An efficient DVHOP localization algorithm based on simulated annealing for wireless sensor network

Omar Arroub[1], Anouar Darif[2], Rachid Saadane[3], My Driss Rahmani[1], Zineb Aarab[1,4]

[1]LRIT-GSCM Associated Unit to CNRST (URAC 29), FSR Mohammed V-Agdal University, Rabat, Morocco
[2]LIMATI, Faculte Polydisciplinaire, University of Sultan Moulay Slimane, Beni Mellal, Morocco
[3]SIR/LAGES-EHTP, Hassania School of Public Labors, Casa-Oasis, Morocco
[4]GENIUS Laboratory-SUPMTI of Rabat, Rabat, Morocco

Article Info

Article history:

Received Mar 19, 2024
Revised Dec 18, 2024
Accepted Feb 27, 2025

Keywords:

DVHOP
Localization
Metaheuristic algorithms
Simulated annealing
WSN

ABSTRACT

In the last decade, the research community has devoted significant attention to wireless sensor networks (WSNs) because they contribute positively to some critical issues encountered in nature and even in industry. On the other hand, localization is one of the most important parts of WSN. Hence, the conception of an efficient method of localization has become a hot research topic. Lastly, it has been invented, a set of optimal positioning methods that make locate a node with low cost and give precise results. In our contribution, we investigate the source of imprecision in the distance vector-hop (DVHOP) localization algorithm. However, we found the last step of DVHOP caused an imprecision in the calculation. Consequently, our work was to replace this step, aiming to reach satisfactory precision. For that purpose, we created three improved versions of this algorithm by adopting two meta-heuristic (simulated annealing, particle swarm optimization) and Fmincon solver dedicated to optimization in the field of WSN node localization. The experimental results obtained in this work prove the efficiency of simulated annealing (SA)-DVHOP in terms of accuracy. Furthermore, the enhanced algorithm outperforms its opponents by varying the percentage of anchors and the number of nodes.

Corresponding Author:

Omar Arroub
LRIT-GSCM Associated Unit to CNRST (URAC 29), FSR Mohammed V-Agdal University
BP 1014 Rabat Morocco
Email: omar_arroub@um5.ac.ma

1. INTRODUCTION

Nowadays, with the continuous development of micro electro-mechanical systems (MEMS), there is a significant interest by researchers in wireless sensor network (WSN) [1], because the latter has shown efficiency in different applications, such as military sensing, smart environmental [2], vehicular ad-hoc network (VANET) [3], healthcare [4], agriculture [5], industry [6], and multimedia [7]. However, localization is an important part of WSN. Indeed, without location's information, messages will be missed. For example, using WSN in order to detect the fire forest. Indeed, bringing the location information to the base station can help the firefighter react rapidly to make the necessary interventions. At this point, the commonly used solution to locate the sensor node in WSN is the global positioning system (GPS). Perhaps, this positioning solution is not practicable in all cases because GPS cannot be used in indoor areas. Besides that, it consumes a lot of energy. In order to mitigate the two issues caused by GPS, some alternative solutions have been invented to the localization problem, in which we equip just a few sensor nodes with GPS named anchors and those anchors help the other unknown nodes be aware of their positions by using network connectivity and some additional calculations. By opting for alternative localization solutions, we avoid the high energy consumption of the localization process.

The localization techniques can be classified into two classes: range-based and range-free techniques [8], [9]. However, range-free methods are based on the connectivity of the network, the advantages of those methods that they don't need any additional hardware, making them more efficient in terms of low cost. In the field of research, the most commonly used range-free techniques include APIT [10], Centroid [11], distance vector-hop (DVHOP) [12], [13] and Amorphous algorithm [14]. On the other hand, range-based techniques [15] make the localization by using time of arrival (TOA) [16], angle of arrival (AOA) [17], time difference of arrival (TDOA) [18] and received signal strength indicator (RSSI). In general, those methods require additional material, but they offer a high level of accuracy, making them more precise and expensive than range-free techniques. Moreover, in some cases, the leakage of deployed anchors in WSN may lead to weak coverage of the network. To mitigate this issue, multi-hop localization algorithms can be used. The specificity of those algorithms is that sensor nodes may be located even if they aren't in communication range with anchors. The most known multi-hop localization algorithm is DVHOP. The advantages of DVHOP reside in its simplicity of implementation. Also, this algorithm gives the results quickly. Besides that, DVHOP can offer good coverage of localization in comparison with other localization algorithms. Its drawback is the low accuracy offered, especially when the network becomes small. Therefore, many improvements have been proposed to enhance the precision of the traditional DVHOP. In our approach, we create three improved versions of DVHOP in order to avoid the least square method adopted by the traditional algorithm because it's the main reason for locating the sensor node imprecisely.

In fact, in uniform deployment, it's been found that DVHOP is a suitable algorithm in terms of coverage of localization and can also offer an acceptable level of accuracy. However, when the network becomes anisotropic due to the presence of an irregularity in the distribution of nodes, the accuracy of the algorithm becomes worse because the hop-size calculation done by DVHOP in a non-uniform network leads to a big inaccuracy in the distance calculation step. Consequently, the average localization error (ALE) of the algorithm is characterized by insufficiency. Aiming to enhance the localization accuracy of DVHOP in non-uniform networks, the research community has invented Amorphous localization algorithm that makes the distance calculation using an offline method. Indeed, Amorphous uses Kleinrock and Silvester formula in order to calculate hop-size for reducing the localization error. In our contribution, we bring DVHOP for improvement in both cases of the distribution (uniform, no-uniform) aiming to correct the issue of resolving the non-linear equations presented in the multilateration process. As we know, the multilateration is the intersection of the circles with the purpose of locating the target. However, more circles are required to calculate the coordinates of the unknown node more precisely. That means more equations are devoted to resolution purposes. In addition to that, those equations are presented in non-linear form. As a consequence, we are facing a huge and complicated problem. Our aim was to transform the cited issue into an optimization problem. Indeed, it was seen that simulated annealing [19], [20], particle swarm optimization (PSO) [21], [22], Fmincon [23], [24] the convenient methods for resolving the cited problem.

The main contributions of this paper are as follows:

i) The importance of WSN has led the research community to investigate more about the problem of localization in WSN. DVHOP belongs to range-free localization techniques; its last step is judged to be the main reason for the imprecision of DVHOP. Furthermore, the traditional formula equation used to retrieve the locations causes an error. Hence, the latter can be reformulated as an optimization problem.
ii) The resolution of the least square method adopted by DVHOP may be interpreted as a minimizing problem that has the ability to be resolved either by simulated annealing, PSO and Fmincon solver dedicated to mathematical optimization under MATLAB. The purpose of those modifications is to enhance the accuracy of the traditional DVHOP.
iii) The performance comparison of simulated annealing (SA)-DVHOP, Fmincon-DVHOP, PSO-DVHOP and DVHOP is carried out under two different network environments. The experimental results prove that the proposed SA-DVHOP has a smaller localization error. Also, it's shown that the improved method is not dependent on the additional anchors to give the best result.

The remainder of the paper is as follows: firstly, we expose different works that have already been done to enhance DVHOP in static WSN with a uniform and random distribution of nodes. In section 3, we define the research methodology followed to accomplish this work. In section 4, we introduce DVHOP in detail by citing its advantages and drawbacks, then we present our improved versions of DVHOP. Simulation is done and discussed in section 5. Finally, we conclude the paper and present future works in section 6.

2. RELATED WORKS

In fact, there are two factors contributing to the imprecision of DVHOP. Firstly, by the least squares method used to solve the non-linear equations. Secondly, by the manner adopted in averaging hop-size. During our research, we find that the research community focuses on resolving DVHOP by replacing the least squares method because the error introduced in the distance calculation step has a slight impact on the accuracy of DVHOP. In general, many scientists adopt nature-inspired meta-heuristic algorithms [25] to enhance the precision of DVHOP. For example, we find that PSO is mostly used to mitigate this issue. Xue [26], adopts PSO and he uses linear decreasing inertia weight (LDIW) [27] to have a balance between the exploration and exploitation phases of PSO in order to minimize the cost function and reach the global solution. Then, the solution extracted by PSO presents the optimal location of the unknown node. Sharma and Kumar [28], the study of localization is extended to three dimensions. In addition to that, the genetic algorithm has been used in order to improve DVHOP. In detail, the process of positioning is summarized in six steps: flooding phase, hop-size calculation, population initialization, crossover, selection and mutation. Briefly, GA-DVHOP [29] changes the last phase of the traditional DVHOP to genetic steps, aiming to reach a better solution. Although that method gives a high degree of accuracy, its drawback resides in its high complexity in comparison with PSO-DVHOP.

Perdana *et al.* [30], showed that Amorphous outperforms DVHOP in terms of accuracy by varying the number of nodes and the percentage of anchors. This study also proves the efficiency of Amorphous in terms of energy consumption. According to the experimental results, it's confirmed that Amorphous reach a satisfactory precision in WSN with a few anchors. However, DVHOP requires more anchors to perform better. Ali *et al.* [31] stated a performance comparison of Amorphous and DVHOP has been done, the metric of evaluation was the accuracy of localization, energy consumption and network overheard. Also, it's shown for both algorithms cited that the accuracy of localization is inversely proportional to the energy consumed by the node. As we know, in a non-uniform network, we need more anchors. Hence, Amorphous outperforms DVHOP in terms of accuracy and energy consumption because Amorphous doesn't require many anchors in its locating process. Han *et al.* [32], use a genetic algorithm to improve DVHOP and they adopt PSO to refine the crossover step. The simulations realized in this research prove the efficiency of the ameliorated version of DVHOP in terms of precision by varying the percentage of anchors.

In this work, we attempt to enhance DVHOP aiming to reduce its imprecision in locating. Our method consists of replacing the least square method with simulated annealing. Indeed, the latter makes the calculation with low complexity, making SA-DVHOP more precise than the traditional DVHOP localization algorithm.

3. RESEARCH METHOD

Our research methodology is as follows: firstly, we study DVHOP deeply by analyzing the reason behind its huge error. Secondly, we formulate the last step of DVHOP into an optimization problem in which we minimize the fitness function. That means we minimize the sum of errors accumulated during the multi-lateration process, obviously to reach the convenient locations of unknown nodes. For that purpose, we have adopted SA algorithm, PSO, and Fmincon solver dedicated to mathematical optimization under MATLAB to replace the step of resolution done by DVHOP localization algorithm. Finally, our experience is split into two phases. In the first step, we prepare an experimental environment by fixing the number of nodes. Then, we compare the performance of our improved version of DVHOP and the traditional DVHOP by varying the percentage of anchors. In the second step, we re-make the comparison by varying the number of nodes and keeping the percentage of anchors at 30%.

4. THE PROPOSED APPROACH FOR DVHOP IMPROVEMENT

We considered a WSN, where n is the number of nodes distributed throughout the field of sensing, whose surface equals 1,000×1,000 m^2. Additionally, we used three optimization methods: PSO, SA, and Fmincon in order to minimize the cost function of DVHOP. Table 1 summarizes all the parameters used for the traditional algorithm DVHOP and the improved algorithms based on PSO, SA, and Fmincon.

4.1. Traditional DVHOP algorithm

DVHOP is a distributed localization algorithm. It was invented by Niculescu and Nath in 2003; it's based on vector distance routing and consists of three different steps, as follows:

- Step1: Flooding
  Each unknown node knows the number of hops to their anchor by a mechanism of broadcast done by anchors.

Table 1. Summary of notations

Symbol	Description
$hopsize_i$	Hop size between anchors
$hopcount_{u,i}$	Number of hop between anchor i and anchor j
$d_{u,i}$	Distance between anchor i and the unknown node
$f(x,y)$	Cost function to optimize
T_k	Temperature of the solid
α	Parameter that express the decrease in temperature
E	Energy of the system
$P(E,T)$	The probability calculated according Bolzman distribution
x_i	Position vector of the particle
v_i	Velocity vector of the particle
p_g	Best position reached by all particles
c_1, c_2	Acceleration coefficients
r_1, r_2	Reel numbers that adjust the displacement of particles
A, b, Aeq, beq, x0, lb, ub	Attributes of Fmincon function

– Step2: Hop-size and distance calculation
After the flooding process, we can obtain the hop-size between anchors according to the formula (1).

$$hopsize_i = \frac{\sum_{j=1\, j\#i}^{n} \sqrt{(x_i-x_j)^2+\left(y_i-y_j\right)^2}}{\sum_{j=1}^{n} hopcount_{ij}} \tag{1}$$

Where (x_i,y_i), (x_j,y_j) represent respectively the coordinates of anchors i,j.
After obtaining hopsize, in (2) is used to calculate the distance between anchor i and the unknown node.

$$D_{u,i} = hopsize_i \times hopcount_{u,I} \tag{2}$$

– Step3: Calculation of unknown node position
In this step, we specify the coordinates of all unknown nodes. For each unknown node, we apply the least square method to estimate its location.

(x,y) denotes the coordinates of the unknown nodes, (a_i,b_i) represents the location of the anchor node, where i=1,2,..n and n is the number of anchors, thus the distance between unknown nodes and n anchors is expressed by the non-linear equations:

$$\begin{cases} (x-a_1)^2+(x-b_1)^2+(z-c_1)^2 = d_1^2 \\ \vdots \\ (x-a_n)^2+(x-b_n)^2+(z-c_n)^2 = d_n^2 \end{cases} \tag{3}$$

Then we find:

$$\begin{cases} x^2-2a_1x+a_1^2+y^2-2b_1y+b_1^2 = d_1^2 \\ \vdots \\ x^2-2a_nx+a_n^2+y^2-2b_ny+b_n^2 = d_1^2 \end{cases} \tag{4}$$

In (4) can be extended to:

$$\begin{cases} -2x(a_1-a_n)+a_1^2-a_n^2-2y(b_1-b_n)+b_1^2-b_n^2 = d_1^2 \\ \vdots \\ -2x(a_{n-1}-a_n)+a_{n-1}^2-a_n^2-2y(b_{n-1}-b_n)+b_{n-1}^2-b_n^2 = d_{n-1}^2 \end{cases} \tag{5}$$

The solution of the system may be interpreted to the resolution of the equation $Ax=b$ where:

$$A = \begin{bmatrix} 2(a_1-a_n) & 2(b_1-b_n) \\ \vdots & \vdots \\ 2(a_{n-1}-a_n) & 2(b_{n-1}-b_n) \end{bmatrix} \tag{6}$$

$$b=\begin{bmatrix} a_1^2 - a_n^2 + b_1^2 - b_n^2 + c_1^2 - c_n^2 - d_1^2 \\ \vdots \\ a_{n-1}^2 - a_n^2 + b_{n-1}^2 - b_n^2 + c_{n-1}^2 - c_n^2 - d_{n-1}^2 \end{bmatrix} \quad (7)$$

In the traditional DVHOP algorithm, the least square estimator done in the last step of the positioning process causes a huge error in locating the target node, which has a big influence on the accuracy of DVHOP. In addition to that, DVHOP requires additional anchors to offer acceptable coverage of localization. Hence, the latter has the disadvantage of high energy costs. As a consequence, it is seen as necessary to bring DVHOP for improvement to overcome its existing disadvantages.

In our approach, we attempt to keep the two first steps of DVHOP and change the last step to an optimization problem. In detail, the proposed method is summarized in three steps: each unknown node knows its number of hops to their anchors through a broadcast done by the anchors. Secondly, we calculate the distance between anchors and unknown nodes on the basis of the hop-size. Lastly, we select a specified meta-heuristic to minimize the sum of errors occurring in the multilateration process. Hence, the positioning problem may be interpreted to solve the minimization of the fitness function mentioned in (8).

$$f(x,y) = \frac{1}{n}\sum_{i=1}^{n} \left| \sqrt{(x-a_i)^2 + (y-b_i)^2} - d_i \right| \quad (8)$$

Where n is the number of anchors, (a_i,b_i) are the coordinates of anchors, d_i is the distance between anchor i and unknown node.

In this part, we discuss three improved versions of the traditional DVHOP localization algorithm ameliorated by adopting two meta-heuristics (SA, PSO) and Fmincon solver dedicated to optimization under MATLAB. Our purpose is to enhance the precision of DVHOP. The two first steps in all enhanced algorithms are similar to the two first steps of DVHOP algorithm because these steps are the main reason for the high coverage of the localization of DVHOP. Hence, we leave these steps as they are and tackle our modifications in the last step of DVHOP. Moreover, we change the least square method adopted by the traditional algorithm to an optimization problem that can be resolved by each of the methods cited above. In the following, we shall cite in detail our improved versions of DVHOP.

4.2. DVHOP algorithm based simulated annealing

SA is a stochastic global search optimization that was introduced by Kirkpatrick *et al.* [33] in 1983. As a normal local search method, it uses a special strategy to avoid the local minima. This meta-heuristic is based on heating and cooling in order to obtain a flawless alloy. In detail, this method alternates the cycles of heating and cooling the metals slowly. The main advantage of this technique is the use of probabilistic methodology, which permits avoiding local solutions and increases the exploration process.

In general, the purpose of SA is to traverse the space of solutions in an iterative manner. We start with an initial solution S_0 (generated randomly) which denotes the initial energy E_0. Additionally, we define a variable called temperature changes from the initial temperature (T_0) (generally high) to the final temperature. It's assumed that an elementary change occurred in the solution at each iteration of the algorithm. This change causes a variation in the energy of the system that we denote ΔE. If ΔE is negative, the new solution is accepted because it improves the cost function. If ΔE is positive, the solution found maximizes the energy of the system. Hence, it's considered worse than the previous solution. As a consequence, the new solution will be accepted with a probability P calculated according to the following Boltzman distribution:

$$P(E,T) = exp(-\Delta E/T) \quad (9)$$

Where T denotes the temperature of the solid.

The choice of temperature is essential to guaranteeing the balance between intensification and diversification of solutions in the space of research. First, the choice of the initial temperature depends on the quality of the starting solution. Indeed, the initial value of the temperature must be relatively high. T is calculated iteratively as follows:

$$Tk + 1 \leftarrow Tk \times \alpha \quad (10)$$

$\alpha \in [0,1]$, α is a parameter that expresses the decrease in temperature of the iteration. The decrease in temperature can also be carried out in stages. That is to say, the decrease only changes after a certain number of iterations. On the other hand, we can also raise the temperature when the search process seems blocked in a region of the search space. We can then consider a high increase in temperature as a process of diversification. While the decrease in temperature corresponds to an intensification process.

In the beginning, we generate the initial solution. Then, we calculate $df = f(x_{new}) - f(x)$ so if $df < 0$, we accept the new solution; otherwise, we accept the solution according to the to the Metropolis rule. Then, we test if the number of iterations is reached, so if the final condition is satisfied, we return the final solution; otherwise, we decrease the temperature and we set the number of iterations to 0. The flowchart indicated in Figure 1 describes in detail the functioning of SA.

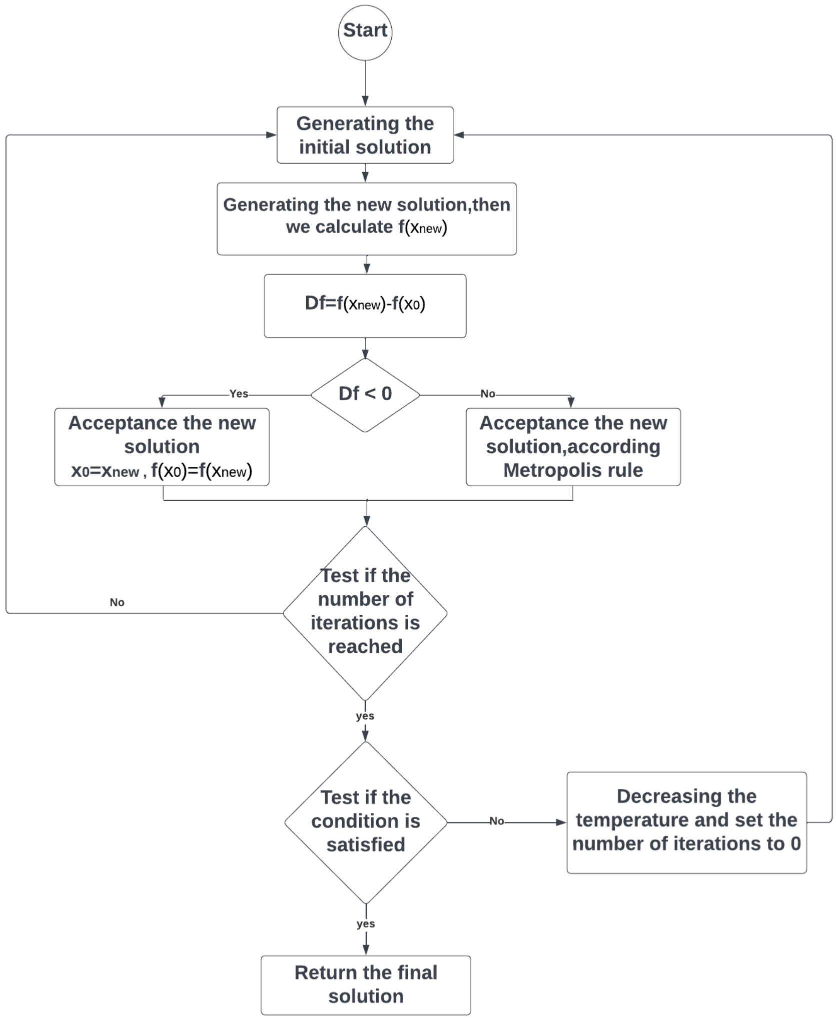

Figure 1. Flow chart of SA algorithm

The principle of SA-DVHOP is as follows: we execute simulated annealing algorithm inside the browsing of unknown nodes, more precisely after the distance calculation step. Indeed, for each solution extracted by simulated annealing algorithm, it will be assigned to an unknown node. The pseudo-code mentioned in Algorithm 1 describes the steps of SA-DVHOP algorithm.

Algorithm 1. SA-DVHOP algorithm

```
Initialization:
number of nodes=NB,
number of anchors=NA,
area of experimentation =1000×1000 m²,
communication range=500 m
1.calculation of hopcounti,j by finding the shortest path between nodes
2.hopsize calculation according (1)
3.calculate the positions of unknown nodes
for i=NA to NB
```

```
  4.distance calculation
  unknown_to_anchrs_dist=hopsize(i) × shortest_path(i,1 to NA);
  5.fitness function f is calculated according (8)
  6.execution of Simulated annealing algorithm
  initialize the temperature T according to the
  cooling scheme (10)
  while (condition of cooling is not satisfied)
         generate a random neighbor S' from S
         calculate ∆E = f(S') - f(S)
         if ∆E ≤ 0
          S←S'
        else
         accept S0 as the new solution with
         probability P(E,T) = exp(-∆E/T)
         end
        update T based on cooling scheme
   end
   return pbest
   6.assign the result of SA to an unknown node
   node.estimated(i,1to 2)=pbest;
end
```

4.3. DVHOP algorithm-based particle swarm optimization

The optimization by particle swarm was invented by Kennedy and Eberhart in 1995. This method is based on the social behavior of animals living in swarms. Indeed, the particles are individuals and they move in order to search for a global solution, according to the following information:

- Each particle has the ability to memorize the best point already passed and it attempts to return to this point.
- Each particle is aware of the best point in its area and it will attempt to go towards this point.

PSO consists of a swarm of particles that fly throughout the space of solutions in order to reach the global solution. Analytically, in R^n, the particle *i* of the swarm (potential solution) is modeled by its position vector $x_i=(x_{i1}, x_{i2},..., x_{in})$ and by its velocity vector $v_i=(v_{i1}, v_{i2},..., v_{in})$. This particle remembers the ideal position that we noted $p_i=(p_{i1}, p_{i2},..., p_{in})$, the best position reached by all the particles of the swarm is noted by $p_g=(p_{g1}, p_{g2},..., p_{gn})$. We can express the velocity vector using (11).

$$V_{i,j}\,(t+1) = v_{i,j}\,(t) + c_1 r_1 \left(p_{i,j}\,(t) - x_{i,j}\,(t)\right) + c_2 r_2 \left(p_{g,j}\,(t) - x_{i,j}\,(t)\right) \tag{11}$$

c_1, c_2 are two constants called acceleration coefficients. r_1, r_2 are two random numbers that existed in the interval [0,1], $v_{ij}(t)$ corresponds to the physical component of the displacement.

The position of particle i is then defined by:

$$X_i(t+1) = x_i(t) + v_i(t+1) \tag{12}$$

the particle swarm is usually represented by a geometric model, assuming that *v* is the velocity of the particle, *x* is the initial position of the particle and p is the optimal position of the particle. We also suppose that the particle swarm is composed of *N* particles. Briefly, in the process of finding the optimal solution, each particle modifies its position and velocity iteratively. That is to say, the updating of the position and velocity of the particle is based on its previous information and the previous best position found by the swarm. The geometric model illustrated in Figure 2 depicts the movement strategy of the particle. The pseudo-code mentioned in Algorithm 2 describes the steps of PSO algorithm.

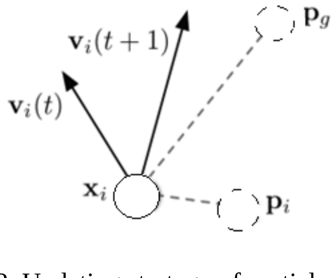

Figure 2. Updating strategy of particle position

Algorithm 2. Particle swarm optimization algorithm

```
Randomly initialize Ps particles: Position and velocity
Assess particle position
while the stopping criterion is not reached
        for i=1,...,Ps
           move the particles according (11), (12)
           if f(xi)<f(pi)
             pi= xi
             if f(xi)<f(pg)
               pg= xi
             end
          end
       end
end
```

In the beginning, we calculate the average hopsize. Then, we calculate the distance between anchors and nodes. In addition, we use PSO algorithm in order to refine results. Indeed, we update the position and the best position of PSO in each iteration of PSO algorithm until we reach the number of iterations. The final solution extracted by PSO denotes the optimal coordinates of the global algorithm PSO-DVHOP. The flowchart indicated in Figure 3 describes in detail the functioning of PSO-DVHOP.

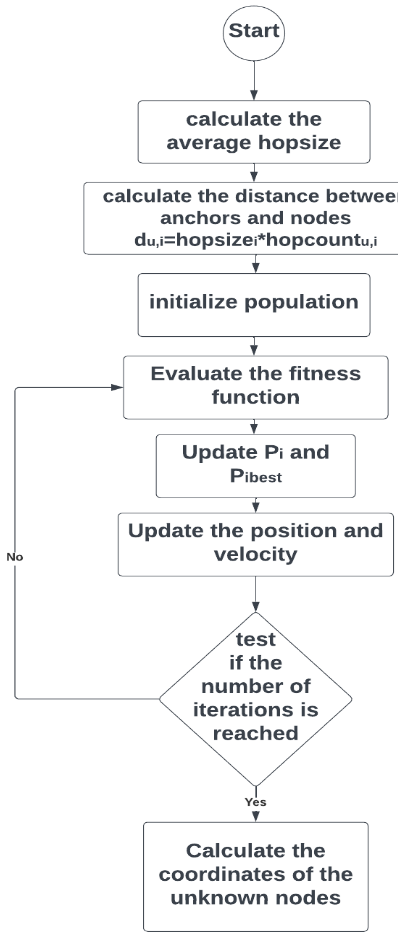

Figure 3. Flow chart of PSO-DVHOP algorithm

4.4. DVHOP algorithm based Fmincon solver

Fmincon solver is presented among the solvers that belong to the library of optimization integrated in MATLAB. Moreover, this predefined function allows us to find the minimum of a constrained non-linear multi-variable function using the interior-point algorithm. However, Fmincon depends on describing the cost function and all accompanying information, such as the initial point of the algorithm and the constraints.

Indeed, the cost function is in the form of @objfun. Also, the constraints and bounds of constraints are represented by the adequate matrix.

There are two approaches for using Fmincon: either in a graphical mode presented by a window of MATLAB with different fields that allow the user to insert the required information (function, start point, and constraints) or by calling Fmincon via the command window of MATLAB. In this case, we must specify the cost function in script.m and we call the constraints and the initial point via command window. In both cases of utilization, when we run the solver, the results are shown, including the reason the algorithm terminated. Obviously, the results denote the final point reached. In our case, we have used the solver via command window. However, our purpose was to resolve the optimization problem formulated as (13).

$$\begin{cases} \min f(x) \\ \text{subject to} \\ A.x \le b \\ Aeq.x = beq \\ lb \le x \le ub \end{cases} \tag{13}$$

Where:
A, *Aeq* denote the matrices of constraints
b, *beq* denote the vectors of constraints
lb, *ub* the upper and lowest values taken by *x*

Seeing that, in the field of sensing, a sensor may be placed without any predefined condition, we haven't set any constraints on our optimization problem. Also, we set the dimension of the problem at two because we do the localization in two dimensions. Hence, *x* is designed by [*x(1) x(2)*]. In addition to that, we refer the upper and lower values taken by *x* to the lower and upper abscissa and ordinate taken by a sensor in the sensing field. In our case, our field has 1000×1000 as a surface, so *lb*, *ub* take respectively the values [0,0] and [1000,1000].

Among the predefined solvers in MATLAB (linprog, ga, fminimax), we have implicitly opted for Fmincon because the latter may be set without gradient supply calculation. Hence, we avoid the additional complexity caused by the derivative calculation. Consequentially, Fmincon makes the optimization in a short period of time in comparison with their variants. Secondly, in both cases of utilization cited above (command window, graphical mode) this special function is simple to implement. Indeed, it just needs to assign each of their attributes properly. Thirdly, Fmincon solver is an efficient tool to avoid being trapped in premature convergence despite the multi-modality presented in our cost function dedicated to optimizing DVHOP. The pseudo-code mentioned in Algorithm 3 describes the steps of FMINCON-DVHOP algorithm.

Algorithm 3. FMINCON-DVHOP algorithm

```
Initialization:
number of nodes =NB,
number of anchors=NA,
area of experimentation =1000×1000 m2,
communication range=500 m
1.calculation of hopcounti,j by finding the shortest path between nodes
for k=1 to NB
   for i =1 to NB
      for j=1 to NB
         if(shortest_path(i,j)> shortest_path(i,k)+ shortest_path(k,j) )
            shortest_path(i,j)= shortest_path(i,k)+shortest_path(k,j);
         end
      end
   end
end
2.hopsize calculation according (1)
3.calculate the positions of unknown nodes
for i=NA to NB
   4.distance calculation
   unknown_to_anchors_dist=hopsize(i) × shortest_path(i,1 to NA);
   5.fitness function f is calculated according (6)
   A=[]; b=[]; Aeq=[]; beq=[]; x0=[0 0];
    lb=[0 0];  ub=[10001000];
   6.assign for each unknown node the result of fmincon
   node.estimated(i,1to 2) = fmincon(f,x0,A,b,Aeq,beq,lb,ub);
end
```

5. RESULTS AND DISCUSSIONS

In this section, we compare the performance of SA-DVHOP, PSO-DVHOP, FMINCON-DVHOP, and DVHOP in terms of accuracy. It's worth mentioning that there are several metrics to judge the quality of the localization algorithm, such as coverage of localization, consumption of energy. Also, it exists several works to assess the algorithms in terms of complexity of calculation. That's to say they use the metric of complexity in order to deduce the time of execution required by the algorithm.

According to our experimentation, DVHOP successfully locates the whole sensor nodes with a reasonable value of parameters (number of nodes, anchor ratio) in several scenarios of simulation. That's to say DVHOP has a high coverage of localization. Also, we assume that the variants SA-DVHOP, PSO-DVHOP, and FMINCON-DVHOP have the same advantage because those improved versions of DVHOP keep the two steps of the traditional DVHOP. Consequently, we don't take into account the evaluation of coverage of localization, and we assess our localization techniques just according to their localization accuracy in a WSN with a uniform and random distribution of nodes. In detail, in our simulations, we first considered a network with a random distribution of nodes. Also, we have assessed our algorithms in grid topology. That is to say, the area of simulation is partitioned into grids, and nodes and anchors are equally distributed throughout these grids. The criterion of comparison is ALE in order to select which localization algorithm is better in a specified configuration in terms of accuracy.

In order to ameliorate DVHOP, we have created three improved versions of the traditional DVHOP. Indeed, we make simulation in two dimensions; for SA-DVHOP we initialize the temperature at 0.1 and the number of neighbors per individual at 5. For PSO-DVHOP, we use 50 individuals, and we initialize the cognitions coefficients at determinate values. Tables 2 and 3 describe the parameter settings of SA-DVHOP and PSO-DVHOP.

Table 2. Parameter settings of SA-DVHOP

Parameter	Value
Dimension	2
Lower bound	0
Upper bound	1,000
Number of iterations	10
Initial temperature	0.1
α	0.99
Population size	10
Number of neighbors per individual	5

Table 3. Parameter settings of PSO-DVHOP

Parameter	Value
Population size	50
Number of iterations	100
c1	1.775
c2	2.8
Dimension	2
Lower bound	0
Upper bound	1,000

To evaluate the performance of each localization algorithm in terms of accuracy of localization. The following metric has been considered: ALE which is the ratio of total localization error to the number of simple nodes. Indeed, ALE is used to assess the precision of each localization algorithm according to different parameters such as node density, anchor node ratio and shape of distribution. Indeed, a specified algorithm is more accurate if it has less ALE. ALE can be expressed as (14).

$$ALE = \frac{\sqrt{(x_t-x_e)^2+(y_t-y_e)^2}}{(n_t-n_h)r} \quad (14)$$

Where (x_t, y_t) and (x_e,y_e) are the true and estimated coordinates of sensor nodes respectively.
n_t denotes the total number of nodes.
n_h presents the non-localized nodes.
r presents the communication range of a node.

5.1. Simulation results

To evaluate the performance of the cited algorithms, we split the simulation scenarios into two parts. Firstly, we make our comparison under a WSN with a grid topology and we vary the percentage of anchors and the number of nodes. In the second part, we remake the comparison with a random topology by varying the same metrics. It's assumed that when we pass from grid topology to random topology, the error of localization increases because the calculation of hop-size by all the algorithms in grid topology is more accurate than that calculated in random topology.

In our simulation, we use a surface whose surface equals 1000×1000 m^2. Also, we note that we use a regular model of communication and the communication range is equal to 400 m. We use the parameter settings listed in the Table 4.

Table 4. Parameter settings of simulations

Parameter	Value
Area	1000×1000 m^2
Total number of nodes	16-25-36-49-64-81
Topology	Uniform, irregular
Percentage of anchors	5%-10%-15%-20%-25%-30%
Communication range	400 m
Model of communication	Regular

In both cases of distribution (uniform and no-uniform) we followed the described strategy to make our evaluation: in the first step, we keep the number of nodes at 36 and the communication range at 400 m and we gradually vary the percentage of anchors. Secondly, we keep the percentage of anchors at 20% and we set the communication range to 400 m. Then, we vary the number of nodes between 16 and 81.

In Figure 4, we can see the initial deployment of nodes throughout the field of sensing with a total number of nodes equal to 25 (5 anchors and 20 unknown nodes). The percentage of anchors is expressed by the ratio of the number of anchors to the total number of nodes. In Figure 4(a), we use a uniform distribution of nodes throughout an area whose surface is equal to 1000×1000 m^2. In Figure 4(b) we keep the same parameters described in Figure 4(a) the only difference is that we change the topology of the network to a random topology.

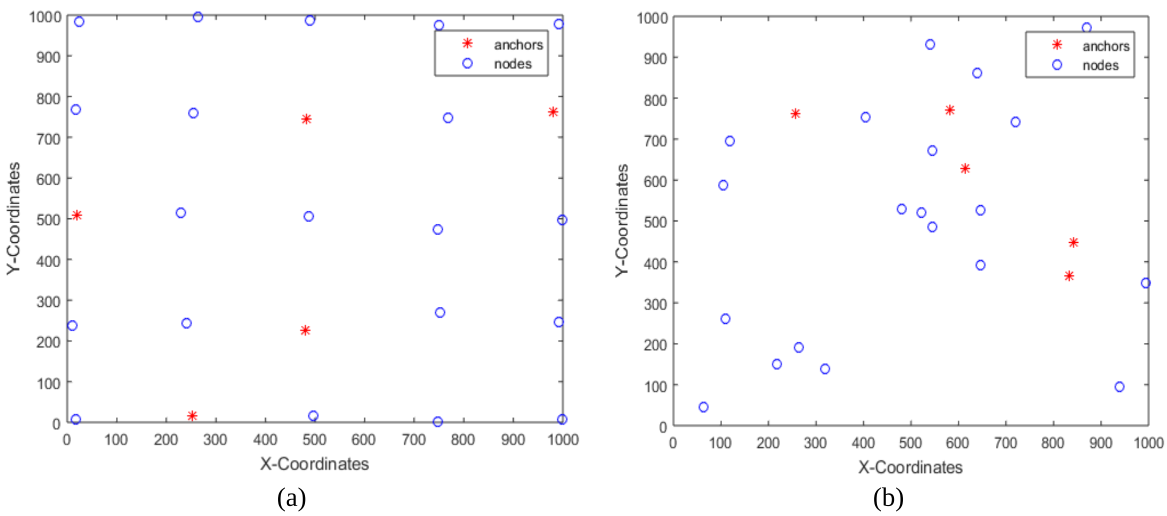

Figure 4. The initial deployment of 25 nodes (percentage of anchors=20%): (a) grid topology and (b) random topology

5.2. Discussions

5.2.1. The comparison under a uniform deployment of nodes

In this part, we shall establish a performance comparison of our algorithms by varying the percentage of anchors. We assume that the number of nodes is 36 and the communication range is set to 400 m. According to the results of Figure 5, it's shown that DVHOP gives the worst result and its performance is increasing by varying the percentage of anchors, because adding more anchors makes the multilateration done by DVHOP more precise. Consequently, DVHOP tends to locate the unknown nodes with great accuracy by increasing the number of anchor nodes.

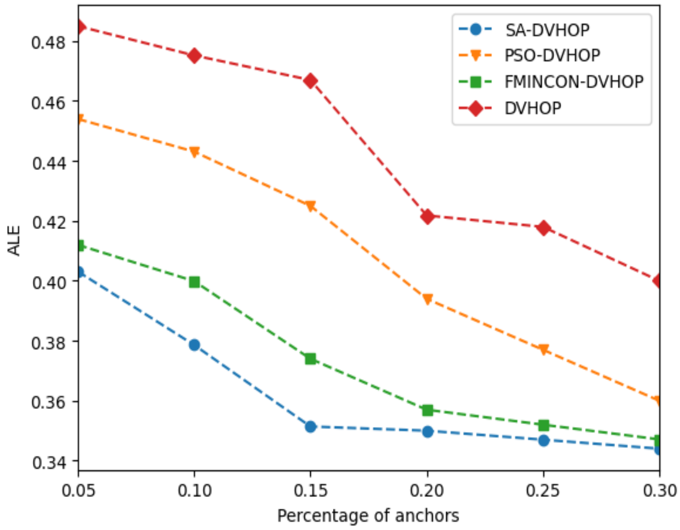

Figure 5. Average localization error with variation in percentage of anchors

On the other hand, it is evident that SA-DVHOP yields the most favorable outcomes, particularly due to its increasing accuracy within the 5% to 15% range of anchor percentage. However, in the 15% to 30% range, the precision of SA-DVHOP experiences a slight decrease. This can be attributed to the efficient cooling scheme employed by SA, wherein the atoms in the alloy are granted freedom of displacement. The temperature is gradually reduced until a static equilibrium of atoms is achieved. In cases where equilibrium is not reached, corrections are made by raising the temperature and slowly cooling the alloy. This process aligns with our cost function, resulting in solutions extracted by SA-DVHOP closely approximating the true solutions. Consequently, SA-DVHOP offers superior results compared to other algorithms. Additionally, both FMINCON-DVHOP and PSO-DVHOP also deliver commendable outcomes, with their accuracy continuing to improve as more anchors are incorporated. In the configuration shown in Figure 6, the communication range is set to 400 m, the percentage of anchors is set to 20% and we change the number of nodes between 16 nodes and 81 nodes. The results of the experiment are shown in Figure 6.

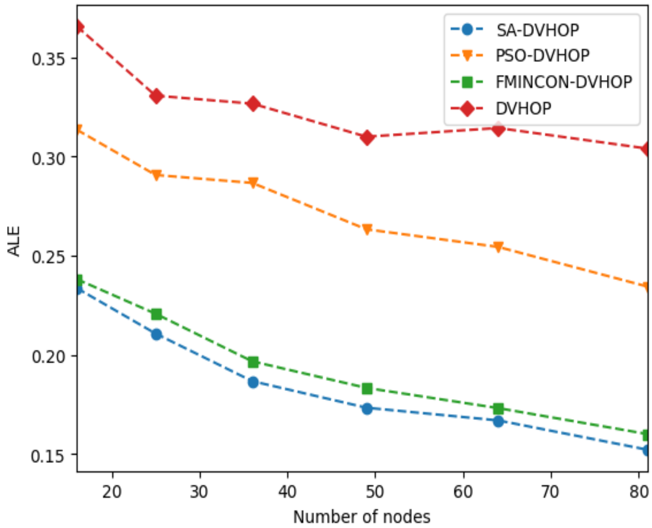

Figure 6. Average localization error with variation in number of nodes with communication range=400 m

According to Figure 6, SA-DVHOP offers superior results compared to other algorithms when we increase the number of nodes. However, in the beginning, precisely within the 16–36 range of nodes, the precision of Fmincon-DVHOP and SA-DVOP experiences a significant decrease. However, in the 36–81 range, the variation of their accuracy shows a slight decrease because 36 nodes denote the number of nodes in which the cited algorithms perform better. In other hand, it's observed that PSO-DVHOP gives a good result and its ALE keeps decreasing, but its performance remains less than that offered by the two mentioned algorithms. In addition to that, it's clearly shown that DVHOP causes a huge error in estimating the position of unknown nodes and it's observed different transitions in its performance variation. That reflects the non-stability in the calculation of least square adopted by the algorithm. Finally, it's concluded that in WSN with a uniform distribution of nodes, the optimization by the meta-heuristic methods has shown efficiency in enhancing the precision of the traditional DVHOP.

5.2.2. The comparison under a random distribution of nodes

In this part, we research the impact of changing the number of anchors on the accuracy of each algorithm. Also, it's noted that the number of nodes is 36 and the communication range is 400 m. According to the results of Figure 7, it's shown that our SA-DVHOP outperforms other localization algorithms when we increase the percentage of anchors because SA-DVHOP keeps the good properties of DVHOP in distance calculation. Additionally, SA-DVHOP shows better results due to the efficiency of SA for resolution purposes. Also, it's noted that Fmincon-DVHOP gives a good result, which means Fmincon is an efficient tool to avoid being trapped in premature convergence. However, the precision offered by SA-DVHOP remains higher than that offered by Fmincon-DVHOP. On the other hand, PSO-DVHOP is ranked in the third position, which means the exploration and exploitation adopted by PSO meta-heuristic suffer from a leak of balance that negatively impacts the resolution of the optimization problem. Consequently, this method doesn't provide the same precise results as the mentioned algorithms. Finally, it's observed that DVHOP gives the worst result when we vary the percentage of anchors.

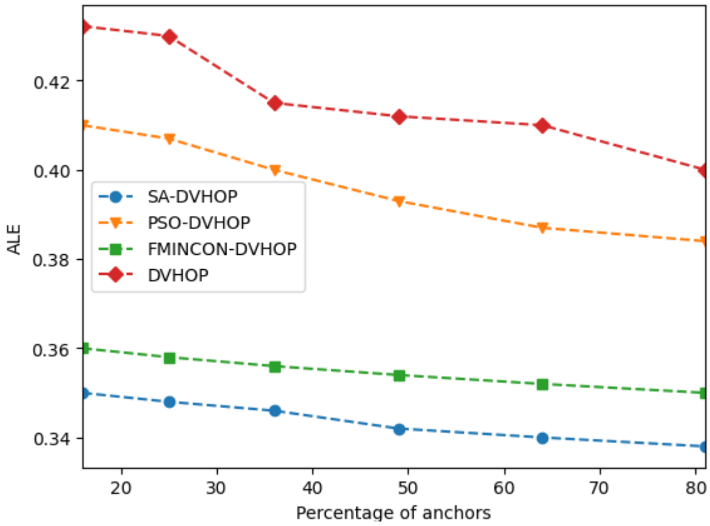

Figure 7. Average localization error with variation in percentage of anchors

In the configuration shown in Figure 8, the communication range is set to 400 m, the percentage of anchors is set to 20% and we change the number of nodes between 16 nodes and 81 nodes. The results of the experiment are shown in Figure 8. According to the results of Figure 8, it's clearly observed that the accuracy of each algorithm increases with increasing communication range. Also, it’s shown that SA-DVHOP outperforms other algorithms. Besides that, it's marked out that DVHOP gives the worst results. Indeed, in a random deployment of nodes, the algorithm is prone to imprecision in averaging the hop-size, which leads the algorithm to calculate imprecisely the distance between unknown nodes and anchors. Hence, the accuracy provided by DVHOP remains less than that offered by the improved algorithms. On the other hand, the localization error of PSO-DVHOP decreases as the number of nodes increases because this algorithm uses a meta-heuristic technique to locate each node instead of adopting the least square method, which is judged to be the reason for the imprecision of DVHOP.

As shown in Figure 8, when the communication range is set to 400 m, it's clearly observed that both Fmincon-DVHOP and SA-DVHOP give the best results and SA-DVHOP is more precise than Fmincon-DVHOP. That reflects the efficient cooling scheme adopted by SA. The accuracy of SA-DVHOP is relatively not sensitive to the additional node for showing a satisfactory result. Consequently, SA-DVHOP is judged to be the most economical and precise algorithm among their variants. Table 5 summarizes previous studies that used optimization approaches to improve DVHOP performance.

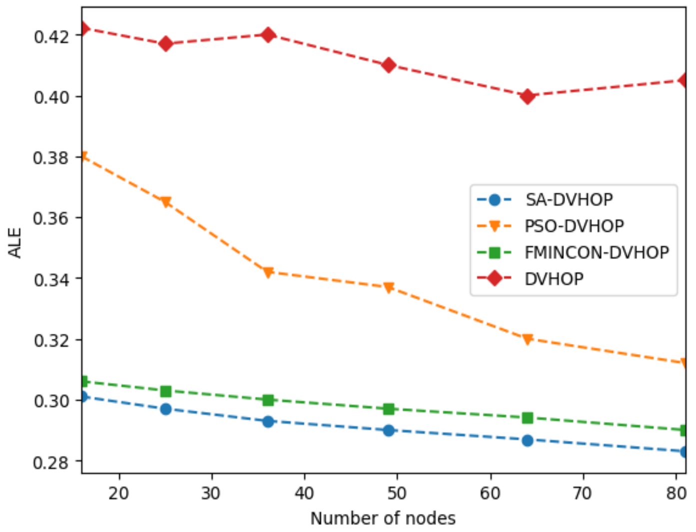

Figure 8. Average localization error with variation in number of nodes with communication range=400

Table 5. Summarizes the relevant works

Researchers	Algorithm	Network settings	Metric	Value
Sharma and Kumar [30]	DVHOP with genetic algorithm in 3D	Random deployment of nodes with variation of percentage of anchors	Average localization error (m)	2.48
		Random deployment of nodes by varying the number of nodes	Average localization error (m)	2.86
Cheng *et al.* [34]	DVHOP with Archimedes algorithm	Random deployment of nodes by varying the number of nodes	Average localization error (m)	0.30
		Random deployment by varying the percentage of anchors	Average localization error (m)	0.37
Zhang *et al.* [35]	DVHOP with quantum-behaved PSO	Varying the communication range with uniform deployment of nodes	Average localization error (m)	0.21
		Varying the number of anchors with uniform deployment of nodes	Average localization error (m)	3.68
		Varying the node density with uniform deployment of nodes	Average localization error (m)	0.18

6. CONCLUSION

In order to ameliorate the performance of DVHOP, this paper proposes an enhanced DVHOP on the basis of a simulated annealing algorithm. Indeed, the main idea of simulated annealing algorithm is to heat and cool slowly the metal until we have a compact solid, and if we don't get a good quality of solid, another heating-cooling process is executed. So, this strategy followed by SA has served us to construct an efficient mathematical tool that is used to optimize the cost function of DVHOP. On the other hand, we have also created another improved version of DVHOP by using PSO and Fmincon. Our simulation consists of comparing the performance of all the improved versions of DVHOP and the traditional DVHOP. The results prove that SA-DVHOP gives the best accuracy in comparison with DVHOP and the others mentioned algorithms.

In our research, we have considered that the communication occurred without irregularity of radio patterns. However, it's proven that when irregularities increase, precision of locating decreases. Most researchers don't take this into account. Hence, it's suggested to bring another analysis study of DVHOP aiming to improve it in a field of sensing with irregular radio patterns as future work.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Omar Arroub	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Anouar Darif		✓				✓		✓	✓	✓	✓	✓		
Rachid Saadane	✓		✓	✓			✓			✓	✓		✓	✓
My Driss Rahmani		✓				✓		✓	✓	✓	✓	✓		
Zineb Aarab					✓		✓			✓		✓		✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] D. Kandris, C. Nakas, D. Vomvas, and G. Koulouras, "Applications of wireless sensor networks: an up-to-date survey," *Applied System Innovation*, vol. 3, no. 1, p. 14, Feb. 2020, doi: 10.3390/asi3010014.
[2] S. L. Ullo and G. R. Sinha, "Advances in smart environment monitoring systems using IoT and sensors," *Sensors*, vol. 20, no. 11, p. 3113, May 2020, doi: 10.3390/s20113113.
[3] A. K. Sangaiah, A. Javadpour, C.-C. Hsu, A. Haldorai, and A. Zeynivand, "Investigating Routing in the VANET Network: Review and Classification of Approaches," *Algorithms*, vol. 16, no. 8, p. 381, Aug. 2023, doi: 10.3390/a16080381.
[4] N. B. Oğur, M. Al-Hubaishi, and C. Çeken, "IoT data analytics architecture for smart healthcare using RFID and WSN," *ETRI Journal*, vol. 44, no. 1, pp. 135–146, Feb. 2022, doi: 10.4218/etrij.2020-0036.
[5] A. Abdollahi, K. Rejeb, A. Rejeb, M. M. Mostafa, and S. Zailani, "Wireless sensor networks in agriculture: insights from bibliometric analysis," *Sustainability*, vol. 13, no. 21, p. 12011, Oct. 2021, doi: 10.3390/su132112011.
[6] W. Liang, K.-C. Li, J. Long, X. Kui, and A. Y. Zomaya, "An industrial network intrusion detection algorithm based on multifeature data clustering optimization model," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 2063–2071, Mar. 2020, doi: 10.1109/TII.2019.2946791.
[7] I. T. Almalkawi, M. Guerrero Zapata, J. N. Al-Karaki, and J. Morillo-Pozo, "Wireless multimedia sensor networks: current trends and future directions," *Sensors*, vol. 10, no. 7, pp. 6662–6717, Jul. 2010, doi: 10.3390/s100706662.
[8] M. Li and Y. Liu, "Rendered path: range-free localization in anisotropic sensor networks with holes," in *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, Sep. 2007, pp. 51–62, doi: 10.1145/1287853.1287861.
[9] O. Cheikhrouhou, G. M. Bhatti, and R. Alroobaea, "A Hybrid DV-hop algorithm using RSSI for Localization in large-scale wireless sensor networks," *Sensors*, vol. 18, no. 5, p. 1469, May 2018, doi: 10.3390/s18051469.
[10] S.-T. Chen, C. Zhang, P. Li, Y.-Y. Zhang, and L.-B. Jiao, "An indoor collaborative coefficient-triangle APIT localization algorithm," *Algorithms*, vol. 10, no. 4, p. 131, Nov. 2017, doi: 10.3390/a10040131.
[11] A. Kaur, P. Kumar, and G. P. Gupta, "A weighted centroid localization algorithm for randomly deployed wireless sensor networks," *Journal of King Saud University - Computer and Information Sciences*, vol. 31, no. 1, pp. 82–91, Jan. 2019, doi: 10.1016/j.jksuci.2017.01.007.
[12] W. Liu, J. Li, A. Zheng, Z. Zheng, X. Jiang, and S. Zhang, "DV-hop algorithm based on multi-objective salp swarm algorithm optimization," *Sensors*, vol. 23, no. 7, p. 3698, Apr. 2023, doi: 10.3390/s23073698.
[13] S. Messous and H. Liouane, "Online Sequential DV-hop localization algorithm for wireless sensor networks," *Mobile Information Systems*, vol. 2020, pp. 1–14, Oct. 2020, doi: 10.1155/2020/8195309.
[14] S. Shen, K. Qian, B. Yang, Y. She, W. Wang, and Y. Wang, "An improved amorphous algorithm in wireless sensor network based on approximate equilateral triangle beacon selection," in *2017 International Conference on Networking and Network Applications (NaNA)*, Oct. 2017, pp. 54–60, doi: 10.1109/NaNA.2017.30.
[15] J. Luomala and I. Hakala, "Adaptive range-based localization algorithm based on trilateration and reference node selection for outdoor wireless sensor networks," *Computer Networks*, vol. 210, p. 108865, Jun. 2022, doi: 10.1016/j.comnet.2022.108865.
[16] K. Yu, Y. J. Guo, and M. Hedley, "TOA-based distributed localisation with unknown internal delays and clock frequency offsets in wireless sensor networks," *IET Signal Processing*, vol. 3, no. 2, pp. 106–118, Mar. 2009, doi: 10.1049/iet-spr:20080029.

[17] H. Li and Z. Cheng, "Angle-of-arrival estimation using difference beams in localized hybrid arrays," *Sensors*, vol. 21, no. 5, p. 1901, Mar. 2021, doi: 10.3390/s21051901.
[18] W. Li and B. Zhao, "Analysis of TDOA location algorithm based on ultra-wideband," in *Communications, Signal Processing, and Systems: Proceedings of the 8th International Conference on Communications, Signal Processing, and Systems 8th*, 2020, pp. 1257–1261, doi: 10.1007/978-981-13-9409-6_149.
[19] P. Gonzalez-Ayala, A. Alejo-Reyes, E. Cuevas, and A. Mendoza, "A modified simulated annealing (MSA) algorithm to solve the supplier selection and order quantity allocation problem with non-linear freight rates," *Axioms*, vol. 12, no. 5, p. 459, May 2023, doi: 10.3390/axioms12050459.
[20] J. Wang, Y. Zhu, C. Zhou, and Z. Qi, "Construction method and performance analysis of chaotic s-box based on a memorable simulated annealing algorithm," *Symmetry*, vol. 12, no. 12, pp. 1–14, 2020, doi: 10.3390/sym12122115.
[21] M. Jain, V. Saihjpal, N. Singh, and S. B. Singh, "An overview of variants and advancements of PSO algorithm," *Applied Sciences*, vol. 12, no. 17, p. 8392, Aug. 2022, doi: 10.3390/app12178392.
[22] G. Li, T. Wang, Q. Chen, P. Shao, N. Xiong, and A. Vasilakos, "A survey on particle swarm optimization for association rule mining," *Electronics*, vol. 11, no. 19, p. 3044, Sep. 2022, doi: 10.3390/electronics11193044.
[23] M. Sertsöz and M. Fidan, "A comparison of PSO and Fmincon methods for finding optimum operating speed and time values in trams," *International Journal of Energy Applications and Technologies (IJEAT)*, vol. 8, no. 2, pp. 48–52, Jun. 2021, doi: 10.31593/ijeat.799129.
[24] A. Vagaská, M. Gombár, and Ľ. Straka, "Selected mathematical optimization methods for solving problems of engineering practice," *Energies*, vol. 15, no. 6, p. 2205, Mar. 2022, doi: 10.3390/en15062205.
[25] S. Rajendran, G. N., R. Čep, N. R. C., S. Pal, and K. Kalita, "A conceptual comparison of six nature-inspired metaheuristic algorithms in process optimization," *Processes*, vol. 10, no. 2, p. 197, Jan. 2022, doi: 10.3390/pr10020197.
[26] D. Xue, "Research on range-free location algorithm for wireless sensor network based on particle swarm optimization," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, p. 221, Dec. 2019, doi: 10.1186/s13638-019-1540-z.
[27] S. Kessentini and D. Barchiesi, "Particle swarm optimization with adaptive inertia weight," *International Journal of Machine Learning and Computing*, vol. 5, no. 5, pp. 368–373, Oct. 2015, doi: 10.7763/IJMLC.2015.V5.535.
[28] G. Sharma and A. Kumar, "Improved range-free localization for three-dimensional wireless sensor networks using genetic algorithm," *Computers & Electrical Engineering*, vol. 72, pp. 808–827, Nov. 2018, doi: 10.1016/j.compeleceng.2017.12.036.
[29] S. Phoemphon, C. So-In, and N. Leelathakul, "Optimized hop angle relativity for DV-hop localization in wireless sensor networks," *IEEE Access*, vol. 6, pp. 78149–78172, 2018, doi: 10.1109/ACCESS.2018.2884837.
[30] D. Perdana, A. Nugroho, and F. Dewanta, "Performance evaluation of DV-HOP and amorphous algorithms based on localization schemes in wireless sensor networks," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 16, no. 3, pp. 1150–1157, Jun. 2018, doi: 10.12928/telkomnika.v16i3.8560.
[31] A. Ali, Y. Ming, S. Chakraborty, and S. Iram, "A comprehensive survey on real-time applications of WSN," *Future Internet*, vol. 9, no. 4, p. 77, Nov. 2017, doi: 10.3390/fi9040077.
[32] D. Han, Y. Yu, K.-C. Li, and R. F. de Mello, "Enhancing the sensor node localization algorithm based on improved DV-hop and DE algorithms in wireless sensor networks," *Sensors*, vol. 20, no. 2, p. 343, Jan. 2020, doi: 10.3390/s20020343.
[33] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983, doi: 10.1126/science.220.4598.671.
[34] M. Cheng, T. Qin, and J. Yang, "Node localization algorithm based on modified archimedes optimization algorithm in wireless sensor networks," *Journal of Sensors*, vol. 2022, pp. 1–18, Jun. 2022, doi: 10.1155/2022/7026728.
[35] D. Zhang, X. Zhang, and H. Qi, "A new location sensing algorithm based on dv-hop and quantum-behaved particle swarm optimization in WSN," *ASP Transactions on Pattern Recognition and Intelligent Systems*, vol. 1, no. 2, pp. 1–17, May 2021, doi: 10.52810/TPRIS.2021.100034.

BIOGRAPHIES OF AUTHORS

Omar Arroub received the bachelor in computer sciences and mathematics from Faculty of Sciences Rabat in 2011. He received the master in computer sciences and networks from Faculty of Sciences and techniques of Tanger in 2014. He is currently a Ph.D. student at University Mohamed V, Morocco. His research interests include WSN, internet of things (IoT), metaheuristic, and machine learning. He can be contacted at email: omar_arroub@um5.ac.ma.

Aouar Darif received the Diplôme d'Etudes Supérieurs Approfondies in Computer Sciences and Telecommunications from Faculty of Sciences Rabat in 2007. He received the Ph.D. degree in computer sciences and telecommunications from Faculty of Sciences of Rabat in 2015. He is currently research and a teaching associate in Multidisciplinary Faculty at University of Sultan Moulay Slimane Beni Mellal, Morocco. His research interests include WSN, mobile edge computing (MEC), IoT, cloud computing, and neural networks. He is an active reviewer of various international conferences and journals. He can be contacted at email: anouar.darif@usms.ac.ma.

Rachid Saadane received the B.S. degree in physic electronic from the Faculty of Science of Rabat, Rabat, Morocco, in 2001. He received the Diplôme d 'Etudes Supérieurs Approfondies in computer sciences and telecommunications from the Faculty of Sciences of Rabat, in 2003. He received the Ph.D. degree in computer sciences and telecommunications from Faculty of Sciences of Rabat jointly with Eurecom Institute in 2007. He currently works as a research and a teaching associate at the EHTP. He can be contacted at email: saadane@ehtp.ac.ma.

My Driss Rahmani is full professor of Computer Science in Mohammed V University in Rabat (Morocco) where he was head of Department of Computer Science until 2015. He holds a Ph.D. in sciences from University of Montpellier 2 (France) in 1989. He has over 25 years of teaching experience (concurrent programming, graphic user interface, compilation, and XML technology). His main area of interest includes WSN, urban modelling, car- following modelling, business process modelling, and smart cities. He can be contacted at email: d.rahmani@um5r.ac.ma.

Zineb Aarab is an assistant professor of computer science and applied computer science at Mohammadia School of Engineers (EMI). She is a collaborator researcher at Computer Science and Telecommunications Research Laboratory LRIT. Her doctoral thesis has been prepared in the Computer Science and Telecommunications Research Laboratory LRIT. Her research interests lie in model-driven engineering (MDE) and information systems (IS) with a focus on context awareness. She can be contacted at email: aarab@emi.ac.ma.