 / Published online: 31 October 2020

Abstract
Despite numerous advantages, the challenges for wireless sensor communication always remains open due to which a continuous effort is being applied to tackle the unavoidable conditions regarding wireless network coverage. Somehow, the uncouth deployment of the sensor nodes is making the tribulation queue longer day by day which eventually has great impact over sensor coverage range. To address the issues related with network coverage and uncouth energy wastage, a sensor node redeployment-based shrewd mechanism (NRSM) has been proposed where new intended positions for sensor node are rummaged out in the coverage area. The proposed algorithm operates in two phases; in first phase it locates the intended node positions through Dissimilitude Enhancement Scheme (DES) and moves the node to new position. While second phase is called a Depuration, when the moving distance between initial and intended node position is shrewdly reduced. Further, different variation factors of NRSM such as loudness, pulse emission rate, maximum frequency, and sensing radius have been explored and related optimized parameters are identified. The performance metric has been meticulously analyzed through simulation rounds in Matlab and compared with state of art algorithms like Fruit Fly Optimization Algorithm (FOA), Jenga-inspired optimization algorithm (JOA) and Bacterial Foraging Algorithm (BFA) in terms of mean coverage range, computation time, standard deviation and network energy diminution. The performance metrics vouches the effectiveness of the proposed algorithm as compared to the FOA, JOA and BFA.

Keywords Node deployment · Shrewd coverage · Position · Depuration · Emission · Diminution

1 Introduction

In the era of wireless communication, the sensor network extensively prevailing its dominance and providing chances to the researcher to explore and discover the diversification in this field. In a Wireless Sensor Network (WSN) the sensor nodes are deployed to observe the surroundings events for some phenomenon of interest, process the sensed data and transmit it [1]. These sensor nodes are typically small in size with inbuilt micro-controllers and radio transceivers [2]. The fundamental issue in observing such environments is the area coverage which reflects how well the region is monitored. Coverage is usually defined as a measure of how well and how long the sensors are able to observe the physical space [3].

The quality of coverage in static sensor is significantly affected by the initial deployment location of the sensors. Unfortunately, sensor deployment cannot be performed manually in most applications [4], for instance, the deployment in disaster areas, harsh environments, and toxic regions. Most of the previous studies showed that, sensors were usually deployed by scattering from an aircraft; however, the actual landing position cannot be uniform due to the existence of obstacles for instance, 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. Therefore, it is essential to make use of sensors, which can move iteratively to a better location that can give the required coverage [5]. To address the sensing coverage area, it is important to understand the mobility control attribute [6], of the sensor nodes. Indeed, sensor nodes have two type of mobility control attributes i.e., centralized and

✉ Shahzad Ashraf
nfc.iet@hotmail.com

1 College of Internet of Things Engineering, Hohai University, Changzhou, China

2 Muhammad Nawaz Sharif University of Engineering and Technology, Multan, Pakistan

distributed [7]. Regarding centralized attribute, the bunch of nodes are centrally monitored by a sink node that overhears the sensing data from neighbouring nodes while in distributed networks, the sensors are self-controlled.

All sensor nodes have limited sensing and communication abilities [8], which make the sensor nodes unable to obtain the entire network information [9]. Due to which sensors are deployed randomly and allowed to move and communicate with their neighbours by exchanging information between them. The miniaturized robotics have overcome some hurdles regarding sensors mobility. Thereby, mobile sensors have the same sensing capability as static sensors and can move freely to correct locations for providing the required coverage. On the other hand, it is not a cost-effective solution.

Keeping all aforementioned challenges, it is motivated to design a sagacious sensor node deployment strategy which should enhance the coverage area by consuming just confine energy metrics. Considering the pattern of a hybrid sensor network [10], which composed of mobile and static sensors we have proposed a Node Redeployment Shrewd Mechanism for Wireless Sensor Network (NRSM). For this purpose, a NRSM algorithm has been designed which focus how to redeploy the sensor nodes to improve network area coverage in hybrid WSNs environment. It is indeed a cost-effective solution towards improving coverage with unevenly deployed sensors.

Initially, algorithm aims to determine where the sensor nodes should be moved while incurring the trivial moving cost. This will result only a confine moving cost including the accumulated moving distance, total number of moves, and communication rounds. The proposed NRSM mechanism ultimately can maintain a balance between coverage with confine resource consumption during node redeployment process.

1.1 Understanding NRSM information flow mechanism

Initially, the nodes are deployed with some random positions [11], with certain velocities to search the shrewd target positions in network coverage area. The minimum distance value and related coordinates are being recorded. After getting best minimum distance value the intended positions are crosschecked otherwise process will be repeat the same step. The further proceedings are explained step by step through Flow chart shown as Fig. 1.

Step 1: Initialize all the parameters including the group size (n), the maximum number of iterations and the initial positions of sensor node group ($X_{initial}$, $Y_{initial}$), step length [12], number of area range points, loudness and pulse rate, minimum and maximum frequency, upper and

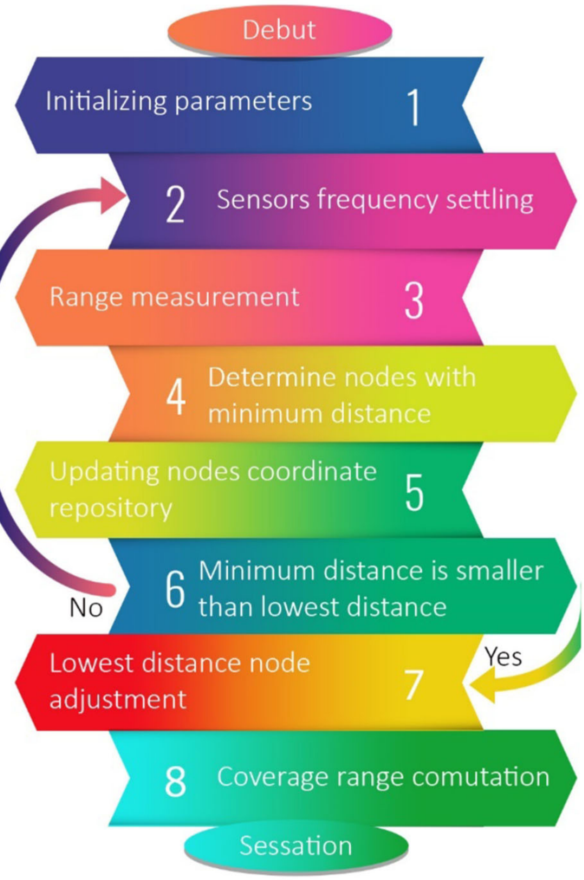


Fig. 1 NRSM information flow chart

lower bounds [13]. All these parameters are being calculated through Eqs. (1, 2)

$$X_{initial}(i) = LB + (UB - LB) * Randomvalue \tag{1}$$

$$Y_{initial}(i) = LB + (UB - LB) * Randomvalue \tag{2}$$

where i varies from 1 to n, LB and UB is lower and upper bounds, and n is the size of sensor node group.

Step 2: The essential parameters of the sensor nodes like positions (x_i^t), velocities (v_i^t) and frequencies for time t are updated as expressed in Eqs. (3–5), where β is an arbitrary vector whose value is lies between 0 and 1, the f_{max} represents maximum frequency and x^* indicate the Shrewd solution.

$$f_i = f_{min} + (f_{max} - f_{min})\beta \tag{3}$$

$$V_i^t = V_i^{t-1} + \left(x_i^t - x^*\right) f_i \tag{4}$$

$$x_i^t = x_i^{t-1} + v_i^t \tag{5}$$

Step 3: The distance of all the sensor nodes from the current area position ($Dist_{n*m}$) [14], is being computed by the Eq. (6)

$$Dist_{n*m} = \sqrt{(X_{n*m} - x_j)^2 + (Y_{n*m} - y_j)^2}, \quad (6)$$

where $X_{\mathrm{n*m}}$ and $Y_{\mathrm{n*m}}$ are initial positions of $n*m$ sensor nodes x_j and y_j are coordinates of j area range.

Step 4: Any sensor node having minimum distance value to the intended node positions are compared and this moving distance is selected.

Step 5: The lowest distance value and related coordinates are recorded in corpus table.

Step 6: The lowest and shrewd distance value is compared with other distance value during every iteration. If no other shrewd distance value is found then this lowest and shrewd value and its coordinates are updated and sensor node shift its position to the intended target in accordance to the condition defined in NRSM algorithm otherwise, repeats step from 2 to 5.

Step 7: The overall network Coverage Range (*CR*) has been computed through Eq. (7). The *M**N* is the network coverage area, m*n represents total summation points of the each sensor node.

$$CR = \frac{N_{m*n}}{M * N} \quad (7)$$

The overall illustration and mechanism have been explained in Sect. 3.

The main contributions have been incorporated into previous finding are listed as below:

- The proposed NRSM algorithm tends to overcome related issues with the network coverage range by shifting already deployed sensor nodes from previous to new positions.
- In some cases, it makes substitution of nodes to adjust the coverage hole.
- The unnecessary sensor movement is also being monitor to reduce the movement distance between nodes which prevents the wastage of the energy resource.
- The simulation results generated through Matlab has vouched the the succulent performance of NRSM when compared with previous work FOA, JOA and BFA.
- The proposed NRSM algorithm accomplished the operation in two junctures, during first juncture the intended target positions of the sensor node is computed through Dissimilitude Enhancement Scheme (DES). The second juncture is referred as Depuration, where the moving distance between nodes is sagaciously reduced, thereby the target positions are achieved.

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

2 Literature review

Usually sensor nodes are deployed to cover the area between distinct boundaries. However, selection of most suitable area is ever remained a challenge. To achieve the sufficient coverage area, the distributed deployment strategy is commonly used to improve the area coverage by moving the sensor nodes from one location to another. For this purpose, the distributed movement algorithms 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 coverage gape (hole) and calculation of new instance location is the prime liability of the deployed sensor node.

All distributed movement algorithms are facing numerous tribulations regarding new instance calculation within the segment area while relocating the new location. No researcher could ever address to overcome the instance reallocation challenge in hybrid environment. Therefore, no wireless network having coverage holes, can successfully carry out its monitoring operation [15]. The researcher tried to incorporate more iterations in their designed model to address the new allocation issue but it drastically increased the implications and causing higher energy consumption.

To some extent, overcome these issues the numerous researchers have made substantial contributions. For example, the motion capability of sensor nodes with relocating ability and dealing with sensor failure have been identified by Qingguo et al. [16]. 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 proposed to use cascaded movement to relocate the redundant sensor. Their suggested model could not control the exorbitant energy drainage and thereby whole network might die after few transmission rounds. On the other hand, Li Jun et al. [17], tried to address the coverage and load balancing issues by minimizing the moving distance and argued a centralized movement solution, based on the Hungarian method. However, the centralized movement technique revealed those sensor nodes having already appropriate positions when impelled to leave the position creating energy holes.

Wang et al. [18], proposed three different distributed movement assisted sensor deployment algorithms, VEC, VOR, and Minimax, to improve the total area coverage. 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 to calculate the new instance location to move 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 sensor node towards the farthest polygon vertex. The nodes approaching to the polygon do not need to move towards the farthest vertex. As a result, this movement may not reduce coverage hole, but might increases the complications.

The identification of new instance location and its relative computation has been calculated through four local displacement conditions by the H. Mahboubi et al. [19], taking into account the circles having 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.

To increase the coverage rate of sensor nodes, various researchers have proposed different optimization techniques. A sensing and perception based Fruit Fly Optimization Algorithm (FOA) was applied by Wen-Tsao Pan [20], to address the position issue of the sensor node which aims to enhance the coverage matter in ideal and obstacle environment. As the fruit flies can reach the food source using their smell and vision organs. Initially, they use osphresis organs to find all kinds of scents in the air. Then they fly toward to food. When they get close to the food, they use their vision organs to get closer. Similar action is adopted for relocating the sensors positions. Despite its advantages, there are critical issues for instance, the first pointing location remains poor. Further, the algorithm significantly traps into local optimum and the update strategy is limited.

An Edge Based Centroid (EBC) algorithm is proposed by Muhammad Sirajo et al. [21], and author claims about enhanced area coverage of monitoring field with minimal energy consumption. This algorithm is based on Voronoi diagram that partitions the sensing field into polygons with one sensor node each to monitor any event in its respective subregion. The sensor node moves to new location at the center of each polygon from location, which improves area coverage. This algorithm depends on certain group of rules that ensures about the center of each polygon before the movement and thereupon the ratio of energy consumption can be lowered. Though this algorithm works smooth but no control over the uncouth movement of the node is addressed due to that sometime a node can make unusual and large displacement which might cause the energy wastage.

Y. Sun et al. [22], presented the idea of an ant based service aware routing mechanism called ASAR. It's a quality service based approach referred as RDS where (*i*) R defines deals the delay and error related issues which requires short bandwidth and high Signal to Noise Ratio (SNR). While (*ii*) D, controls the congestion issue in high SNR. The last (*iii*) S, address the stream query service where low SNR and trivial data traffic-related problems are handled. There are generic shortcomings such as no any central mechanism is given to maintain the optimum and alternate path during overloading. Further, it is a cluster-based approach which deals only routing issues between cluster head and the sink node and efficiency is dependent on service quality which does not remain same.

In pursuit of a better coverage technique, a majority of scholars have tried to use intelligent algorithms like, Genetic Algorithm (GA) [23], and Particle Swarm Optimization (PSO) [24], to solve the issue. Though, fruit fly algorithm is simple and practicable than GA and PSO but due to unavoidable limitations the researcher is still exerting their efforts to develop shrewder algorithm. Table 1, exhibits various comparison 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 coverage area. All sensor nodes have various coverage range characterized by area where these sensors are being deployed, the accuracy, the environment factors and resolution. The coverage area depends on various factors such as the signal strength generated from the source, distance between the sensor node and source and the rate of attenuation in propagation. For example, 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.

3.1 Problem formulation

For proposed coverage model, a two-dimensional coverage area 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 targeted area

Table 1 Comparative analysis among various algorithm with proposed NRSM

Algorithm	Working ground	Expediency	Impairments	Comparison with proposed NRSM
Genetic algorithm (GA)	Stochastic search methodology through generic system, within a population it impels the recombination and mutation	It is faster and have ability to find best quality solution in trivial time, possessed parallel capabilities. Easily discovers the global optimum	Never guarantee for optimal solution. Hard to choose parameters like number of generations, population size. It is expensive	It functions in hybrid environment, ensures about relocating 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 single intersection of all dimensions forms a particle	It can overcome the unconstrained minimization issue. Provides the derivative free technique, it is less sensitivity, less dependent of a set of initial points. It can generate high-quality solutions	It can easily fall into local optimum in high-dimensional space and has a low convergence rate in the iterative process. 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	Simple implementation, provides robust solution for complex issues	Vanish in a local minimum, requires large computing time, 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, movement take place either in clockwise or counter clockwise direction	Used for unconstrained numerical optimization, having dual movement i.e., swimming and tumbling called chemotaxis	Having weak ability to perceive the environment and vulnerable to perception of local extreme, hard to deal with complex optimization problems	As it operates in two stages, thereupon no vulnerabilities can slow down the performance, each stage performs independently
Ant colony optimization (ACO)	Based on social behaviour of the insects, the optimization process is initialized by random solutions	Rapid discovery of good solutions with guaranteed convergence	Dependent sequences of random decisions, having complicated theoretical analysis, uncertain time to convergence	The Depuration technique in second stage reduce the moving distance and there exists no uncertainty
Harmony search (HS)	Based on musical instrument harmony, process for better harmony movement	No setting value is required, can deal with discrete and continuous variables, can ignore the local optima	Encounters high dimensional multimodal issue, causes unproductive iterations, poor local search	Due to hybrid environment, the local search is free of followed by factors, thus 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, having broad applicability, complex functions	Slow process, higher number of objective function evaluation, 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, select minimum cost node subset through the roulette method, bridge between optimal solution and short computation time	Address the Energy-Efficient Coverage issues, having stochastic approach to conduct random exploration, if sensor node cannot cover an area the other node take avail the chance	The detection probability decreases exponentially as the distance becomes greater	Have shrewd control over moving distance therefore no uncouth movement can degrade the overall communication

m, thereby a full couplet of sensor node can be defined as given in Eq. (8):

$$S = \{S_1, S_2, \ldots..S_n\} \quad (8)$$

The position of i^{th} node is defined as $S_i = (x_i, y_i)$ where $i = (1, 2, \ldots n)$. 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, being a random variable for an event that 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 an event presage can be defined by the discrete coverage model expressed in Eq. (9)

$$P(Si, xA, yA) = \begin{cases} 1, d(Si, xA, yA) \le R_s \\ 0, \text{ other case} \end{cases} \quad (9)$$

The Euclidean distance [25], of *ith* sensor node from segment area A(*x,y*) can be computed by Eq. (10).

$$P(Si, xA, yA) = \sqrt{(x - x_i)^2 + (y - y_i)^2} \quad (10)$$

All coverage pints within the coverage range are measured as unity covered by the particular sensor whereas, the points outside of this coverage range is regarded as 0. The shrewd objective of coverage optimization issue is to provide sufficient Coverage Range (*CR*) [26], using less number of sensor nodes. The *CR* is used to estimate the performance of sensor network. Generally, It is assumed that segment area point can be covered by any sensor node only once.

3.2 The proposed NRSM model

At present, among all optimization algorithms the DES [27], is considered as a fasted optimization scheme therefore we found it sagacious and motivated to take full advantage for our proposed NRSM algorithm. Thus, the coverage range tribulations in WSN is being resolved by redeployment of sensor nodes through DES strategies and therefore the stages of NRSM design model are being explained one by one.

3.2.1 Stage 1. Locating intended target positions of the instance

The bodacious-instance coverage mechanism (NRSM) is an investigative search technique that utilizes the shrewd coverage mechanism. It exploits the instance of potential solutions, individuals, to probe the search range. It initialize the parameters while addressing the coverage area issue as depicted in Eq. (11),

$$X_i = (x_{i1}, \ldots, x_{ii}, \ldots, x_{iD}) \quad (11)$$

considering $1 \le i$, as the area range and $\in x_{ii}\,[a_i, b_i]$, where *ai* and *bi* denotes the lower and upper bound of the i^{th} node, respectively and D represents the diameter of the sensor range accompanied with surrounding positions. After every transmission round t, the corresponding re-allocation round presages the new expected position of the bodacious instance node which is expressed as Eq. (12)

$$V_i(t+1) = X_{bodacious} + F(X_{r2}(t) - X_{r3}(t)) + F(X_{r4}(t) - X_{r5}(t)) \quad (12)$$

.

The X_{shrewd} indicates the appropriate position of the node while r represents the transmission round and F points a scaling factor that is a distance control parameter between initial and the new instance position. To increase the sensing range, the position parameter $V_i(t+1)$ is incorporate the value of predicted instance $X_i(t)$, thereby yields a temporal position $Q_i(t+1)$ as expressed in Eq. (13)

$$Q_{i,j}(t+1) = \{V_{i,j}(t+1), if(rand[0,1] < FCRorj = J_{rand}) \\ X_i, j(t), \text{ for other case} \quad (13)$$

The rand(0,1) represents a uniformly distributed random positions, while j_{rand} exhibits randomly predicted positions within the range [1, D]. The FCR came up as a Fractional Control Parameter ∈ [0, 1], which shows the inherited characters of previous instance position.

Proceeding towards final position, the temporal position $Q_i(t+1)$ is being compared with predicted instance $X_i(t)$. The newly generated position that possessed greater fitness metric among rest of the positions is our intended position of the instance given in Eq. (14)

$$X_i(t+1) = \begin{cases} Q_i(t+1), if(f(Q_i(t+1)) \ge f(X_i(t))), \\ X_i(t), \text{ other case} \end{cases} \quad (14)$$

here $f(X)$ represnets the intended target position of the instance. The sensor network performs the virtual movement and as long it achieves the intended position of the instance sensor in accordance to the Eq. 14, physical displacement has been performed accordingly.

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 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 understand as: Consider the initial positions of an *ith* instance node s_i is $P_{i0}(x_{i0}, y_{i0})$ and the *jth* instance node s_j

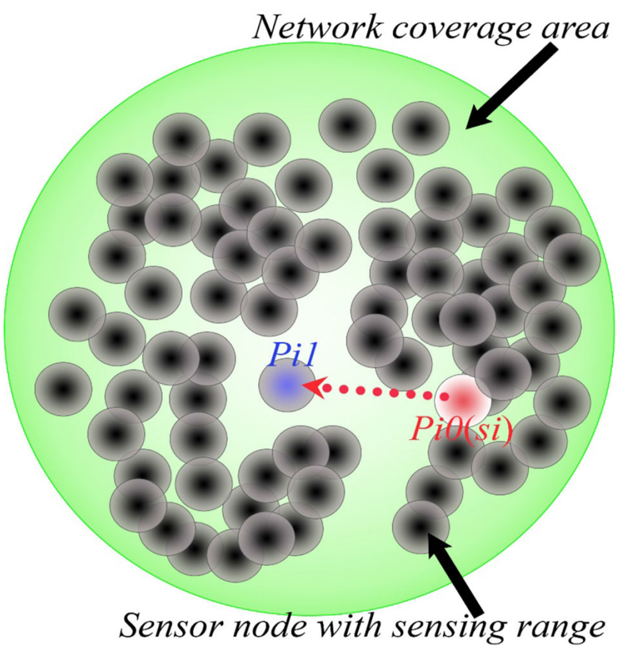


Fig. 2 NRSM network topology

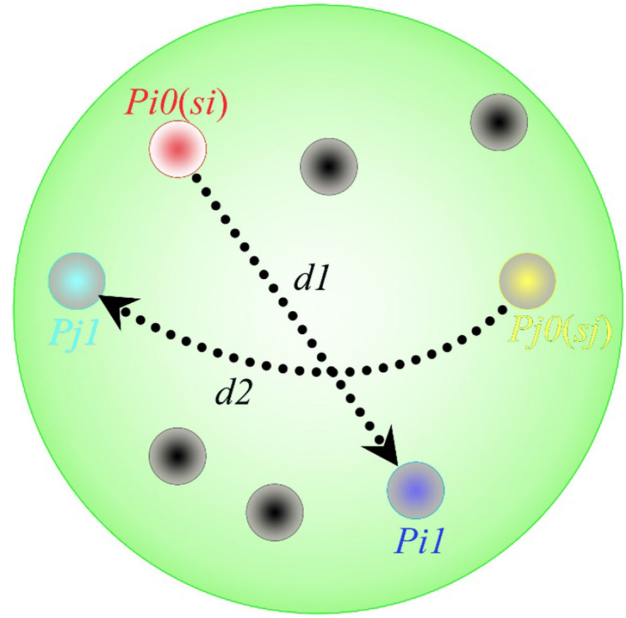


Fig. 3 Sensor nodes intended positions

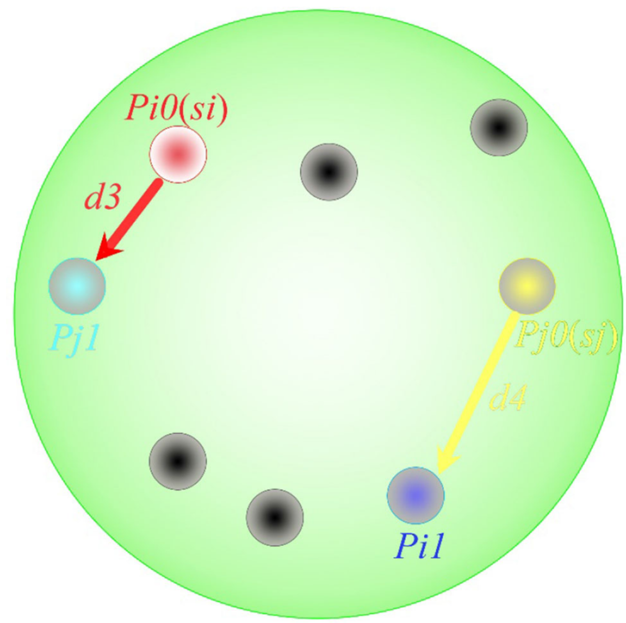


Fig. 4 Sensor node achieving final positions

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 NRSM algorithm searches the new intended positions of all instance node in the coverage area and systematically reduce the number of instance node that are needed to be moved. The sensing range may even be fully overlap by other nodes, these nodes are called redundant nodes as illustrated in Fig. 2, The sensor node s_i displace from p_{i0} to p_{i1}, thereby coverage range R_{area} (S) shows no substantial change has been recorded which ratifies that no movement is required by the s_i node. Therefore, the substantial nodes can be removed from the queue which eventually decreases the distance.

In Fig. 3, the positions of sensor node are being updated thereby at initial state, the moving distance of s_i and s_j is $d_1 + d_2$ and after the displacement, it will be updated to $d_3 + d_4$ as depicted in Fig. 4. 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 be occurred in coverage area but the area coverage distance rate will be extended.

The sensor nodes that eager to update their moving position will be substitute with the moving position of the nodes which are stationary and does not require to move further. This step can prevent the nodes to make unnecessary and longer movement. In case the node does not possess sufficient energy while reaching at intended position, the other surrounding node will surrogate the liability. Consider Fig. 5, initially, the node s_i does not plan to leave its position while at the same time node s_j wish to shift its position from P_{j0} to P_{j1} thereupon it established a vacillated link (marked by dotted green color). This node tried to displace from P_{j0} to P_{j1} but remains hiatus because the distance from node sj to pj is greater than the distance between s_i to P_j i.e., $d3 < d2$ therefore instead of node s_j, the algorithm shrewdly shifts node s_i to the intended new position of node *pj* by keeping the s_j node stationary at P_{j0}. 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

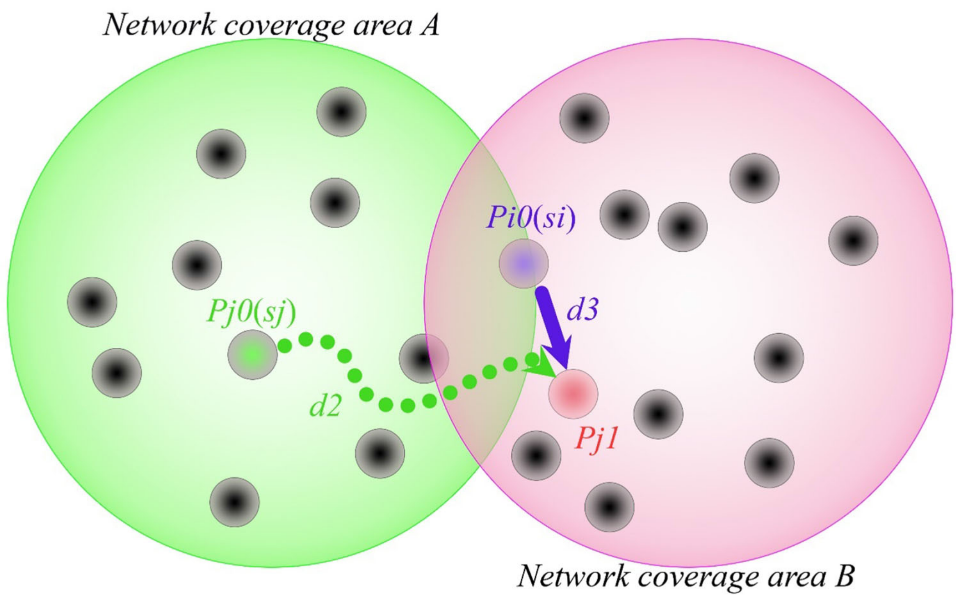


Fig. 5 NRSM movement and distance control mechanism to avail new positions

Table 2 Simulation parameters for NRSM

Simulation parameter contents	Setting value
Deployment area	60×60 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

average moving distance of the instance node are reduced which enhance the coverage area distance rate.

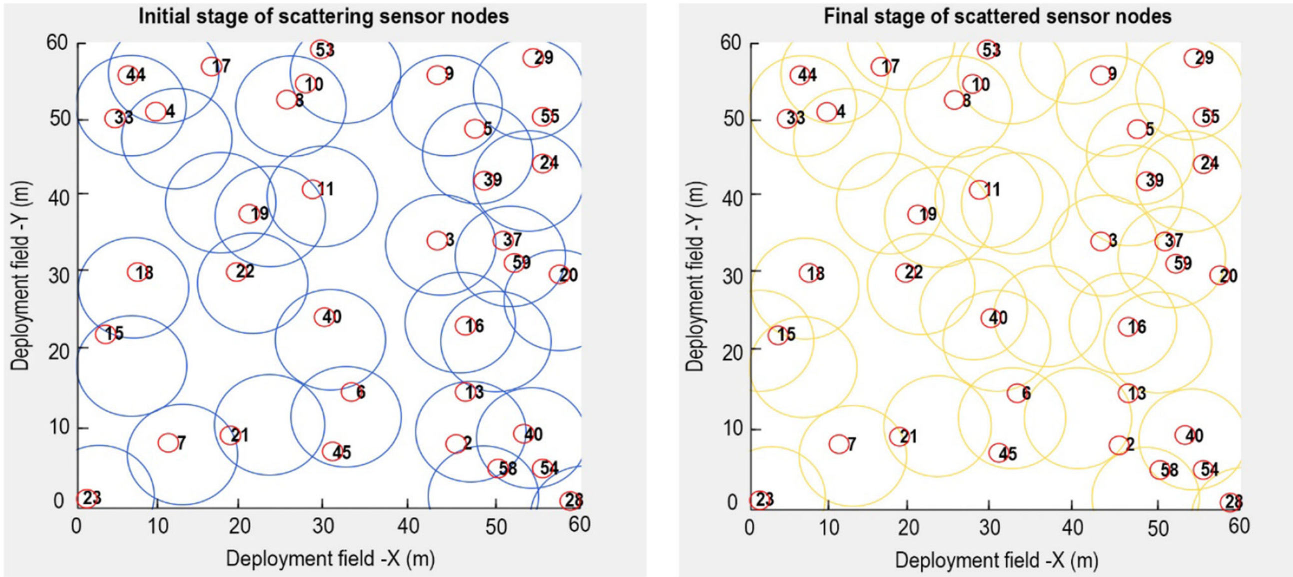
4 Simulation results and discussion

To validate the efficiency of node deployment based on NRSM, the simulation trials are conducted using MATLAB R2016a [12]. The performance among NRSM, FOA, JOA and BFA are carried out using simulation setup parameters given in Table 2. To observe the performance of aforementioned algorithms, near about 60 sensor nodes were deployed randomly in the monitoring area of size 60×60 m^2. To demonstrate the performances of proposed NRSM FOA, JOA and BFA (for energy case only) the initial and final sensor node deployment is illustrated in Figs. 6 and 7.

As the transmission begins, it can be clearly understanding that node deployment based on (NRSM) has minimum redundancy and is utmost uniform as compared to node deployment by the FOA mechanism. Table 3 signifies the influence of pulse emission rate (r) on coverage of sensor nodes. The value of r changes from 0.1 to 1 whereas 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, instance mechanism is simulated 50 times and greatest value of coverage is picked every time. The maximum value of coverage after performing NRSM is attained 93.54% at pulse emission rate of 0.9. As instances move towards respective target they emit a greater number of pulses, therefore, the pulse emission rate will be high when sensor nodes move close to the range points [13]. Thereupon, value of pulse emission rate is kept to 0.9.

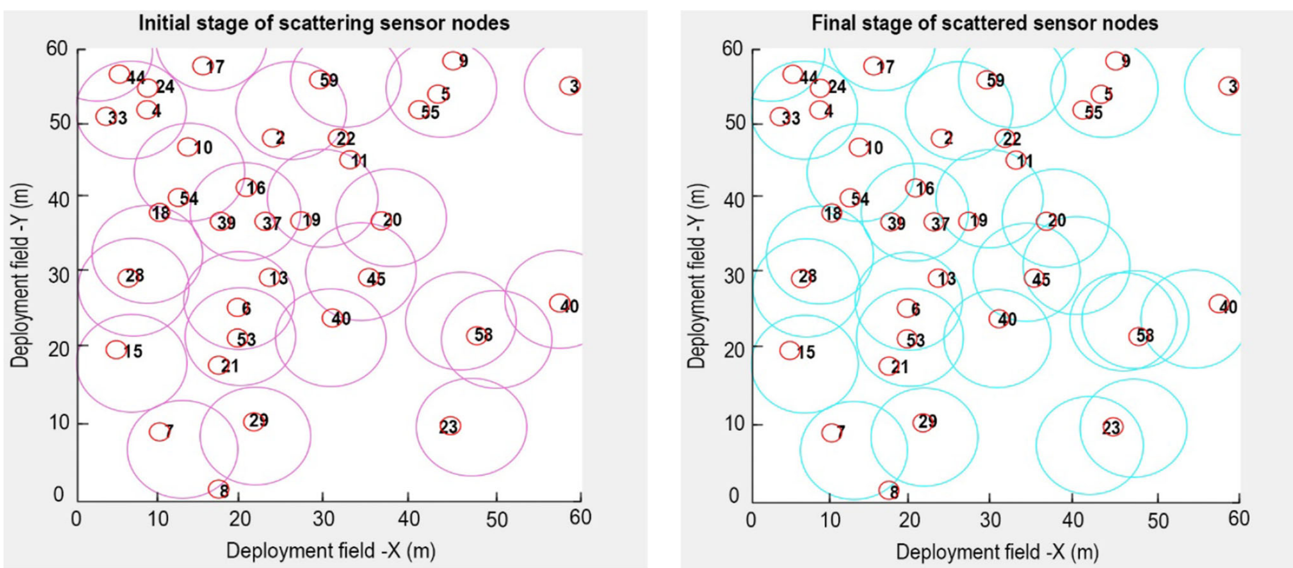
Further to analyze the effect of loudness parameter of instance mechanism on the coverage rate of sensor nodes, the value of loudness (A_o) is varied from 0.1 to 1 while pulse emission rate (r) is set to 0.9 and value of other parameters such as is 0.5, sensing radius (r_s) is fixed to 5 m. Table 4, shows the variations of loudness, initial and final coverage rate of nodes after implementing NRSM. The NRSM is run 50 times and best value of initial and final coverage rate is selected. The coverage rate after executing NRSM is obtained highest about 93.1% at 0.2 value of loudness. When sensor nodes getting near to the range point the intensity of emitted pulses is low, therefore loudness parameter should be kept low. Thereupon, the value of loudness parameter is fixed to 0.2.

In addition to this Table 5, demonstrates the effect of maximum frequency (f_{max}) [28], on coverage; its value has been changed from 0.1 to 2. The constraints of instance mechanism for instance pulse emission rate, loudness and sensing



(a) (b)

Fig. 6 **a** The initial and **b** final FOA sensor node deployment are kept constant to 0.9, 0.2 and 5



(a) (b)

Fig. 7 **a** The initial and **b** final deployment of sensor nodes by NRSM

radius are kept constant to 0.9, 0.2 and 5 respectively. For each variation of maximum frequency the proposed mechanism has been executed 50 times and supreme values of coverage before and after execution of instance mechanism has been chosen. The best value of coverage after implementing NRSM 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 range points on coverage rate of nodes, Value of range point has varied from 0.1*0.1 m to 1*1 m. The various simulation factors such as pulse emission rate, maximum frequency, sensing radius and loudness are kept constant to 0.9, 1.3, 5 and 0.2 respectively. In Table 6 for every value of coverage point NRSM runs 50 time and uppermost values of coverage rate has been taken. The highest value of coverage rate of nodes is obtained after running NRSM is 93.41% when range points are set to 0.6 m*0.6 m. Consequently, the range points have been kept constant to 0.6 m*0.6 m. Further, the sensing radius is varied from 1 to 10 m.

Table 3 Influence of pulse emission rate on coverage rate

Pulse emission (r)	Initial coverage range (%)	Final coverage range (%)
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 range (%)	Final coverage range (%)
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

Table 5 Effect of f_{max} on coverage range

f_{max} (f)	Initial coverage range (%)	Final coverage range (%)
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

Table 6 Influence of range points on network coverage

Range points (m*m)	Initial coverage range (%)	Final coverage range (%)
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

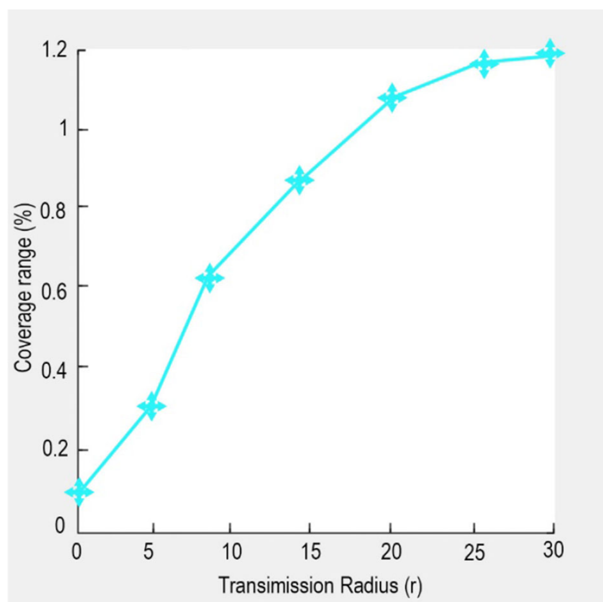
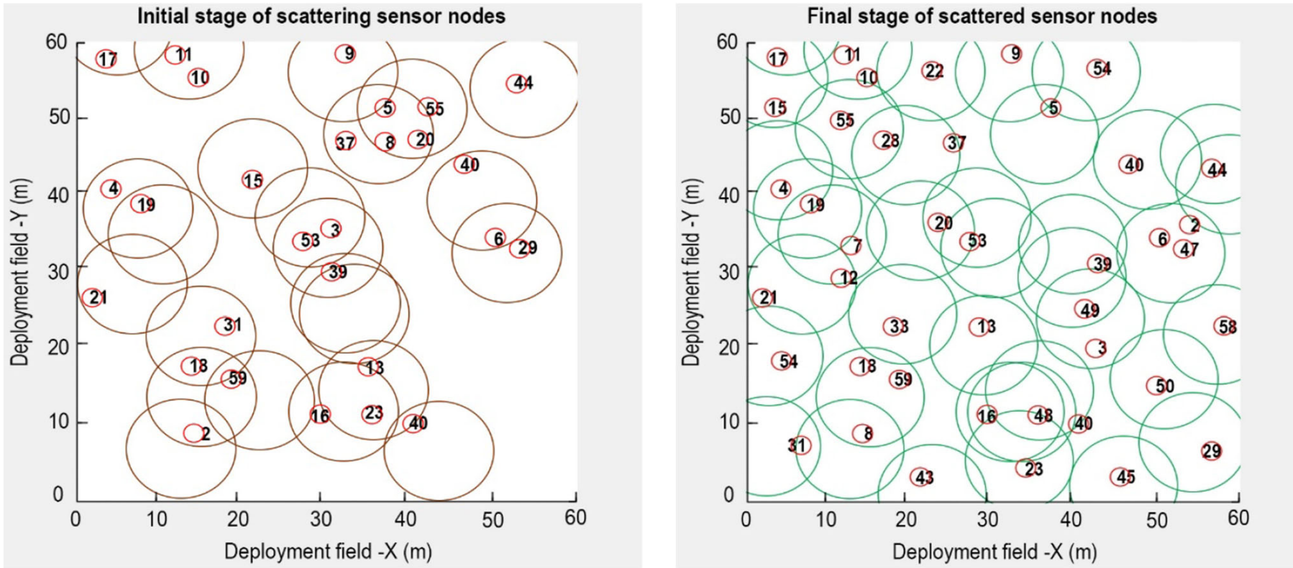


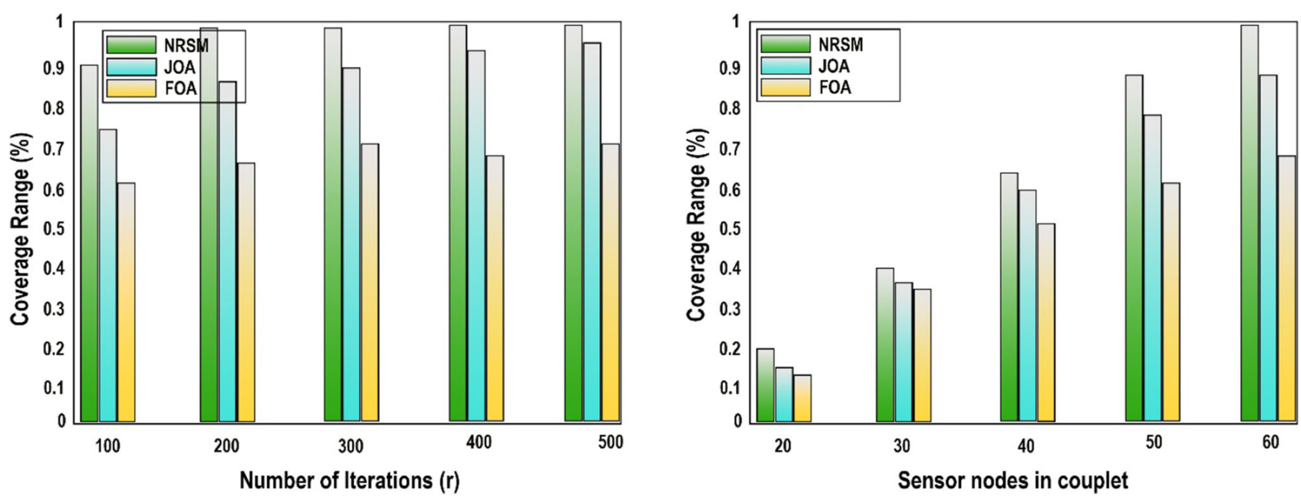
Fig. 8 Coverage range for varying sensing radius of sensor nodes by NRSM

Figure 8, signifies the variations of coverage range after applying NRSM w.r.t. changes in the sensing radius of node. The parameters of NRSM for example range points, loudness, pulse emission rate and maximum frequency are set as 0.6*0.6 m, 0.2, 0.9 and 1.3 respectively. It is clear from Fig. 8, as the sensing radius has increased, thereby 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 trade-off



(a) (b)

Fig. 9 **a** Initial deployment of sensor nodes for NRSM. **b** Final deployment of sensor nodes by NRSM



(a) for various iterations (b) for varying number of sensor nodes

Fig. 10 Coverage range comparative analysis for NRSM, JOA and FOA. **a** For various iterations, **b** for varying number of sensor nodes

between the sensing radius and cost, while sensing radius of node is increased the cost of sensor nodes also increased.

The tuned values of various constraints of NRSM such as loudness, maximum frequency, sensing radius, pulse emission rate and range points are 0.2, 1.3, 6, 0.9 and 0.6*0.6 m respectively. To validate the performance of node deployment based on NRSM after setting above constraints values, the initial and final node deployment after executing (NRSM) are shown in Fig. 9. Thereupon, it can be obviously seen that node deployment based on NRSM has lowest redundancy than NRSM and FOA. To further demonstrates the effectives of coverage range curve for NRSM compare to FOA for various iterations as shown in Fig. 10. The iterations are varied from 0 to 500. The convergence speed of NRSM is exorbitant as compared to FOA. The NRSM converged arounds are 150 iterations whereas FOA converges around reached 350 iterations due to exploitation characteristics of the sensor nodes.

The NRSM has achieved more coverage rate about 99.46% as compared to 93.37%, 88.33% of JOA and FOA. To overwhelm the effect of randomness NRSM, instance mechanism optimization and fruit fly algorithms are run 15 times respectively. The deployment results in terms of average coverage rate, standard deviation, best and worst coverage values

Table 7 Deployment results for NRSM, JOA and FOA

Algorithms	FOA		JOA		NRSM	
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%

Table 8 Comparison of computation time of NRSM, JOA and FOA

Algorithms	FOA	JOA	NRSM
Computation time (s)	0.28	0.019	0.016

for NRSM, JOA and FOA are represented in Table 7. It can be obviously seen from Table 7, that NRSM has achieved the average coverage range about 98.29% as compared to 91.91%, 85.16% of JOA and fruit fly algorithm. Further the standard deviation for node deployment based on NRSM is lowest, therefore NRSM is more stable as compared to FOA and JOA. The best and worst coverage values for NRSM are 99.46% and 97.31% as compared to 94.30% and 90.02%, 87.49% and 78.20% for JOA and FOA based on node deployment.

Further the comparison of NRSM, JOA and FOA in terms of computation time is represented in Table 8. The computation time for NRSM is less i.e. 0.016 s as compared to 0.019 s, 0.28 s for JOA and FOA. The NRSM and JOA converges at 25 iterations whereas FOA converged at 500 iterations, therefore the speed of NRSM and JOA is more and converges faster at earlier stage because of its exploitation feature as compared to fruit fly algorithm.

During each transmission round, the overall energy diminution analysis is illustrated in Fig. 11. It can be seen that between 20 and 32 nodes all algorithms going to die. The proposed NRSM has consumed only 100 to 150 J of energy approaching to relative position as compare to JOA, FOA and BFA. The consumed energy increases when the coverage degrees required increase, since the sensor nodes require more energy to cover target positions and therefore it takes more energy for sensing and communication tasks.

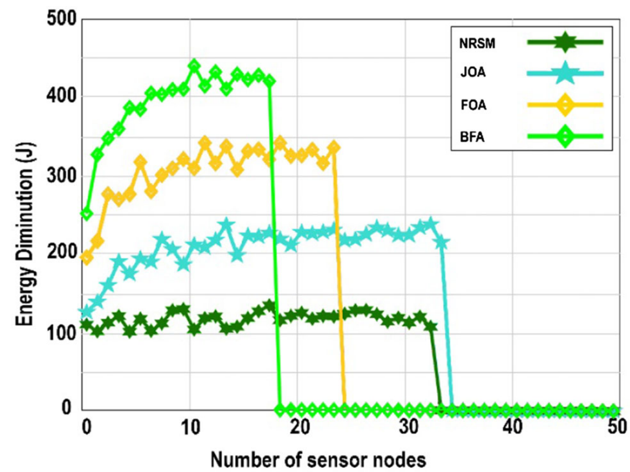


Fig. 11 Coverage range for varying sensing radius of sensor nodes by NRSM

5 Conclusion

Wireless sensor networks are severely facing the coverage issues therefore a shrewd coverage mechanism is presented in this study. The proposed algorithm Node Redeployment Shrewd Mechanism (NRSM) has been designed to overcome the tribulations occurred due to the uncouth deployment of the sensor node which ultimately has great impact over network coverage range. The NRSM functions in two phases, in first phase it searches the new intended node positions through Dissimilitude Enhancement Scheme (DES) and moves the node to new position. For second phase, the distance measurement between moving sensor node and the intended position is reduced and number of sensor movements are also being controlled sagaciously. This process is called Depuration.

The analysis of various factors of NRSM such as loudness, range points, emission rate and radius of nodes, and frequency have also been identified. The performance metrics of NRSM has been obtained by conducting simulation test through MATLAB and meticulously compared with previous well-known algorithms FOA, JOA and BFA. The simulation results vouched that NRSM has attained mean coverage rate about 98.29% which is higher as compared to rest of the algorithms. The proposed NRSM algorithm has appeared with higher coverage range and less computation time compared to all. In future the various evolutionary optimization algorithms can be applied to solve the node deployment issues to enhance the coverage range phenomenon.

Funding This work is completely self-funded, thereby no any financial agency's role is available.

References

1. F. Ait Aoudia, M. Gautier, M. Magno, O. Berder, and L. Benini, "A generic framework for modeling MAC protocols in wireless sensor networks," IEEEACM Trans. Netw., 25(3):1489–1500, 2017, doi: https://doi.org/10.1109/TNET.2016.2631642.
2. S. Ashraf, M. Gao, Z. Chen, S. Kamran, and Z. Raza, "Efficient node monitoring mechanism in wsn using Contikimac protocol," Int. J. Adv. Comput. Sci. Appl., 8(11), 2017, doi: https://doi.org/10.14569/IJACSA.2017.081152.
3. 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, 1st ed. 2018 edition. New York, NY: Springer, 2018.
4. "Shortest path routing protocol based on the vertical angle for underwater acoustic networks." https://www.hindawi.com/journals/js/2019/9145675/. Accessed May 12, 2020.
5. S. Ashraf and T. Ahmed, "Machine learning Shrewd approach for an imbalanced dataset conversion samples," J. Eng. Technol. JET, 11(1), 2020, https://journal.utem.edu.my/index.php/jet/article/view/5896. Accessed Jul 05, 2020 [Online]
6. Ashraf, S., Ahmed, T., Raza, A., Naeem, H.: Design of Shrewd underwater routing synergy using porous energy shells. Smart Cities **3**(1), 74–92 (2020). https://doi.org/10.3390/smartcities3010005
7. "Flower pollination algorithm based localization of wireless sensor network - IEEE Conference Publication." https://ieeexplore.ieee.org/document/7453299/. Accessed May 12, 2020.
8. Ashraf, S., Ahmed, T., Saleem, S., Aslam, Z.: Diverging mysterious in green supply chain management. Orient. J. Comput. Sci. Technol. **13**(1), 22–28 (2020). https://doi.org/10.13005/ojcst13.01.02
9. J. Zhang, Y. Lei, C. Chen, and F. Lin, "Directional probability perceived nodes deployment based on particle swarm optimization". 2016. https://journals.sagepub.com/doi/full/https://doi.org/10.1155/2016/2046392. Accessed Jun 07, 2020.
10. Das, S., Biswas, A., Dasgupta, S., Abraham, A.: Bacterial foraging optimization algorithm: theoretical foundations, analysis, and applications. In: Abraham, A., Hassanien, A.-E., Siarry, P., Engelbrecht, A. (eds.) Foundations of computational intelligence volume 3: global optimization, pp. 23–55. Springer, Berlin (2009)
11. Ashraf, S., Aslam, Z., Yahya, A., Tahir, A.: Underwater routing protocols analysis of intrepid link selection mechanism, challenges and strategies. Int. J. Sci. Res. Comput. Sci. Eng. **8**(2), 1–9 (2020). https://doi.org/10.26438/ijsrcse/v8i2.19
12. Ashraf, S., Gao, M., Mingchen, Z., Ahmed, T., Raza, A., Naeem, H.: USPF: underwater shrewd packet flooding mechanism through surrogate holding time. Wirel. Commun. Mob. Comput. **2020**, 1–12 (2020). https://doi.org/10.1155/2020/9625974
13. S. Ashraf, A. Raza, Z. Aslam, H. Naeem, and T. Ahmed, "Underwater resurrection routing synergy using astucious energy pods," J. Robot. Control JRC, 1(5), 2020, doi: https://doi.org/10.18196/jrc.1535.
14. Ashraf, S., Saleem, S., Chohan, A.H., Aslam, Z., Raza, A.: Challenging strategic trends in green supply chain management. Int. J. Res. Eng. Appl. Sci. JREAS **5**(2), 71–74 (2020)
15. Ashraf, S., Arfeen, Z.A., Khan, M.A., Ahmed, T.: SLM-OJ: surrogate learning mechanism during outbreak juncture. Int. J. Mod. Trends Sci. Technol. **6**(5), 162–167 (2020). https://doi.org/10.46501/IJMTST060525
16. Zhang, Q., Fok, M.: A two-phase coverage-enhancing algorithm for hybrid wireless sensor networks. Sensors **17**(12), 117 (2017). https://doi.org/10.3390/s17010117
17. Li, J., Zhang, B., Cui, L., Chai, S.: An extended virtual force-based approach to distributed self-deployment in mobile sensor networks. Int. J. Distrib. Sens. Netw. **8**(3), 417307 (2012). https://doi.org/10.1155/2012/417307
18. Wang, G., Cao, G., La Porta, T.F.: Movement-assisted sensor deployment. IEEE Trans. Mob. Comput. **5**(6), 640–652 (2006). https://doi.org/10.1109/TMC.2006.80
19. Mahboubi, H., Aghdam, A.G.: Distributed deployment algorithms for coverage improvement in a network of wireless mobile sensors: relocation by virtual force. IEEE Trans. Control Netw. Syst. **4**(4), 736–748 (2017). https://doi.org/10.1109/TCNS.2016.2547579
20. W.-T. Pan, "A new fruit fly optimization algorithm: taking the financial distress model as an example," Knowl.-Based Syst., 26: 69–74, 2012, doi: https://doi.org/10.1016/j.knosys.2011.07.001.
21. Aliyu, M.S., Abdullah, A.H., Chizari, H., Sabbah, T., Altameem, A.: Coverage enhancement algorithms for distributed mobile sensors deployment in wireless sensor networks. Int. J. Distrib. Sens. Netw. **12**(3), 9169236 (2016). https://doi.org/10.1155/2016/9169236
22. M. Abazeed, N. Faisal, S. Zubair, and A. Ali, "Routing protocols for wireless multimedia sensor network: a survey," J. Sens., 2013. https://www.hindawi.com/journals/js/2013/469824/. Accessed Jul 27, 2020.
23. Yoon, Y., Kim, Y.-H.: An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks. IEEE Trans. Cybern. **43**(5), 1473–1483 (2013). https://doi.org/10.1109/TCYB.2013.2250955
24. L. Sun, X. Song, and T. Chen, "An improved convergence particle swarm optimization algorithm with random sampling of control parameters," J. Control Sci. Eng., 2019. https://www.hindawi.com/journals/jcse/2019/7478498/. Accessed Jul 27, 2020.
25. "How to Calculate Euclidean Distance." https://sciencing.com/how-to-calculate-euclidean-distance-12751761.html. Accessed May 12, 2020.
26. Ashraf, S., Ahmad, A., Yahya, A., Ahmed, T.: Underwater routing protocols: analysis of link selection challenges. AIMS Electron. Electr. Eng. **4**(3), 234–248 (2020). https://doi.org/10.3934/ElectrEng.2020.3.234
27. Storn, R., Price, K.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J. Glob. Optim. **11**(4), 341–359 (1997). https://doi.org/10.1023/A:1008202821328
28. Ashraf, S., Ahmed, T.: Dual-nature biometric recognition epitome. Trends Comput. Sci. Inf. Technol. **5**(1), 008–014 (2020). https://doi.org/10.17352/tcsit.000012