Journal of Universal Computer Science, vol. 29, no. 12 (2023), 1535-1553
submitted: 19/3/2023, accepted: 5/7/2023, appeared: 28/12/2023

Wireless Sensor Network Coverage Optimization for Internet of Things

Yunwu Xu
(Electrical Engineering College, Guangdong Songshan Polytechnic, Shaoguan, China
https://orcid.org/0009-0000-7945-1995, yunwuxu7@gmail.com)

Yan Li
(School of Computer and Information Engineering, Guangdong Songshan Polytechnic, Shaoguan, China
https://orcid.org/0009-0004-7398-2874, yanli794@yahoo.com)

Abstract: The objective of this work is to improve the existing Wireless Sensor Network coverage optimization method. The pigeon-inspired optimization algorithm was first evaluated, and its shortcomings were noted. The pigeon-inspired optimization method was then enhanced with the good point set, Yin-Yang optimization algorithm, and opposition-based learning. To test the improved algorithm, five representative standard functions were chosen: sphere function (f1), Rosenbrock function (f2), Levy function (f3), Schwefel function (f4), and Levy function N.13 (f5). The algorithm's speed of convergence may be determined by the first two functions, which are unimodal. The final three functions, which are multimodal, can extract several local optimal values from the local optimum. In comparison with other known algorithms, the improved Yin-Yang PIO algorithm showed the highest optimization accuracy and stability. Three sets of experiments were performed to optimize the WSN coverage with different parameters. The first series of experiments suggest that Yin–Yang PIO has the best optimization effect, with a coverage rate of 99.51% (10.22% higher with PIO and 6.41% higher compared with PSO). The second and third series of experiments show that Yin-Yang PIO significantly increased the WSN coverage ratio, up to 99.9%. The algorithm can be applied to optimize WSN coverage in various environments. Future research can extend the research scope to include other optimization problems in IoT.

Keywords: Algorithm optimization, Pigeon-inspired optimization, Opposition-based learning, Coverage ratio, Good points set, Coverage efficiency
Categories: H.3.5, J.7
DOI: 10.3897/jucs.103738

1 Introduction

The penetration of the Internet of things (IoT) technologies into the everyday life of people is currently rapid [Serror et al., 20]. With the advent of the fifth generation (5G) mobile communication technology, this process became even faster [Zhou et al., 21]. One of the core components of the Internet of Things is the Wireless Sensor Network (WSN) - a network made up of several resource-constrained sensor nodes [Khan et al., 15, Kintonova et al., 22, Shi et al., 12]. The military, healthcare industry, and weather monitoring all employ WSN extensively [Bhardwaj and Kumar, 19]. The full coverage of the surveillance area is desired, but it requires a large number of nodes to be deployed. Not only it is costly, but it may also lead to communication conflicts. This problem calls for an algorithm that would allow covering a large area with fewer nodes [Akhmetbaev et al., 19, Beketaeva et al., 19, Liu et al., 21]. The practical significance of this study lies in the fact that the proposed algorithm requires fewer wireless network nodes, thereby reducing the operating costs and the product price.

1.1 Literature Review

The dynamic network coverage optimization based on the binary coverage model is the major focus of the present research on WSN coverage. Voronoi-based network coverage optimization is one illustration, although it has challenging theoretical and computational challenges [Sagynganova et al., 22, Sung and Yang, 14]. Another one uses sophisticated algorithms that stay clear of difficult theoretical deductions, such the firefly algorithm [Idrees and Couturier, 22], whale optimization method [Toloueiashtian et al., 22], and particle swarm optimization (PSO) [Zhang, 20]. These techniques are probably going to reach the local optimum.

A bionic intelligent optimization method having the benefits of a straightforward construction, simple comprehension, and powerful global search capability is the pigeon-inspired optimization algorithm. It is used at the moment for picture segmentation [Liu et al., 18], proportional integral derivative parameter modification [Sun and Duan, 14], and control of unmanned aerial vehicles [Duan et al., 15].

In order to control big civil airplanes, this algorithm has been used. In order to optimize the control settings of an autonomous landing system on an aircraft carrier, authors [Dou and Duan, 17] devised PIO based on the Levy flight. A cross PIO algorithm with cognitive aspects was suggested by the authors [Tao and Li, 18]. In this approach, the two operators were combined, and the map and compass operator was given a cognitive component in the form of a non-linear increase. The landmark operator also received a compression factor, which improved the algorithm's accuracy. The researchers [Wang et al., 19] created a cooperative PIO with a dynamic distance threshold to broaden the population's variety. After that, the technique was successfully used to solve the picture threshold segmentation problem. An enhanced PIO for the fuzzy variation operator was put out by authors [Xia et al., 22]. The fuzzy variation operator was added in this technique, which was influenced by the differential evolution algorithm, to enhance the position update formula and, as a result, the search capability.

YYPO is a novel intelligent optimization technique developed by authors [Punnathanam and Kotecha, 16] that aids in achieving Yin-Yang equilibrium in the search space. The opposition-based learning approach was added to the algorithm to undertake a centralized search of solutions that are in opposition to the existing ones. Authors [Xu et al., 20] suggested an enhanced YYPO algorithm based on chaotic search and the complex operator.

Such modification enhanced the optimization ability of the algorithm. YYPO was later combined with PSO to solve the uncapacitated warehouse location problems; the optimization results were satisfactory [Heidari et al., 17]. Authors [Wang et al., 21] combined YYPO with a seagull optimization algorithm, thereby enhancing the algorithm performance.

Research is currently underway to optimize WSN coverage using a metaheuristic algorithm. A coverage optimization technique based on an enhanced adaptive PSO was suggested by the authors [Wu et al., 16]. While the evolution factor and the aggregation factor in the inertial weight coefficient enhanced the adaptive ability of the algorithm, the collision resistance strategy introduced into the iteration process guaranteed the diversity of the swarm. Based on the augmented PSO algorithm, a new technique for WSN coverage was introduced [Kong and Yu, 19]. It is made up of a mutation operator and an inertia coefficient that speed up convergence and broaden the population, enhancing network coverage. In order to increase network coverage, researchers [Lu et al., 18] suggested a WSN coverage optimization strategy that involved changing the positions of two nodes.

The present study aims to enhance the existing WSN coverage optimization algorithm. To do this, the following objectives were established: (1) to assess the weaknesses of the existing PIO algorithm; (2) to improve the PIO algorithm by adding the good point set, the Yin-Yang optimization algorithm, and opposition-based learning; (3) to conduct a performance analysis of the improved algorithm on representative standard functions and experimental studies on the WSN coverage optimization problem.

The further structure of this article includes the following sections: Section 2 Materials and Methods (2.1 Pigeon-inspired Optimization Algorithm, 2.2 Yin-Yang Pair Optimization Algorithm, 2.3 Model of Wireless Sensor Network Coverage, 2.4 The suggested Strategy); Section 3 Results (3.1 Algorithm Performance Analysis, 3.2 Modeling of Wireless Sensor Network Coverage Optimization); Section 4. Conclusions and References.

2 Materials and Methods

2.1 Pigeon-inspired Optimization Algorithm

The homing habits of pigeons serve an inspiration for the PIO [Duan and Qiao, 14]. Pigeons frequently use the magnetic field, the sun, and landmarks when navigating. The pigeon utilizes the magnetic field and the sun to navigate when it is some distance from the destination place. It will use the well-known landmark to get there when it gets closer to the target site. The pigeon will follow the familiar pigeon if there is no known landmark. The PIO method contains two different types of operators, namely a map and compass operator and a landmark operator, in accordance with these pigeon behavior features.

Formula (1) is used to update the speed of the i-th pigeon in the D-dimensional choice space, while formula (2) is used to update the position (2). These two formulas are written as:

$$V_i(t) = V_i(t-1) \cdot e^{-Rt} + rand \cdot (X_g - X_i(t-1)) \quad (1)$$

$$X_i(t) = X_i(t-1) + V_i(t) \quad (2)$$

where $V_i(t)$ is the velocity of the i-th pigeon in the t-th generation, $V_i(t-1)$ is the velocity of the i-th pigeon in the generation $(t-1)$, R is a map and compass factor, $rand$ denotes a random number in the range of [0,1], X_g represents the global best solution, $X_i(t)$ is the position of the i-th pigeon in the t-th generation, $X_i(t-1)$ is the position of the i-th pigeon in the generation $(t-1)$.

Landmark operator: Individuals with poor fitness were eliminated, and each iteration update was cut in half in accordance with formula (3). The centered $X_c(t)$ and updated positions of the pigeons were then calculated according to formulas (4) and (5), respectively. The formulas were:

$$N_p(t) = \frac{N_p(t-1)}{2} \tag{3}$$

$$X_c(t) = \frac{\sum X_i(t) \cdot fitness(X_i(t))}{N_p \cdot \sum fitness(X_i(t))} \tag{4}$$

$$X_i(t) = X_i(t-1) + rand \cdot (X_c(t) - X_i(t-1)) \tag{5}$$

where $N_p(t)$ denotes the number of the pigeon in the t-th generation, $N_p(t-1)$ denotes the number of pigeons in the generation $(t-1)$, $X_i(t)$ is the position of the i-th pigeon in the t-th generation, $X_i(t-1)$ is the position of the i-th pigeon in the generation $(t-1)$, $fitness$ represents the fitness function.

2.2 Yin-Yang Pair Optimization Algorithm

All variables in this method were normalized between 0 and 1. The algorithm's optimization was based on an exchange mechanism between the points P_1 and P_2, where P_1 emphasizes local search and P_2 emphasizes global search. The hyperspheres' centers are P_1 and P_2, and their radii are δ_1 and δ_2. Take note of the periodic diminishing and growing patterns in δ_1 and δ_2, respectively.

The splitting stage and the archiving step are the two primary sections of the algorithm. The archive updates I were chosen between the integers I_{min} and I_{max} prior to the splitting step. Every time P_1 and P_2 travel through I splitting phases, they are updated with the $2I$ points and preserved in the archive.

Splitting stage. In the event that the dimension is D, just one point, designated as point P, will be divided at a time during the splitting step. There are two approaches to produce the updated points P:

One-way splitting: A $2D \times D$ matrix will serve as the storage location for the $2D$ identical copies, and each point in S will be updated in accordance with the following formula (6):

$$S_j^j = S^j + r\delta \text{ and } S_{D+j}^j = S^j - r\delta, \text{ where } j = 1,2,3 \dots D \tag{6}$$

When the search radius is denoted by, the subscript denotes the point number, the superscript is the decision variable number, and r is a random value between [0,1].

D-way splitting: Unlike one-way splitting, where only one variable is altered at each point, this splitting modifies all variables at each location. The points in S will now be updated in accordance with the formula, and a binary matrix B of dimension $2D \times D$ will be produced (7).

$$\begin{cases} S_k^j = S^j + r(\delta/\sqrt{2}) \text{ if } B_k^j = 1, \\ S_k^j = S^j - r\left(\frac{\delta}{\sqrt{2}}\right) \text{ else,} \\ \text{where } k = 1,2,3 \dots 2D \text{ and } j = 1,2,3 \dots D \end{cases} \tag{7}$$

Point P will be replaced with the updated point with the best fitness.

Archiving step. Picking the point from the archive with the best fitness is the first step. The selection procedure moves on to the remaining points in the archive if the point is superior than P_1. If it is, the two points exchange values. The values of the two points are switched if the point is superior to P_2.

δ_1 and δ_2 will be adjusted according to formulas (8) and (9):

$\delta_1 = \delta_1 - (\delta_1/\alpha)$ (8)

$\delta_2 = \delta_2 - (\delta_2/\alpha)$ (9)

where δ_1 denotes the search radius of the point P_1; δ_2 denotes the search radius of the point P_2; α is the factor for expansion and contraction.

2.3 Model of Wireless Sensor Network Coverage

N sensor nodes were positioned inside the monitoring area, each with a detecting radius of r and a transmission radius of R, supposing that the monitoring area is a $2D$ target plane. Normally, the transmission radius is set to $R = 2r$ to maintain network connectivity. The set of sensor nodes can be expressed using formula (10):

$$G = \{g_1, g_2, g_3, \dots, g_N\} \tag{10}$$

where (a_i, b_i) are the coordinates of each node.

$M \times n$ pixel points can be used to divide the monitoring region. The pixel point F coordinates were x_F, y_F, and formula (11) was used to calculate the distance between F and each sensor node g_i:

$$d(g_i, F) = \sqrt{(x_F - a_i)^2 + (y_F - b_i)^2} \tag{11}$$

where $d(g_i, F)$ denotes the distance from point F to sensor node g_i. Formula is used to determine the likelihood of being seen by the sensor node g_i at the pixel location F (12):

$$p(g_i, F) = \begin{cases} 1 \; d(g_i, F) \le r \\ 0 \; d(g_i, F) > r \end{cases} \tag{12}$$

where r is the sensing radius and $p(g_i, F)$ is the likelihood of being detected by the sensor node g_i at the pixel position F.

In real-world scenarios, each pixel may be experienced by many sensor nodes, and formula (13) illustrates the joint likelihood that a pixel F is perceived by a collection of nodes G:

$$p(G, F) = 1 - \prod_{g, \in G}[(1 - p(g_i, F)] \tag{13}$$

Formula (14) illustrates the coverage ratio of the sensor node placed in the monitoring area:

$$COV(G) = \frac{\sum_{F \in m \times n} p(G, F)}{m \times n} \tag{14}$$

where $m \times n$ is the total number of pixels, and $p(G, F)$ is the chance that the node set G will detect the pixel point F.

2.4 The Suggested Strategy

2.4.1 Set of good points

Individuals in the classic PIO are irregularly distributed in the solution space because the initial population is chosen at random. This effort concentrated on improving the initialization phase with a set of positive points since a population of pigeons that is evenly dispersed will increase the search capabilities.

G_d is a unit cube in the d-dimensional Euclidean space. Let $r \in G_d$ if

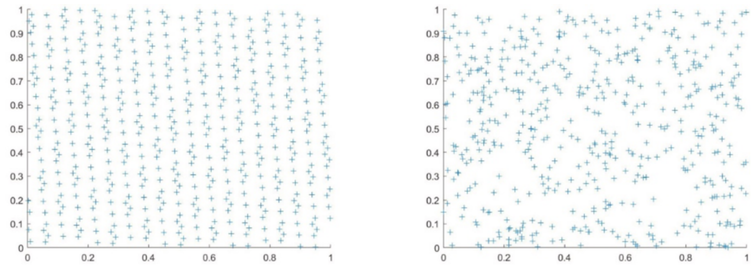
$$P_n(k) = \{\left(\left\{r_1^{(n)} \cdot k\right\}, \left\{r_2^{(n)} \cdot k\right\}, \dots \left\{r_s^{(n)} \cdot k\right\}\right), 1 \le k \le n\} \tag{15}$$

The deviation of formula (15) is equal to $\varphi(n)$, and $\varphi(n)$ satisfies the following equation: $\varphi(n) = C(r, \varepsilon)n^{-1+\varepsilon}$, where n is the number of nodes, $\{r_d^{(n)} \cdot k\}$ represents the fractional part, $C(r, \varepsilon)n^{-1+\varepsilon}$ is a constant related to r and ε, and ε is any positive number. $P_n(k)$ is a good point set and r is a good point taken as $r = \{2\cos\left(\frac{2\pi k}{p}\right), 1 \leq k \leq s\}$, where p is the smallest prime that satisfies $(p-3)/2 \geq s$. In formula (16), the enhanced initialization technique is displayed:

$$x_i(j) = \left(ub_j - lb_j\right) \cdot \left\{r_j^{(i)} \cdot k\right\} + lb_j \tag{16}$$

where ub_j is the upper bound of the j-th dimension, a lb_j is the lower bound of the j-th dimennsion.

Figure 1 displays 500 people who were produced using the random approach with the excellent point set in the range [0,1]. Figure 1(a) displays the outcome of the good point set, while Figure 1(b) displays the outcome of the random approach (b).



(a) Using good point set initialization (b) Using random initialization

Figure 1: Good Point Set Initialization Versus Random Initialization

When the excellent point set is utilized for initialization, as can be seen in Figure 1, the distribution of the pigeon population is more uniform. The number of people shown in Figure 1 is not significant and is illustrative.

2.4.2 Yang-Yin optimization technique

A schematic diagram of PIO combined with the Yin–Yang pair optimization algorithm (YYPO) is shown in Figure 2.

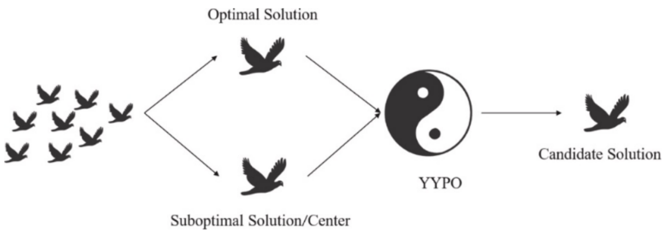


Figure 2: Schematic Diagram of Yin–Yang PIO

The location of the ideal solution has a significant influence on the early search since, in the map and compass operator, the individuals will approach the pigeon that has the finest fitness. The best and second-best points of the current population are used as beginning points P_1 and P_2 of YYPO before performing the map and compass operations. The dividing and archiving stages must then be entered, followed by a comparison between the outcome and the top member of the current population. The best member of the present population is replaced with the candidate solution if it is superior to the best individual.

Each iteration of the landmark operator will determine the pigeon population's center position, and the accuracy of this calculation will subsequently influence the algorithm's search performance. The ideal pigeon and the center position are utilized as the starting positions P_1 and P_2 of YYPO after each center position computation. The dividing stage and the archiving stage must then be entered, and the final step is to compare the outcome with the center position. The candidate solution is used to replace the center location of the existing population if it is better than the center position.

2.4.3 Opposition-based learning

The technique can swiftly converge if the points in the current population are close to the ideal point, but if they are distant from it, the process is more likely to become stuck at the local optimum. The current issue may be efficiently solved by simultaneously searching in opposing directions for the optimum answer. It was suggested to use opposition-based learning. The addition of opposition-based learning has improved a number of current techniques, including reinforcement learning, artificial neural networks, intelligent algorithms, etc. [Mahdavi et al., 18].

Assuming that the n-dimensional point is $Q(x_1, x_2, \ldots, x_n)$, the opposite solution to this point is $Q'(x_1', x_2', \ldots, x_n')$. The calculation method of Q' is shown in formula (17):

$$x_j' = a_j + b_j - x_j \tag{17}$$

where: x_j' is the j-th dimension variable of point Q', a_j is the maximum value of j-th dimension in the current population, b_j is the minimum value of j-th dimension in the current population, x_j is the j-th dimension variable of point Q.

Figure 3 provides a scheme of the opposition-based learning.

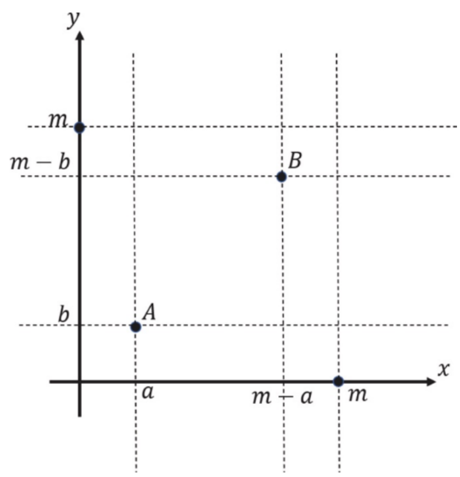


Figure 3: Scheme of the Opposition-based Learning

It is expected, as shown in Figure 3, that when using PIO to process a two-dimensional function, the maximum value in each dimension is m, the lowest value is 0, and the beginning point is $A(a, b)$, where $0 \leq a \leq b \leq m$; the opposite solution of this point is the point $B(m - a, m - b)$. The pigeon population uses a map and compass to conduct a global search. The opposition-based learning will aid in the development of the pigeons' capacity to seek across a greater area at this stage. The population's opposing solution is determined once the pigeon's location has been updated. The fitness of the original and opposite populations is then compared, and the pigeons with the highest fitness are kept in the new population.

2.4.4 Yin-Yang PIO process

The block diagram showing the Yin-Yang PIO process is depicted in Figure 4.

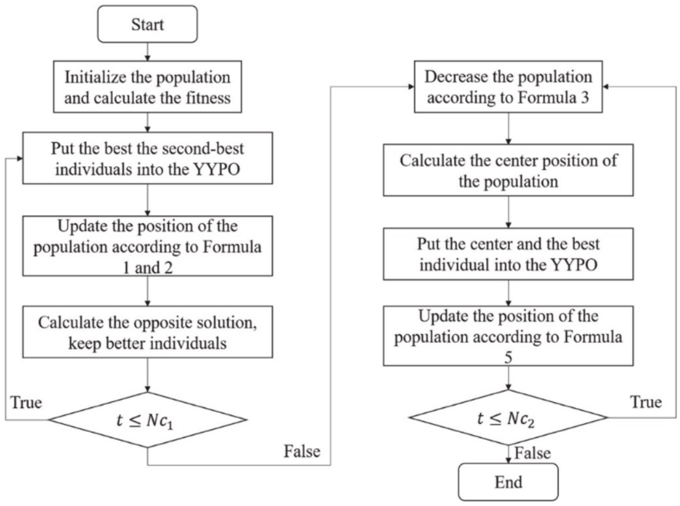


Figure 4: The Flowchart of Yin-Yang PIO

As seen in Figure 4, the Yin-Yang PIO process has several stages.

1: Calculating population fitness after providing a good point set to the pigeon population.

2: Placing the best and second-best individuals in YYPO as the initial values of P_1 and P_2.

3: Updating the position of the population according to formulas (1) and (2).

4: Retaining the individuals with the best fitness by using the opposition-based learning strategy.

5: Going to step 2 if t is less than Nc_1; otherwise, going to step 6.

6: Population reduction using a formula (3).

7: Use a method to determine the location of the population center (4). The initial values of P_1 and P_2 in YYPO are set to represent the population's center and best suited individuals.

8: Updating the population's location using a formula (5).

9: Going to step 6 if t is less than Nc_2; otherwise, ending the program.

3 Results and Discussion

3.1 Algorithm Performance Analysis

This article examines the M2M communication relationship, which is based on intelligent sensors and machines that are connected to a shared system in real-time and operate offline. Using AnyLogic software simulation, the main computer of the security operations centre is monitored. AnyLogic PLE is the simulation software that combines discrete event, system dynamics, and agent-based simulation methods and can model any real-world system or process. This makes it the perfect simulation software for researchers. It helps to introduce them to the principles of simulation and all modern modeling approaches, as well as teaching them how to choose the right abstraction level. Figure 3 shows the M2M communication control structure. It contains a data set derived from data collected by intelligent sensors on computers under the control of the Security Operations Centre (SoC).

Five sample standard functions were chosen to assess the performance of the Yin-Yang PIO algorithm: the sphere function (f_1), the Rosenbrock function (f_2), the Levy function (f_3), the Schwefel function (f_4), and the Levy function N.13 (f_5). These five test functions may be split into two categories: f_1 and f_2 are unimodal functions that can gauge how quickly the algorithm will converge, while f_3, f_4, and f_5 are multimodal functions with several local optimal values that can be derived from the local optimum.

The specific function information is shown in Table 1.

Function	x^*	$f(x^*)$	Interval	Measurement
f_1	$[0,0,0 \ldots,0]$	0	$[-100,100]$	30
f_2	$[1,1,1 \ldots,1]$	0	$[-2.048,2.048]$	30
f_3	$[1,1,1 \ldots,1]$	0	$[-10,10]$	30
f_4	$[420.96,\ldots,420.96]$	0	$[-500,500]$	30
f_5	$[1,1]$	0	$[-10,10]$	2

Note: x^* is the function's putative minimum point; $f(x^*)$ is the function's theoretically minimal value.

Table 1: Functions

Table 1 shows that while the theoretical minimum values of the functions are the same, the functions differ by the theoretical minimum point.

This study contrasts the PIO, YYPO, and PSO algorithms with the Yin-Yang PIO method. The number of populations in the Yin-Yang PIO is N = 50, the maximum number of iterations is IterMax = 500, the maximum number of iterations for the map and compass operator is Nc_1 = 350, and the maximum number of iterations for the landmark operator is Nc_2 = 150. PIO's parameter settings are identical to Yin-Yang PIO's. The maximum number of iterations for YYPO is 500, the maximum number of archive updates is 4, the maximum number of archive updates is 2, the expansion/contraction factor is 10, and the search radii are both set to 0.5.

The individual learning factor is set to c_1= 2, the social learning factor is set to c_2= 2, and the inertial weight is set to ω = 0.9 for PSO. The iterative maximum is IterMax=500. In MATLAB 2018b, each method was separately ran 20 times, and the outcomes are displayed in Table 2.

Function	Model	Maximum	Minimum	Mean value	Dispersion
f_1	PSO	3.03E+00	1.25E+00	2.10E+00	3.27E-01
	YYPO	7.15E-07	6.83E-11	4.33E-08	2.52E-14
	PIO	1.55E-116	1.14E-177	7.75E-118	1.20E-233
	YYPIO	0.00E+00	0.00E+00	0.00E+00	0.00E+00
f_2	PSO	1.68E+02	3.17E+01	5.09E+01	8.87E+02
	YYPO	7.98E+01	1.94E+01	3.45E+01	3.41E+02
	PIO	2.87E+01	9.21E-02	1.64E+01	1.61E+02
	YYPIO	3.68E-01	0.00E+00	6.57E-02	1.08E-02
f_3	PSO	8.67E+00	1.67E+00	5.01E+00	4.06E+00
	YYPO	3.40E-03	4.44E-12	4.11E-04	8.56E-07
	PIO	1.36E+00	4.67E-08	1.95E-01	1.69E-01
	YYPIO	8.30E-09	1.50E-32	4.22E-10	3.44E-18
f_4	PSO	8.34E+03	5.00E+03	6.33E+03	7.32E+05
	YYPO	6.03E+00	3.82E-04	3.03E-01	1.82E+00
	PIO	4.51E+03	1.60E-03	1.98E+03	3.15E+06
	YYPIO	3.82E-04	3.82E-04	3.82E-04	0.00E+00
f_5	PSO	8.83E-05	5.33E-08	1.44E-05	3.90E-10
	YYPO	1.63E-12	7.61E-16	2.34E-13	2.08E-25
	PIO	3.19E-02	2.32E-07	4.60E-03	7.75E-05
	YYPIO	3.32E-13	1.35E-31	3.32E-14	1.04E-26

Table 2: Results of the test functions

Table 2 shows that Yin-Yang PIO has the best minimum value, mean value, and variation when compared to other methods, demonstrating its superior optimization accuracy and robust stability. In experiments involving the functions f_3, f_4 and f_5, Yin-Yang PIO demonstrated an outstanding ability to globally search for the optimal solution. Regarding the function f_4, Yin-Yang PIO has higher stability even if it has the same lowest value as Yin-Yang PIO and did so in 20 independent trials. Its variance is also zero.

3.2 Modeling of Wireless Sensor Network Coverage Optimization

3.2.1 Comparison of Yin-Yang PIO with PSO and PIO

To determine the Yin-Yang PIO performance on the WSN coverage optimization problem, it was compared with PSO and PIO. When solving the WSN coverage optimization problem, formula (14) was used as an objective function when. The solution vector is made up of the nodes' horizontal and vertical coordinates. Therefore, if there are 20 nodes, the solution vector has a size of 40. The coverage efficiency was used to evaluate the optimization results. The average coverage speed of each sensor in a WSN, which might indicate the level of node redundancy, is known as coverage efficiency. The less redundant nodes there are, the greater the coverage efficiency. Formula (18) is comparable to the calculating process:

$$CE = \frac{\bigcup_{i=1,2,\ldots,N} s_i}{\sum_{i=1,2,\ldots,N} s_i} \tag{18}$$

where CE is the coverage efficiency, S_i is the size of the coverage area of the $i-th$ node, N is the number of nodes.

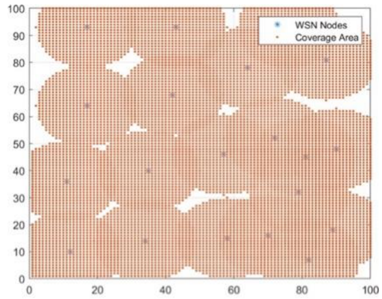
Twenty sensor nodes were dispersed over the 100×100 m monitoring area. Each sensor node's detecting radius was set at 15 meters, its transmission radius to 30 meters, and its maximum iteration number to 80. Table 3 and Figure 5 show the coverage ratio for each technique.

Analyzing the data given in Table 3, we can conclude that the studied algorithms differ significantly in terms of Comparison ratio and Coverage efficiency. In particular, the PSO algorithm has a Comparison ratio value of 93.10%, the PIO algorithm has a value of 89.29%, and the YYPIO algorithm has a value of 99.51 %. It can be noted that the YYPIO indicator is characterized by the value of a Coverage ratio of 6.9% relative to more than PSO and by 11.5% relative to more than PIO.

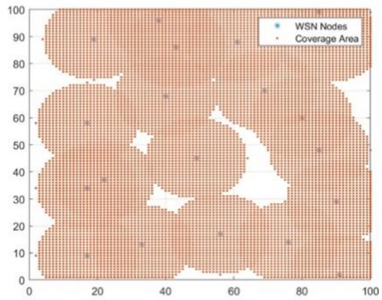
As for the Coverage efficiency indicator, the PSO algorithm has a value of 69.67%, the PIO algorithm has a value of 68.55%, and the YYPIO algorithm has a value of 91.67%. It can be noted that the indicator of YYPIO is characterized by the value of Coverage efficiency of 31.6% relative to more than PSO and by 33.7% relative to more than PIO.

Model	Coverage ratio, %	Coverage efficiency, %
PSO	93.10	69.67
PIO	89.29	68.55
YYPIO	99.51	91.67

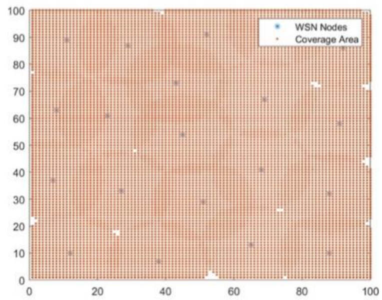
Table 3: Comparison of algorithms



(a) PSO



(b) PIO



(c) Yin-YangPIO

Figure 5: Node Coverage Distribution

Based on the data given in Figure 5, it can be stated that the use of the YYPIO algorithm makes it possible to place WSN nodes more efficiently, which makes it possible to significantly expand the Coverage area.

Figure 6 displays the convergence curve using the three approaches.

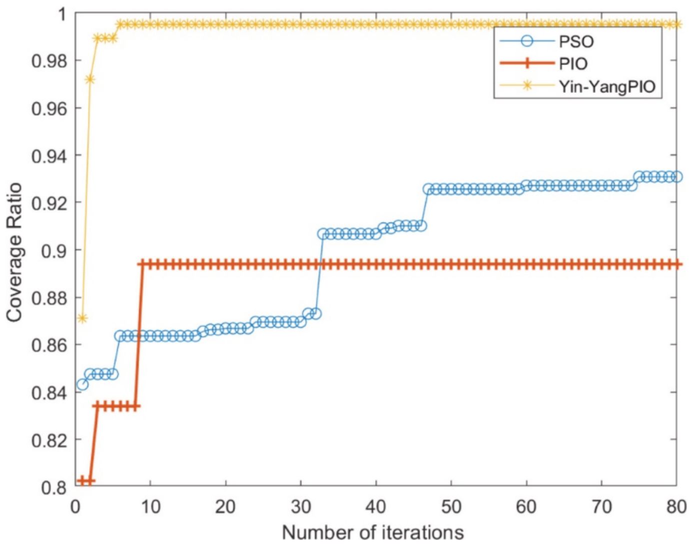


Figure 6: The Convergence Curve

Figure 6 shows the results of studies on the effect of the number of iterations on the Coverage Ratio indicator when using such algorithms as PSO, PIO, and YYPIO. It can be argued that an increase in the number of iterations unambiguously leads to an increase in the Coverage ratio. In particular, after 6 iterations, the value of the Coverage Ratio for the YYPIO algorithm increased from 87.12% to 99.51%. The value of the coverage ratio of the PIO algorithm after 8 iterations increased from 80.23% to 89.67%, and that of the PSO algorithm after 74 iterations from 84.23% to 93.52%. Based on this, it can be concluded that the Yin-Yang PIO method exhibits higher convergence while handling the WSN coverage optimization issue when compared to PIO and PSO.

Sensing radii were adjusted to 13, 14, and 15 m in order to investigate optimization impacts under various sensing radii. Other parameters remained same. The experimental results are presented in Table 4.

Method	Radius, m		
	13	14	15
PSO	80.27	86.30	93.11
PIO	77.82	82.27	89.30
YYPIO	91.74	96.96	99.50

Table 4: Comparison of algorithms

As seen from Table 4, Yin-YangPIO has the highest coverage rate regardless the sensing radius, and all three algorithms reach more than 90%.

The results of the experiments given in Table 4 show that as the radius increases from 13 to 15 m, the coverage ratio increases for all three studied algorithms. In particular, the Coverage ratio indicator increases for the PSO algorithm from 80.27% to 93.11%, for the PIO indicator - from 77.82% to 89.30%, and for the YYPIO indicator - from 91.74 to 99.50%.

As for the assessment of the Coverage ratio, for any radius values in the range from 13 to 15 m, the largest value of the Coverage ratio is characterized by the YYPIO algorithm (91.74% -99.50%). The worst values of the Coverage ratio are characterized by the PIO algorithm (77.82%-89.30%). The values of the PSO algorithm for all values of the radius take an intermediate value and are 80.27% -93.11%.

4 Discussions

In the second round of tests, 30 sensor nodes were dispersed across a monitoring area of 100×100 meters. Each sensor node's detecting radius was 13 meters, its transmission radius was 26 meters, and its maximum iteration was 80. The three techniques suggested in Wu et al. [16], Mu et al. [15], and Yang et al. [15] are compared to the effectiveness of the Yin-YangPIO algorithm. The coverage ratio using each of these methods is presented in Table 5 and Figure 7.

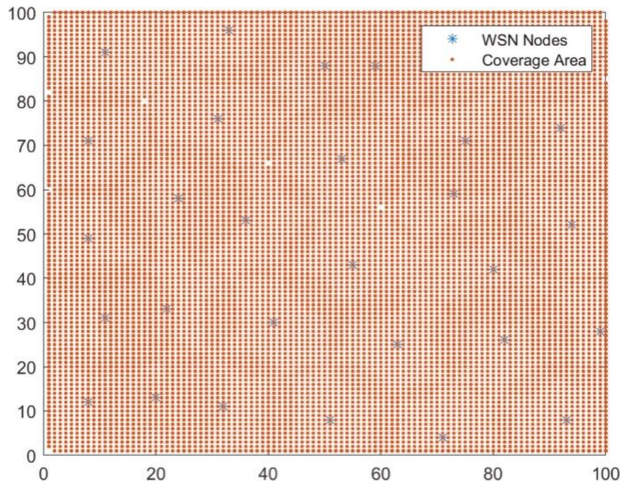


Figure 7: Node Coverage Distribution

The node coverage distribution diagram shows that using the Yin-Yang PIO algorithm leads to almost 100% coverage of the territory. It can be noted that the model proposed by Mu et al. covers 97.0% of the experimental area, the model proposed by Yang et al. covers 94.0% of the experimental area, and the model proposed by Wu et al. [16] covers 99.8% of the experimental area.

Model	Coverage ratio, %
[Mu et al., 15]	97.0
[Yang et al., 15]	94.0
[Wu et al., 16]	99.8
YYPIO	99.9

Table 5: Comparison of WSN parameters

As can be shown from Table 5 and Figure 7, the WSN coverage ratio of the YinYang PIO has improved to a maximum of 99.9% when compared to the approaches suggested in [Wu et al., 16, Mu et al., 15, Yang et al., 15]. The monitoring area for the third set of trials was 100 by 100 meters and had 45 sensor nodes. Each sensor node had a 10 m detecting radius, a 20 m transmission radius, and 80 iterations maximum. It was compared to the approach suggested in [Lu et al., 21] and Yin-YangPIO. Table 6 and Figure 8 show the coverage ratio for each technique. As can be observed from Table 6, the sensor node coverage is more uniform and the Yin-YangPIO optimization results shown here are better by more than 2% when compared to the enhanced artificial bee colony method presented in [Lu et al., 21].

Model	Coverage ratio, %
[Lu et al., 21]	96.07
YYPIO	98.46

Table 6: Comparison of WSN coverage ratio

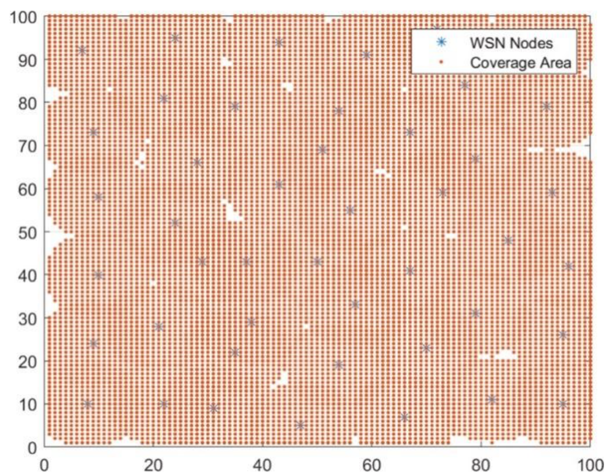


Figure 8: Node Coverage Distribution

The WSN coverage ratio obtained in this study (more than 98%) is significantly higher than that reported in Deepa and Venkataraman [21] using the whale optimization algorithm enhanced with levy flight (95.2%).

There is a next limitation: in order to avoid PIO from entering the local optimum, the search range and solution space should be expanded in the map and compass operator. Today, optimization algorithms are being widely used for the motion planning of complex optimization problems i.e. clusters, swarms and multi-objective by research scholars. Mostly, bio-inspired algorithms along with its variants have been proposed to increase the convergence speed and overall stability of the system. In order to maximize WSN coverage in more complicated contexts, future research will concentrate on improving algorithm efficiency. The method will be improved to address other IoT optimization issues.

5 Conclusions

This study proposes a WSN coverage optimization method based on Yin-Yang PIO. To make the pigeon population in the solution space more uniformly distributed, the initialization phase uses a decent point set. The ideal solution and central location were improved by combining YYPO with PIO because the optimal value of the population in PIO plays a significant effect. The population's optimum and suboptimal people are placed in YYPO as P_1 and P_2, respectively, in the map and compass operator, and the population center and optimal individuals are placed in YYPO as P_1 and P_2 , respectively, in the landmark operator, to increase the precision of optimization. The findings demonstrate that, in order to avoid PIO from entering the local optimum, the search range and solution space should be expanded in the map and compass operator. Opposition-based learning must be included with the map and compass operators in order to expand the population's variety.

The comparative experiments conducted in this study show that Yin-Yang PIO has a greater optimization ability and stronger stability compared with PIO, PSO and YYPO. This finding is of significant scientific and practical value.

Three sets of experiments were performed to optimize the WSN coverage with different parameters. The first series of experiments suggest that Yin–Yang PIO has the best optimization effect, with a coverage rate of 99.51% (10.22% higher with PIO and 6.41% higher compared with PSO). The second and third series of experiments show that Yin-Yang PIO significantly increased the WSN coverage ratio, up to 99.9%.

In order to maximize WSN coverage in more complicated contexts, future research will concentrate on improving algorithm efficiency. The method will be improved to address other IoT optimization issues.

Acknowledgements

1. Characteristic Innovation Projects of Colleges and Universities in Guangdong Province in 2021: Research and application of new miniature radar in intelligent parking lot （Serial number：2021KTSCX225）
2. 2023 Guangdong Provincial Higher Vocational Colleges and Universities Curriculum Civics Demonstration Program Project on Mobile Communication Technology (No. KCSZ04128)
3. Research and application of AI-based 5G green and low-carbon network construction under the "dual-carbon" goal, Shaoguan Science and Technology Bureau support for researchers, 2023 (No. 230327098036966)
4. 2021 University-level Excellence Online Open Course Mobile Communication Technology (No. 2021JPKC02)

References

[Akhmetbaev et al., 19] Akhmetbaev, D. S., Dzhandigulov, A. R. (2019): Development of algorithms for a new topological method for calculating current distribution coefficients in complex electrical networks. Eurasian Journal of Mathematical and Computer Applications, 7(3), 4–12. https://doi.org/10.32523/2306-6172-2019-7-3-4-12.

[Beketaeva et al., 19] Beketaeva, A. O., Bruel, P., Naimanova, A. Z. (2019): Detailed comparative analysis of interaction of a supersonic flow with a transverse gas jet at high pressure ratios. Technical Physics, 64, 1430-1440. https://doi.org/10.1134/S1063784219100049

[Bhardwaj and Kumar, 19] Bhardwaj, R, Kumar, D. (2019): MOFPL: Multi-objective fractional particle lion algorithm for the energy aware routing in the WSN. Pervasive and Mobile Computing, 58, 101029. https://doi.org/10.1016/j.pmcj.2019.05.010.

[Deepa and Venkataraman, 21] Deepa, R., Venkataraman, R. (2021): Enhancing Whale Optimization Algorithm with Levy Flight for coverage optimization in wireless sensor networks. Computers and Electrical Engineering, 94, 107359. https://doi.org/10.1016/j.compeleceng.2021.107359.

[Dou and Duan, 17] Dou, R., Duan, H. B. (2017): Lévy flight-based pigeon-inspired optimization for control parameters optimization in automatic carrier landing system. Aerospace Science and Technology, 61, 11–20. https://doi.org/10.1016/j.ast.2016.11.012.

[Duan and Qiao, 14] Duan, H. B., Qiao, P. X. (2014): Pigeon-inspired optimization: A new swarm intelligence optimizer for air robot path planning. International Journal of Intelligent Computing and Cybernetics, 7(1), 24–37. https://doi.org/10.1108/IJICC-02-2014-0005.

[Duan et al., 15] Duan, H. B., Qiu, H. X., Fan, Y. M. (2015): Unmanned aerial vehicle close formation cooperative control based on predatory escaping pigeon-inspired optimization. Scientia Sinica Technologica, 45(6), 559–572. https://doi.org/10.1360/N092015-00125

[Heidari et al., 17] Heidari, A., Kazemizade, O., Hakimpour, F. (2017): A new hybrid Yin-Yang-pair-particle swarm optimization algorithm for uncapacitated warehouse location problems. International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, 42, 373–379. https://doi.org/10.5194/isprs-archives-XLII-4-W4-373-2017.

[Idrees and Couturier, 22] Idrees, A. K., Couturier, R. (2022): Energy-saving distributed monitoring-based firefly algorithm in wireless sensors networks. The Journal of Supercomputing, 78(2), 2072–2097. https://doi.org/10.1007/s11227-021-03944-9.

[Khan et al., 15] Khan, I., Belqasmi, F., Glitho, R., Crespi, N., Morrow, M., Polakos, P. (2015): Wireless sensor network virtualization: A survey. IEEE Communications Surveys Tutorials, 18(1), 553–576. https://doi.org/10.1109/COMST.2015.2412971.

[Kintonova et al., 22] Kintonova, A., Povkhan, I., Mussaif, M., Gabdreshov, G. (2022): Improvement of Iris Recognition Technology for Biometric Identification of a Person. Eastern-European Journal of Enterprise Technologies, 6(2(120), 60-69. https://doi.org/10.15587/1729-4061.2022.269948

[Kong and Yu, 19] Kong, H., Yu, B. (2019): An improved method of WSN coverage based on enhanced PSO algorithm. In: 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC). Chongqing, China: IEEE, 1294–1297. https://doi.org/10.1109/ITAIC.2019.8785849.

[Liu et al., 18] Liu, W., Shi, H., Pan, S., Huang, Y., Wang, Y. (2018): An improved multi-threshold image segmentation algorithm based on pigeon-inspired optimization. In: 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics. Beijing, China: IEEE, 1–5. https://doi.org/10.1109/CISP-BMEI.2018.8633236.

[Liu et al., 21] Liu, L. J., Liu, D. Y., Liu, T. T. (2021): Application of improved Sine-cosine optimization algorithm in WSN coverage. Mathematics in Practice and Theory, 11, 129–137.

[Lu et al., 18] Lu, X. L., Cheng, W., He, Q., Yang, J. H., Xie, X. L. (2018): Coverage optimization based on improved firefly algorithm for mobile wireless sensor networks. In: 2018 IEEE 4th Int. Conf. Computer and Communications. Chengdu, China: IEEE, 899–903. https://doi.org/10.1109/CompComm.2018.8780713.

[Lu et al., 21] Lu, C., Li, X., Yu, W., Zeng, Z., Yan, M., Li, X. (2021): Sensor network sensing coverage optimization with improved artificial bee colony algorithm using teaching strategy. Computing, 103, 1439–1460. https://doi.org/10.1007/s00607-021-00906-0.

[Mahdavi et al., 18] Mahdavi, S., Rahnamayan, S., Deb, K. (2018): Opposition based learning: A literature review. Swarm and Evolutionary Computation, 39, 1–23. https://doi.org/10.1016/j.swevo.2017.09.010.

[Mu et al., 15] Mu, Y. Y., Qiao, X. G., Zhang, M. (2015): The application of bee colony optimization algorithms based on Voronoi in the coverage of wireless sensor networks. Chinese Journal of Sensors and Actuators, 28, 1525–1530. https://doi.org/10.3969/j.issn.1004-1699.2015.10.019

[Punnathanam and Kotecha, 16] Punnathanam, V., Kotecha, P. (2016): Yin-Yang-pair optimization: A novel lightweight optimization algorithm. Engineering Applications of Artificial. Intelligence, 54, 62–79. https://doi.org/10.1016/j.engappai.2016.04.004.

[Sagynganova et al., 22] Sagynganova, I., Kalinin, A., Smagulova, K., Lissitsyn, D., Abulkhairov, D. (2022): SMART system for the implementation of rational heat-supply regimes. Polityka Energetyczna, 25(2)m 137-146. https://doi.org/10.33223/epj/149889

[Serror et al., 20] Serror, M., Hack, S., Henze, M., Schuba, M., Wehrle, K. (2020): Challenges and opportunities in securing the industrial internet of things. IEEE Transactions on Industrial Electronics, 17(5), 2985–2996. https://doi.org/10.1109/TII.2020.3023507.

[Shi et al., 12] Shi, H.-Y., Wang, W.-L., Kwok, N.-M., Chen, S.-Y. (2012): Game theory for wireless sensor networks: A survey. Sensors, 12(7), 9055–9097. https://doi.org/10.3390/s120709055.

[Sun and Duan, 14] Sun, H., Duan, H. B. (2014): PID controller design based on prey-predator pigeon-inspired optimization algorithm. In: 2014 IEEE International Conference on Mechatronics and Automation. Tianjin, China: IEEE, 1416–1421. https://doi.org/10.1109/ICMA.2014.6885907.

[Sung and Yang, 14] Sung, T.-W., Yang, C.-S. (2014): Voronoi-based coverage improvement approach for wireless directional sensor networks. Journal of Network and Computer Applications, 39, 202–213. https://doi.org/10.1016/j.jnca.2013.07.003.

[Tao and Li, 18] Tao, G., Li, Z. (2018): A crossed pigeon-inspired optimization algorithm with congnitive factor. Journal of Sichuan University: Natural Science Edition, 55(2), 295–300.

[Toloueiashtian et al., 22] Toloueiashtian, M., Golsorkhtabaramiri, M., Rad, S. Y. B. (2022): An improved whale optimization algorithm solving the point coverage problem in wireless sensor networks. Telecommunication System, 79, 417–436. https://doi.org/10.1007/s11235-021-00866-y.

[Wang et al., 19] Wang, Y., Zhang, G., Zhang, X. (2019): Multilevel image thresholding using tsallis entropy and cooperative pigeon-inspired optimization bionic algorithm. Journal of Bionic Engineering, 16(5), 954–964. https://doi.org/10.1007/s42235-019-0109-1.

[Wang et al., 21] Wang, J., Li, Y., Hu, G. (2021): Hybrid seagull optimization algorithm and its engineering application integrating Yin–Yang pair idea. Engineering with Computers, 38, 2821–2857. https://doi.org/10.1007/s00366-021-01508-2.

[Wu et al., 16] Wu, Y., He, Q., Xu, T. (2016): Application of improved adaptive particle swarm optimization algorithm in WSN coverage optimization. Chinese Journal of Sensors and Actuators, 29(04), 559–565. https://doi.org/10.3969/j.issn.1004-1699.2016.04.016

[Xia et al., 22] Xia, X. G., Luo, J. T., Wang, X. (2022): Improved pigeon-inspired optimization algorithm of fuzzy variation operator. Journal of Henan University of Technology (Natural Science Edition), 43(3), 39–44. http://open.oriprobe.com/articles/62764192/mo_hu_bian_yi_suan_zi_de_gai_jin_ge_qun_you_hua_su.htm.

[Xu et al., 20] Xu, Q., Ma, L., Liu, Y. (2020): Yin-Yang-pair optimization algorithm based on chaos search and intricate operator. Journal of Computer Applications, 40(8), 2305–2312. https://doi.org/10.11772/j.issn.1001-9081.2020010089

[Yang et al., 15] Yang, C., Gao, W., Liu, N., Song, C. (2015): Low-discrepancy sequence initialized particle swarm optimization algorithm with high-order nonlinear time-varying inertia weight. Applied Soft Computing, 29, 386–394. https://doi.org/10.1016/j.asoc.2015.01.004.

[Zhang, 20] Zhang, Y. (2020): Coverage optimization and simulation of wireless sensor networks based on particle swarm optimization. International Journal of Wireless Information Networks, 27(2), 307–316. https://doi.org/10.1007/s10776-019-00446-7.

[Zhou et al., 21] Zhou, I., Makhdoom, I., Shariati, N., Raza, M. A., Keshavarz, R., Lipman, J., Abolhasan, M., Jamalipour, A. (2021): Internet of things 2.0: Concepts, applications, and future directions. IEEE Access, 9, 70961–71012. https://doi.org/10.1109/ACCESS.2021.3078549.