Journal of Electrical Engineering & Technology
https://doi.org/10.1007/s42835-021-00961-9

ORIGINAL ARTICLE

New Binary Particle Swarm Optimization Algorithm for Surveillance and Camera Situation Assessments

Chebi Hocine[1] · Abdelkader Benaissa[1]

Received: 26 May 2020 / Revised: 15 October 2021 / Accepted: 13 November 2021

Abstract

The problem of optimal camera placement (OCP) is the objective today, to identify the configurations of camera networks that optimize the area of interest under a set of constraints. In this article, we presented a novel technique based on new binary Particle Swarm Optimization (NBPSO) stimulated probability is anticipated to solve the camera placement problem. Ensuring correct visual coverage area of the monitoring hole with a minimum number of cameras is required. The illustration coverage is defined by practical and consistent assumptions attractive into description camera characteristics. After that, we proposed evolutionary of algorithms based on NBPSO and particle swarm optimization (PSO) are adapted to answer this optimal coverage based camera placement problem. These techniques are introduced in the context of processing constrained optimizations. We consider the case of presence of obstacles which could affect the placement of the cameras. Indeed, there could be a set of points not observed because of the imposed geometry of the obstacles, which occlude the fields of view entering their areas. The performance of NBPSO method stimulated is compared by the techniques many (e.g., genetic algorithms-based (GA), binary Particle Swarm Optimization BPSO, SA…). The results of the simulation were developed for the 2D scenarios which showed the fragrances of the proposed approach. Indeed, for a major element case, NBPS stimulated gives results good the ones obtained by adapting all variants.

Keywords Optimization · Camera networks · Optimal camera placement · Binary Particle Swarm Optimization (NBPSO) · Obstacles

1 Introduction

Since their creation of the camera, visual networks have known an ever growing success within scientific and industrial communities. Thanks to its various advantages, this technology has been able to establish itself as a key player in the placement of current cameras.

Meta-heuristic optimization algorithms are approaches that have been of great interest from several applications of researchers in different fields such as operational research, computer engineering, bioinformatics and automation. Their interest is due to their flexibility, their robustness, their simplicity and their flexibility to ensure good results as NBPSO can solve a variety of optimization problems in a reasonable calculation time.

Perfectly that research possesses and carried out in the field of true methods, the consistency NP of the underlying difficulty produces their use largely in practice. Also it is made possible and accessible to control or monitor large areas (areas) using cameras, the prayers that we are coming now for this problem have gone beyond the adequacy of basic research area displacement processes. Towards this logic, the property of metaheuristics has largely devoted tensions to an optimal placement of the camera.

Indeed, the quality of the monitoring in such application depends on the position and poses of the cameras. The main driving force of this work is to improve the off-line camera placement for surveillance applications. Camera placement depends on feasible location of cameras, obstacles present in sensitive areas, and the assigned priority of the area. Hence the placement problem becomes an optimization problem with inter related and competing constraints. Since constrained discrete optimization problems do not have efficient

✉ Chebi Hocine
chebi.hocine@yahoo.fr; chebi.hocine@univ-boumerdes.dz; hocine.chebi@univ-sba.dz

[1] Faculty of Electrical Engineering, Djillali Liabes University of Sidi Bel Abbes, Sidi Bel Abbès, Algeria

Published online: 25 November 2021

algorithmic solution (time complexity, memory saturation), evolutionary algorithms are used.

In this article, we have examined the deployment and adjustment of evolutionary techniques such as new binary particle swarm optimization (NBPSO), [1–8] be widely used and adapted to BILP difficulties at a very wide scale. And approximate to find a global solution with the fewest errors. On the other hand, the methods of particle swarm optimization (PSO) require some improvements to provide theoretical solutions of the global optima. In order to increase the probability rate to reach the good and the improved global solution of the optimum, we present in this article, an active approach based on the stimulated probabilities of BPSO (BPSO-IP) to maximize the spatial optical coverage d 'a 2D in a surveillance environment.

We present in this paper the NBPSO approach to optimize the spatial illustration coverage of a 2-D in milieu. This paper is organized as follows: Sect. 2 reviews the associated of OCP on this topic. In Sect. 3, modeling the placement problem with the presence of obstacle and optimization problem is proposed. A review for the NBPSO on Evolutionary method is set in Sect. 4. Assesses and discusses the consequences obtained for diverse scenarios In Sect. 5.

1.1 Background on Camera Placement

Optimal camera placement is a very challenging problem that has been proven to be NP-hard for most of the formulations of sensor deployment [7]. Indeed, placement problems are often represented as an integer linear programming (ILP) model [9, 10]. However, when using deterministic ILP methods in large scale problems most computers used failed to obtain a solution and got stack due to the solution search requirements and memory issues [10–15].

In the case of genetic algorithm (GA) [16], extended by Holland in 1992, is examined and will concede as the preferable evolutionary algorithm. Optimizer based on biogeography (BBO) [17, 18], There are certain other popular evolutionary algorithms: evolutionary programming (EP) [19], evolution strategy (ES) [20], differential evolution (DE) [21], incremental probability-based learning (PBIL)) [22], and genetic programming (GP) [23].

Perfectly that research possesses and carried out in the field of true methods, the consistency NP of the underlying difficulty produces their use largely in practice. Also it is made possible and accessible to control or monitor large areas (areas) using cameras, the prayers that we are coming now for this problem have gone beyond the adequacy of basic research area displacement processes. Towards this logic, the property of metaheuristics has largely devoted tensions to an optimal placement of the camera.

In contrast to traditional sensor networks which assume mono-directional sensors, camera networks are considered as directional sensors which introduce additional complexity to the camera placement problem [24, 25]. Similar to the mono-directional case, this problem is often described as an ILP [24]. To tackle such complexity, several heuristics have been proposed to find suboptimal solutions and good approximations [26–28].

However, the exact solution is proven to be NP-Hard [29]. Similarly, Floodlight Illumination Problems deal with the illumination of planar regions by light sources [30].

In Computational Geometry, very good progress has been made in solving optimal guard location problems for a polyp/Jai area, e.g., Art Gallery Problem (AGP) and its variants. In this problem, the main task is to determine a minimal number of guards and their static positions, such that all points in a polygon are observed [31–34]. However, the exact solution is proven to be NP-Hard [36]. Similarly, Floodlight Illumination Problems deal with the illumination of planar regions by light sources [30]. A variant of the AGP is known as Watchmen Tours where guards are allowed to move inside the polygon [37–39]. The objective is to find an optimal number and route for guards guaranteeing the detection of some intruder with an unknown initial position and unlimited speed. Authors in [Suzuki 2001] introduce another variant of watchman problem termed boundary search where guards are allowed to move only along the boundary of the polygon.

Nevertheless, all above works in AGP employ unrealistic assumptions about camera's capabilities such as unlimited field of view, infinite depth of field, and/or infinite servo precision and speed. In order to avoid such assumptions, some works take into account camera placement limiters, by proposing some algorithms and methods suitable for solving the camera placement problem [40–42]. The drawbacks with these algorithms are that they are only able to handle orthogonal polygons, have a bad time complexity, produce no mathematically optimal solution and assume that cameras have similar capabilities. This assumption makes the above algorithms unsuitable for most real-world computer vision applications. In fact, one objective of our work is to bridge the gap between the highly theoretical, well-established computational geometry and more realistic requirements of computer vision since the aim here is to ensure maximum coverage of a given area to be monitored formed by different zones of priority (coverage threshold) with optimum parameters such as locations and number of cameras. We have also added some realistic requirements, such as:

- The case of heterogeneous cameras with different fields-of-views at different levels of cost;
- The limited effective range of cameras;
- The possibility area containing obstacles which limit the line of sight of the different cameras to be placed.

According the literature review presented above, the placement problem is assimilated to an optimization problem with interrelated and competing constraints. However, it is known that constrained discrete optimization problems do not have efficient algorithmic solution in large scale. These problems are usually solved using evolutionary algorithms. Among the different evolutionary techniques, Genetic Algorithms (GA) and standard Particle Swarm Optimization (PSO) are found to be the most used in the context of sensors and cameras placement: [43] presented a brief survey explaining how PSO is tailored to address wireless sensor networks issues such as optimal deployment, node localization, clustering and data-aggregation while [44] used standard PSO to solve coverage problems with randomly deployed sensors [45]. Presented a sequential PSO technique to address sonar sensor placement problem. Their algorithm consists in selecting a sensor and subsequently optimizing its placement with respect to the other sensors (assumed fixed) using the standard PSO algorithm [46] proposed a framework based on hybrid GA to deploy and to configure a set of given sensors [47]. Presented three dynamic PSO-based techniques including PSO algorithm and learning automata to solve the deployment problem. It is shown from these reviews that PSO is the most suited and appropriated algorithm to solve placement problems in large scale [48–53]. However, almost of the cited papers from the literature differ deeply from our problem formulation and its solving using a more realistic sensing model. In addition, PSO algorithm still requires improvement to theoretically provide global optima when dealing with binary or discrete variables. In order to increase the probability of reaching the right and best solution (global optima), we propose in this chapter an efficient BPSO inspired probability (BPSO-IP) based approach to optimize the spatial visual coverage of a 2-D and 3-D security monitoring environment.

Modeling a Camera's FoV in Space: Among these factors, the recovery rate and the distance to the subject seem to us to be the most sensitive factors to the placement of the cameras. To do this, we modeled a "reverse virtual camera" using a light source created with the attribute "point light" in an Autodesk Maya scene. There is a range of specific modifiers for this, including the GOBO function, which simulates the grip of a light frustum using a specific texture mask. We will also use the linear dimming attribute to approximate, as shown in Fig. 5, the effect produced by the distance from the camera to the subject. This very simple model will make possible the visual and qualitative appreciation of the desired effect as long as the relative quantities mobilized in the model remain homogeneous with each other.

For our work, we used an experimental architectural model chosen for its restrictive topological characteristics (developed, masking, self-occlusion) and then put to the test by exploiting four camera placement protocols among the most used during survey campaigns.: they are mostly recommended by the community of researchers interested in this field of study and put into practice in the field by end users, experts or non-experts:

- Terrestrial circular campaign,
- Horizontal oblique aerial circular campaign,
- Oblique aerial circular campaign plunging at 45°,
- Nadir15 shooting while respecting the most commonly applied intra and inter-band recovery factors (80 and 60%).
- For this "reverse virtual camera" which will therefore project a light beam in the format of the characteristics of the on-board camera by the drone used for the test sessions—the MAVIC PRO brand DJI—we have retained the following parameters Table 1:

  The Field Of View calculation is done by the following formula:
- A horizontal FOV = 2. arctan (xc/(2. Focal length in mm))
- Vertical FOV = 2. arctan (yc/(2. Focal length in mm))
- Diagonal FOV = 2. arctan (d/(2. Focal length in mm))

Figure 1 summarizes all of the factors involved in the calculation of the camera parameters. This model was put to the test by a real measurement of the field of the on-board camera.

This allows us to deduce the theoretical detection angle (angle of coverage) taking into account the parameters announced by the manufacturer, which will be equal to approximately 84.03°. We can thus calculate the area covered by standing 10 m away from the subject in Nadir shooting: Horizontal overlap = tan (FOV hz/2) 0.2 and Vertical overlap = tan (FOV vt/2) 0.2.

The angle of view is a measure of the part of the world that is found in the image with photography and comparable techniques. This means that half of the world is visible at an angle of 180°. In practical applications, the image is usually a rectangular plane, so the angle of view should also indicate how it was measured. The viewing angle is generally expressed in degrees, measured on a diagonal of the image: $\alpha = 2\arctan(\frac{d}{2f})$ where d is the length of the image diagonal, the image sensor or the film, and f is the focal length of the lens. The diagonal of the image is the distance from one corner of the image to the opposite corner. This can be calculated with the Pythagorean Theorem: $d = \sqrt{\left(h^2 + v^2\right)}$ with h is the width and v is the height of the image.

2 Problem Definition

In this section consider the case of the presence of obstacles which could affect the placement of cameras. Indeed, there could be a set of points not observed because of the imposed

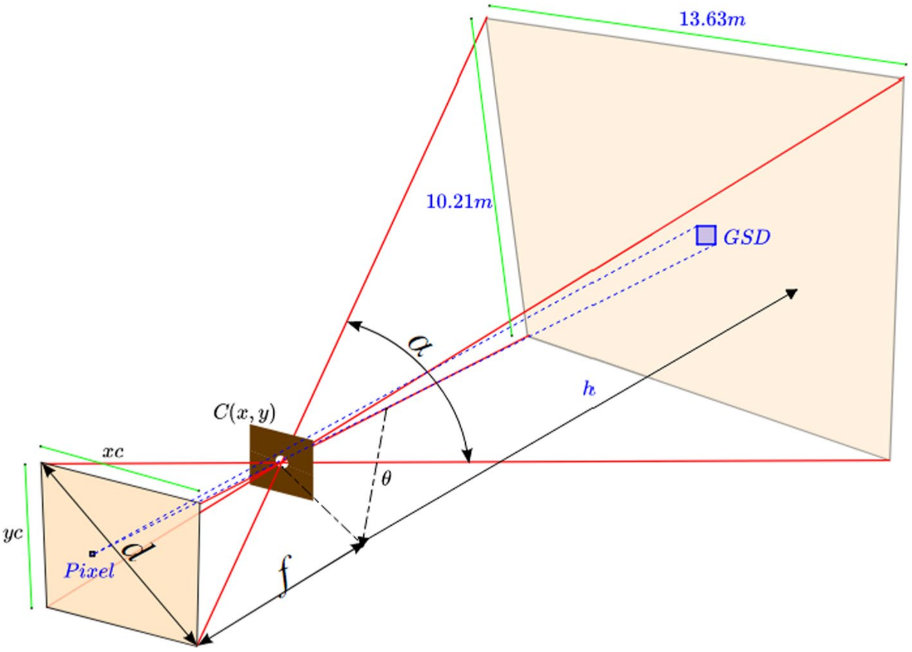


Fig. 1 Optical characteristics of the MAVIC Pro on-board camera with d = 8.052849 mm, f = 4.73 mm, h = 10.00 m, and the diagonal α = 80.82°

geometry of the obstacles, which occlude the fields of view entering their areas. Figure 2 presents a simplified example of this type of placement; where we regarded the perimeter of the obstacles as being the allocated positions of the cameras. The consideration of this hypothesis allows us to represent enclosures or building present in a monitoring area.

Each obstacle is represented by a polygonal form. Where, each perimeter edge of th6 obstacle is subdivided in a set of multiple edges with equal lengths depending to a preset threshold. Therefore, we recover the. New points representing the vertexes of each sub each perimeter edge. These points will become the points of a new grid representing the allocated positions of the cameras. It results from them finally two grids. One for the points to be covered and one for the allocated placement of the cameras.

The placement model remains almost the same as stated above. The only change will be in the manner of defining the two following binary variables $S_{i1j1\varphi nc}$ and $V_{cp}^{nc}(i_1, i_1, \varphi_1, i_2, j_2)$. These later will be expressed as follows:

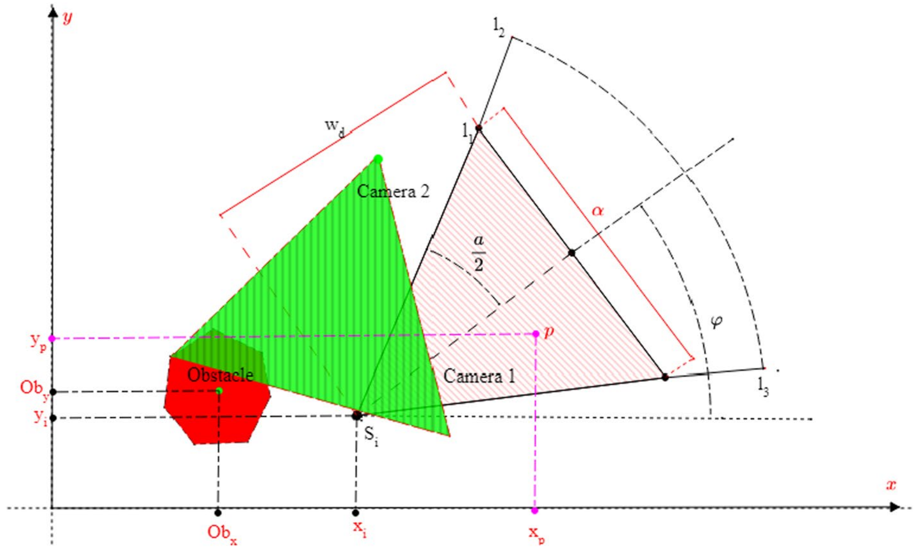


Fig. 2 Field-of-view Γ_i of camera S_i in 2D space with the presence of obstacle in placement problem

$$S_{i1j1\varphi nc} = \begin{cases} 1 \; \textit{If a camera of type nc with orientation } \varphi \textit{ is placed at point} \left(i_1, j_1\right) \textit{o grid} 1 \\ 0 \; \text{otherwise} \end{cases} \tag{1}$$

$$V_{cp}^{nc}\left(i_1, i_1, \varphi_1, i_2, j_2\right) = \begin{cases} 1 \; \textit{If a camera of type nc with orientation}\varphi\textit{is placed at point}\left(i_1, j_1\right) \textit{ of grid}1 \textit{ and cover the point}\left(i_1, j_1\right)\textit{of grid}2 \\ 0 \; \textit{otherwise} \end{cases} \tag{2}$$

However, solving this type of camera placement model can be infeasible. This case can happen when different cameras at different allocated positions cannot cover some points due to their constrained positions in another grid different to the point's grid. To overcome to this problem, we relaxed the problem stated above where instead of minimizing the number of cameras while insuring total points coverage; the objective will be to minimize the number of cameras while maximizing the number of coverage points. This optimization problem is formulated as follows:

Let a new variables $w1$ defined as:

$$w1_{i2j2} = \begin{cases} 1 \; \textit{if the point}(i2, \; j2)\textit{iscovered} \\ \qquad 0 \end{cases} \tag{3}$$

The objective function becomes as follows:

$$G1 = \sum_{nc=1}^{NC} k_{nc}\left(\sum_{\varphi=1}^{f_\varphi} \sum_{i_1=1}^{f_{x1}} \sum_{j_1=1}^{f_{y1}} S_{i1j1\varphi nc}\right) - \sum_{i_1=1}^{f_{x2}} \sum_{j_2=1}^{f_{y2}} w1_{i2j2} \tag{4}$$

Subject to the following constraints:

$$\begin{cases} w1_{i2j2} + \sum_{nc=1}^{NC} k_{nc}\left(\sum_{\varphi=1}^{f_\varphi} \sum_{i_1=1}^{f_{x1}} \sum_{j_1=1}^{f_{y1}} S_{i1j1\varphi nc}.V_{cp}^{nc}\left(i_1, j_1, \varphi_1, i_2, j_2\right)\right) \geq 0 \\ \sum_{nc=1}^{NC} \sum_{\varphi=1}^{f_\varphi} S_{i1j1\varphi nc} \leq L, 0 \leq i_2 \leq f_{x2}; 0 \leq j_2 \leq f_{y2} \end{cases} \tag{5}$$

The area covered by or visible from a camera relates to research known as visibility analysis. The observable points or areas of a camera are restricted; by geographical features, like buildings and trees, as well as the characteristics of a camera. Such characteristics include working distance, horizontal and vertical angle limits: There, fore, the visibility between each camera to be placed and the point to be viewed is analyzed. The visibility is decided by the line segment connecting each camera and grid points coordinate within its field of view.

As shown in Fig. 3, the line segment AB crosses no block. Therefore, point B is visible from camera A and vice versa. The line segment AC crosses a block. Therefore, point C is invisible from camera A and vice versa.

(a) Placement problem with possibility of selecting essential zones

A zone of interest or (essential zone), is a critical and very important zone which deserves more attention at the time of monitoring operation; such as the entry of a company, the armament in a military barracks,.. etc.). However, in our work; a zone of interest is equivalent to a selected set of points since the coverage area is approximated by a grid of points. The fact of selecting essential zones can be beneficial in large scale coverage, where a large number of cameras are required to cover the entire area to be monitored. The cameras should observe all essential regions and as many other points as possible. Here, we introduce a new parameter $w2$, defined as follows:

$$w2 = \begin{cases} 1\textit{if the point}(i2, j2)\textit{is an essenatial point} \\ 0 \; \textit{otherwise} \end{cases} \tag{6}$$

The optimization problem is formulated as follows:

$$G2 = \sum_{nc=1}^{NC} k_{nc}\left(\sum_{\varphi=1}^{f_\varphi} \sum_{i_1=1}^{f_{x1}} \sum_{j_1=1}^{f_{y1}} S_{i1j1\varphi nc}\right) - \sum_{i_1=1}^{f_{x2}} \sum_{j_2=1}^{f_{y2}} (1 - w2_{i2j2})w1_{i2j2} \tag{7}$$

Subject to:

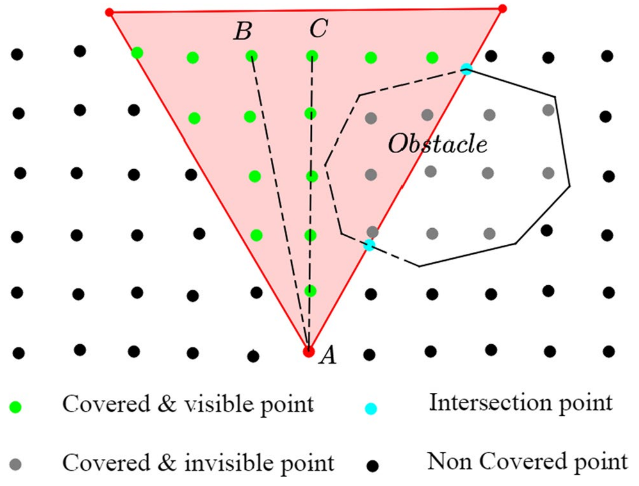


Fig. 3 Test illustrating the influence of the obstacle on the range of vision

$$\begin{cases} \sum_{nc=1}^{NC}\sum_{\varphi=1}^{f\varphi}\sum_{i_1=1}^{f_{x1}}\sum_{j_1=1}^{f_{y1}} S_{i1j1\varphi nc}\cdot V_{cp}^{nc}(i_1,i_1,\varphi_1,i_2,j_2)\geq M, if \quad w2_{i2j2}=1 \\ -w1_{i2j2}=\sum_{nc=1}^{NC}\sum_{\varphi=1}^{f\varphi}\sum_{i_1=1}^{f_{x1}}\sum_{j_1=1}^{f_{y1}} S_{i1j1\varphi nc}\cdot V_{cp}^{nc}(i_1,i_1,\varphi_1,i_2,j_2)\geq 0, if \quad w2_{i2j2}=1 \\ 0\leq i_2\leq f_{x2};\quad 0\leq j_2\leq f_{y2} \end{cases} \tag{8}$$

3 Novel Binary Particle Swarm Optimization (NBPSO).

As our problem of camera network placement is treated on discrete optimization space where variables can take a value of 0 or 1 only, we focus our study first on the case of the PSO. The binary PSO algorithm poses two problems in the case of placing cameras in an unstructured network and main concerns, in the first case the parameters of the binary PSO and in the second is the problem with the memory capacity of the binary PSO.

In this section, the camera position of the swarm is updated as in a continuous or binary variant. The primary difference between this algorithm and another version of the binary PSO algorithm is the speed accuracy. In this case, as in the continuous variant of PSO, the speed of a particle position is the speed at which the particle position changes its value in bits. We introduce two vectors for each particle position like $\overrightarrow{V}_i^0$ and $\overrightarrow{V}_i^1$. As $\overrightarrow{V}_i^0$ is the probability of the bits of the particle at zero, against the probability $\overrightarrow{V}_i^1$ is the bits of the particle evolve by one. Because in the equation for updating these speeds, which will be shown the inertia term is used, these speeds are not additional. Also, the probability of conversion in the j-th bit of the i-th particle is given as follows:

$$V_{ij}^{c}=\begin{cases} V_{ij,}^{1} & if\ x_{ij}=0 \\ V_{ij'}^{0} & if\ x_{ij}=1 \end{cases} \tag{9}$$

In this way, simply calculates the velocity of the particles to improve optimal placement. The algorithm proposed for $\overrightarrow{V}_i^0$ and $\overrightarrow{V}_i^1$ is given as follows:

Consider the optimal position visited for a particle is P_{ibest} and the optimal global camera position value for the particle is P_{gbest}.

Identically examine that the j-th bit of the i-th optimal particle value is equal to one. So, to develop the bit j-th of the i-th particle towards its best position and direction of the cameras, the speed goes to the change of one ($\overrightarrow{V}_i^1$) for that the enlarged particle and the speed of change to zero ($\overrightarrow{V}_i^0$) reduce. Using this concept, the following regulations have been extracted:

$$\begin{cases} if\ \ P_{ibest}^{j}=1 & then\ \ d_{ij,1}^{1}=c_1r_1 & and\ \ d_{ij,1}^{0}=-c_1r_1 \\ if\ \ P_{ibest}^{j}=0 & then\ \ d_{ij,1}^{0}=c_1r_1 & and\ \ d_{ij,1}^{1}=-c_1r_1 \\ if\ \ P_{gbest}^{j}=1 & then\ \ d_{ij,2}^{1}=c_2r_2 & and\ \ d_{ij,2}^{0}=-c_2r_2 \\ if\ \ P_{gbest}^{j}=0 & then\ \ d_{ij,2}^{0}=c_2r_2 & and\ \ d_{ij,2}^{1}=-c_2r_2 \end{cases} \tag{10}$$

where d_{ij}^1, d_{ij}^0, are the two temporary values in the area of interest. r_1 and r_2 of the order of (0,1) which are managed randomly optimized at any iteration. Again c_1, c_2 are two established variables which are determined by the user.

$$\begin{cases} V_{ij}^{1}=wV_{ij}^{1}+d_{ij,1}^{1}+d_{ij,2}^{1} \\ V_{ij}^{0}=wV_{ij}^{0}+d_{ij,1}^{0}+d_{ij,2}^{0} \end{cases} \tag{11}$$

where w is represent the term inertia. This work, the algorithm has improved global variables if the jth bit is zero, in this case the speed (V ij0) will be increased. And the likelihood also decreases with the same ratio.

After the speed of the particles V_{ij}^0 and V_{ij}^1 obtained changes according to Eq. (6). Subsequently we perform the transformation to normalized values with the use of the sigmoid function $V'_{ij}=sig\left(V_{ij}(t)\right)=\frac{1}{1+e^{-V_{ij}(t)}}$.

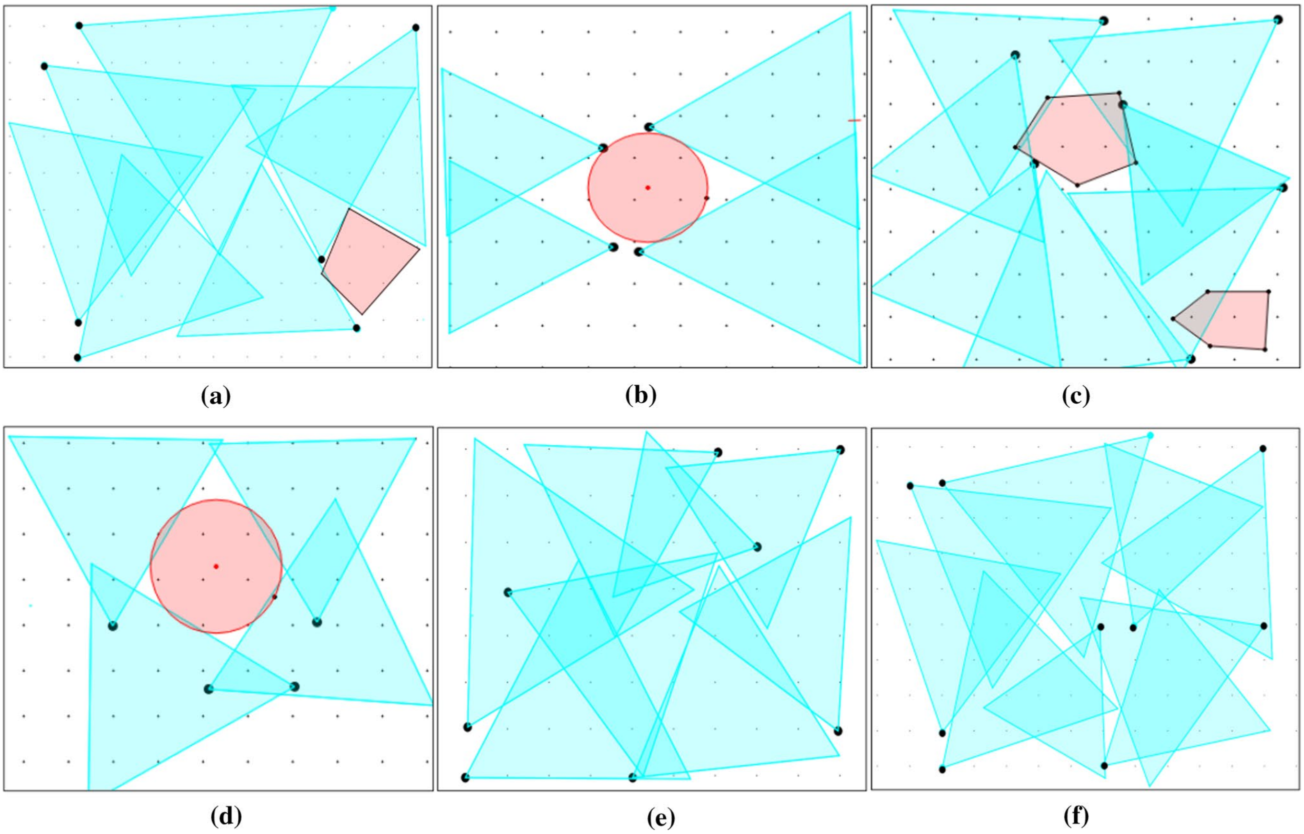
$$x_{ij}(t+1)=\begin{cases} \bar{x}_{ij}(t), & if\ r_{ij}<V'_{ij} \\ \bar{x}_{ij}(t), & if\ r_{ij}>V'_{ij} \end{cases} \tag{12}$$

With the variable x_{ij} is the complement of the variable $\bar{x}_{ij}$, that is to say $\bar{x}_{ij}$ Differently from x_{ij}. And r_{ij} is a uniform random value between 0 and 1.

The meaning of the elements used in the speed relation is precisely the same as for the continuous PSO. The density of inertia used now reserves the anterior direction of the particle bits towards the improved personal bit or the improved overall bit for either the 1 or the 0. Again, the previous directions of the particles bits are improved. Using Eq. (8), the prior particle value is taken as an extract, similarly as in BPSO, for a single velocity of the particle. Also, for improved gains for better learning of the experiences of the proposed algorithm are detailed:

ALGORITHM: Proposal from the NBPSO for optimizing camera placement

```
Input:   Initialize camera positions $X_i = \{x_1, x_2, \dots x_m\}$, the camera position is initialized randomly. The camera positions
         $X_i$ are selected by the binary values of 0 or 1.
         Initialize the orientation position of cameras $\theta_i = \{\theta_1, \theta_2 \dots \theta_n\}$.
Output:  Optimized position and orientation of cameras $(X_i, \theta_j)$
Algorithm
         1.  Improved the F function for each particle, using the present position $X_i$
// Compare the values obtained from each position and orientation to its improved performance:
         2.  If $F(X_i) < F(P_{ibest})$ do
         3.      $F(P_{ibest}) = F(X_i)$;
         4.      $P_{ibest} = X_i$;
// Compare for each camera characteristic (position and orientation) a better overall value:
         5.  If $F(X_i) < F(P_{gbest})$ do
         6.      $F(P_{gbest}) = F(X_i)$;
         7.      $P_{gbest} = X_i$;
         8.  Modify the speed of each particle, ie for $\vec{V}_i^0$ and $\vec{V}_i^1$ according to equation (8).
         9.  Calculate the speed of each change of bits, $\vec{V}_i^c$ according to equation (6).
         10. Generate $r_{ij}$ randomly between (0, 1). Replace each particle with a new position or orientation according to
             equation (9).
         11. Go to step 2 and repeat until the parameters have completely converged.
```



(a) (b) (c)

(d) (e) (f)

Fig. 4 In several tests of optimal location of surveillance cameras in an obstacle and uniform environment

4 Experimental Results

In this section, we present the results of NBPSO for the problem of placing cameras in a uniform network which semi has constraints (obstacle, unstructured,…). One of the difficulties in the generation of an NBPSO algorithm is to be successful in the representation, that is to say the assignment matching adapted between the result of the problem and the state of the NBPSO particles. For NBPSO represents a dimensional network composed of particles, for a whole value a binary column indicates the orientation and the position of the camera to extend.

The algorithm is simulated in this work to solve the problem of placing the camera network in a uniform system with a 2D grid of 2×2 points and 2 position and orientation values. Several studies, brought in research works, extend the PSO to direct the constrained optimization problems. In this work, an active framework to solve the constrained difficulty described in the previous one is introduced for all the methods based on the swarm of particles. The maintain feasibility line is used NBPSO to direct the time constraints and the memory to satisfy all the intoxication constraints.

Two changes have been introduced to direct the constraints of the NBPSO algorithm. In the first case, the

Table 1 Camera specifications for the comparison test

Consideration	Details
Focal length (in mm)	4.73 mm
Pixel size (mm)	0.00161057×0.00161057
Useful photosite size	4000×3000 pixels
Field Of View (manufacturer data)	78.8°
Vertical FOV	54.11 465°
Diagonal FOV	80.82802°
Horizontal FOV	68.56051°

The total number of photosites announced by the manufacturer is 12.71 Mpix, a portion of which is used when shooting, the "peripheral residual" of 0.71 Mpix acting as an adjustment variable used via the integrated steadyshot function

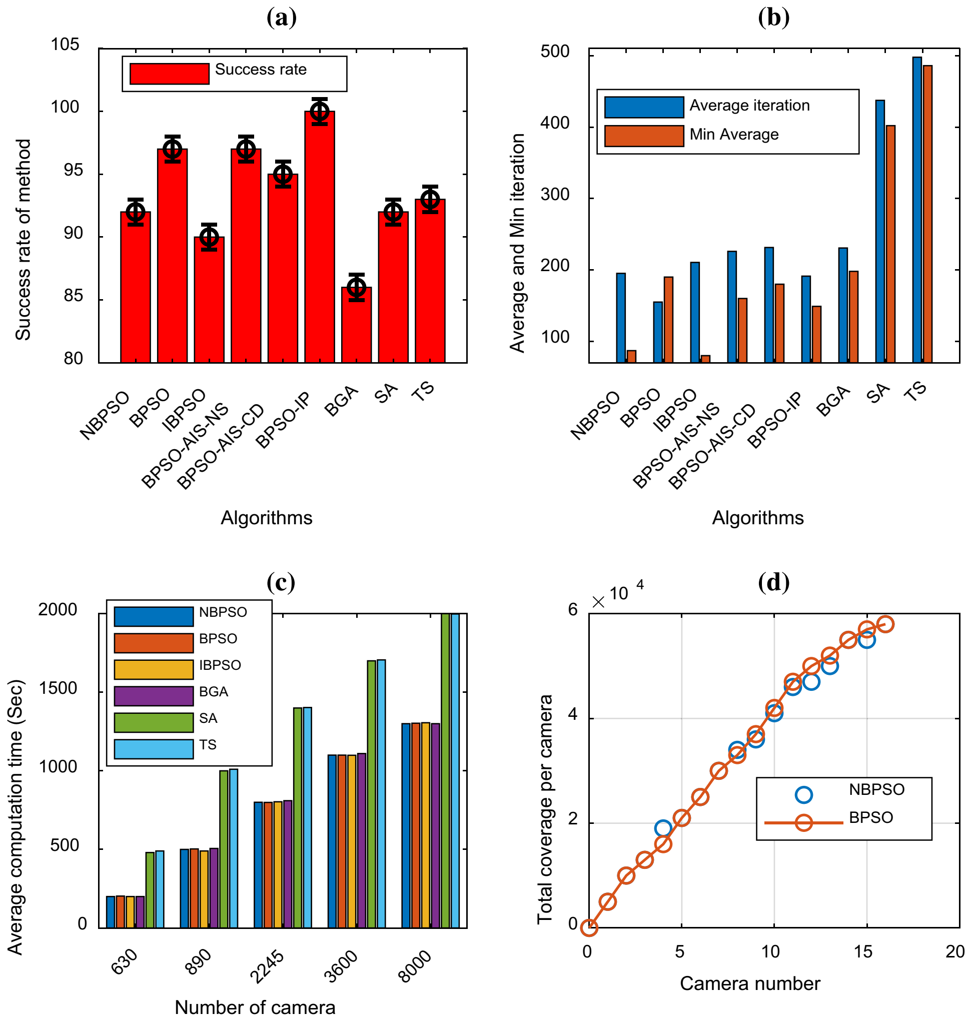


Fig. 5 Evaluation results of the proposed method for optimal location of surveillance camera network in a uniform and obstacle environment. **a** Success rate of this method, **b** Leverage and minimum iteration, **c** Inference of number of cameras on average computation time, **d** Camera number influence on total covered per camera

parameters of the binary PSO and in the second, the problem of the memory capacity of the binary PSO. The idea for these changes is to bring a problem to an unconstrained problem using NBPSO.

Experiment tests have been completed using some approaches based on the PSO algorithm presented in Figs. 3 and 4. (NBPSO, BPSO, Improved BPSO (I-BPSO), BPSO Using an Artificial Immune System—Based on Negative Selection (BPSO-AIS-NS), BPSO-AIS Based on Compatibility Degree (BPSO-AIS-CD), BPSO Inspired Probability (BPSO-IP)).

Before observing the results of the comparison, we show some results removed by the NBPSO algorithm in a case of random camera deployment. We estimated the NBPSO algorithm with several dimensions and types of cameras the results are given in by the Fig. 4e–f. Subsequently, we estimated the NBPSO algorithm using a medium with obstacle see Fig. 4a–d.

In all the Fig. 4a–f, the squares represent the borders of the area to be covered while the light green triangles represent the grid of the area detected by camera using our NBPSO approach. The grid nodes to cover are the points of interest. The large black dots represent the optimal position of the cameras to be deployed using the proposed NBPSO algorithm. Figure 4 illustrates examples where the majority of all the points of the grid observed are covered with an optimal placement of the cameras while making the cameras to be located only on the accepted black grid.

In all the results illustrating the cases where we rather have a uniform or complex coverage area, are presented in Fig. 4. In this figure, we can observe the effect of the presence of an obstacle in a visual field. All grid cameras are covered and each obstacle and taken into consideration. However, some small areas in some cases are not covered. This localization can be avoided by increasing the sampling frequencies of the grid. The results of the proposed NBPSO algorithm are illustrated in Fig. 4.

Figure 5 shows the different results obtained for the 2D surveillance benchmark used to plot all the algorithms used. Each technique was performed 80 times. For each algorithm realization, the optimal configuration and the number of iterations to reach the conclusion. The success rate, the median number of iterations, are after procedures for each method. We can distinguish in the Fig. 4 that our proposed algorithm precedes the visible between the algorithms and gives the corrected results to reach the global optimum.

The goal is to minimize the total number of cameras in a placement and therefore minimize the cost of cameras to install while ensuring that all points on the grid are hidden by the deployed cameras. In our work, the area coverage is occupied which states that all the points of the grid are covered. Also, the variation of the spatial sampling frequency improved the percentage of the coverage area (see Fig. 5a).

On the other hand, the Fig. 5b represents the concept of average and maximum number of iterations obtained for each algorithm. Note that the method used is convergent only from another method. The Fig. 5c presents the results of comparison in terms of computation time several algorithms for other values of optimization variables. The NBPSO algorithm vastly outperforms the rest of the methods. Indeed, the computation time interval between NBPSO and the rest of the algorithms increases with the number of cameras installed in the camera placement problem. In the Fig. 5d represents the comparison between PSO and NBPSO in measurement the rate of cover of the surveillance camera.

We compare our method with other techniques such as (NBPSO, BPSO, Improved BPSO (I-BPSO), BPSO Using an Artificial Immune System—Based on Negative Selection (BPSO-AIS-NS), BPSO-AIS Based on Compatibility Degree (BPSO-AIS-CD), BPSO Inspired Probability (BPSO-IP)) that it gave good and comparable result.

5 Conclusion

In this article, we have approached the request for automatic placement of the camera network using scalable features. The results of simulations have demonstrated the advantage of the proposed algorithm, NBPSO to solve the difficulties of coverage with the presence of obstacle and a uniform area, in particular for large-scale measurements to solve the problem of terms of memory and computing time. All the proposed techniques being uncertain, the major difficulty was to ensure a global optimum for a uniform structure. For this reasoning, this observation allowed and clarified that the result of the overall optimum for the optimization methods. The manipulations proposed in this article we conclude from original problems in the future to progress the robustness of NBPSO method. We will learn the possibility of combining the NBPSO algorithm with other algorithms to speed up the processing process responding to uniform media and to guarantee the global optimal. We also propose to apply this technique in the future in robotic navigation applications.

References

1. Kritter J, Brévilliers M, Lepagnot J, Idoumghar L (2019) On the optimal placement of cameras for surveillance and the underlying set cover problem. Appl Soft Computing 74:133–153

2. Eberhart R, Kennedy J (1995) Particle swarm optimization. In: Proceedings of the IEEE international conference on neural networks. Citeseer, pp 1942–1948.
3. Kennedy J, Eberhart RC (1997) A discrete binary version of the particle swarm algorithm. In (1997) IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation. IEEE , pp4104–4108
4. Shi Y, Eberhart RC (1999) Empirical study of particle swarm optimization. In: Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406). IEEE, New York, pp 1945–1950.
5. Chuang LY, Tsai JH, Yang CH (2010) Binary particle swarm optimization for operon prediction. Nucleic Acids Res 38(12):e128–e128
6. Taha M, Abualnadi D, Hasan O (2016) Model Order Reduction using fractional order systems. In: 2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE). IEEE, New York, pp 199–204.
7. Morsly Y, Aouf N, Djouadi MS, Richardson M (2011) Particle swarm optimization inspired probability algorithm for optimal camera network placement. IEEE Sens J 12(5):1402–1412
8. Gherboudj A, Chikhi S (2011) BPSO Algorithms for Knapsack Problem. In: Özcan A, Zizka J, Nagamalai D (eds) Recent trends in wireless and mobile networks. CoNeCo 2011, WiMo 2011. Communications in Computer and Information Science, vol 162. Springer, Berlin, Heidelberg.
9. Chakrabarty K, Iyengar SS, Qi H, Cho E (2002) Grid coverage for surveillance and target location in distributed sensor networks. IEEE Trans Comput 51(12):1448–1453
10. Hörster E, Lienhart R (2006) On the optimal placement of multiple visual sensors. In: Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks, 2006, pp 111–120.
11. Indu S, Chaudhury S, Mittal NR, Bhattacharyya A (2009) Optimal sensor placement for surveillance of large spaces. In: 2009 Third ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC). IEEE, New Year, pp 1–8.
12. Bahrami M, Amiri MJ, Mahmoudi MR, Koochaki S (2017) Modeling caffeine adsorption by multi-walled carbon nanotubes using multiple polynomial regression with interaction effects. J Water Health 15(4):526–535
13. Haghbin H, Mahmoudi MR, Shishebor Z (2015) Large sample inference on the ratio of two independent binomial proportions. J Math Extension.
14. Pan JJ, Mahmoudi MR, Baleanu D, Maleki M (2019) On comparing and classifying several independent linear and non-linear regression models with symmetric errors. Symmetry 11(6):820
15. Mahmoudi MR, Nematollahi AR, Soltani AR (2015) On the detection and estimation of the simple harmonizable processes. Iran J Sci Technol (Sciences) 39(2):239–242
16. Holland JH (1992) Genetic algorithms. Sci Am 267(1):66–73
17. Simon D (2008) Biogeography-based optimization. IEEE Trans Evol Comput 12(6):702–713
18. Ma H, Simon D, Siarry P, Yang Z, Fei M (2017) Biogeography-based optimization: a 10-year review. IEEE Trans Emerg Top Comput Intell 1(5):391–407
19. Yao X, Liu Y, Lin G (1999) Evolutionary programming made faster. IEEE Trans Evol Comput 3(2):82–102
20. Beyer HG, Schwefel HP (2002) Evolution strategies—a comprehensive introduction. Nat Comput 1(1):3–52
21. Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11(4):341–359
22. Baluja S (1994) Population-based incremental learning: a method for integrating genetic search based function optimization and competitive learning. Technical report CMU-CS-94- 163, Carnegie Mellon University, USA.
23. Koza JR (1994) Genetic programming as a means for programming computers by natural selection. Stat Comput 4(2):87 112.
24. Osais Y, St-Hilaire M, Yu FR (2009) On sensor placement for directional wireless sensor networks. In: 2009 IEEE International Conference on Communications. IEEE, New York, pp 1–5.
25. Soro S (2007) Application-aware resource management in wireless and visual sensor networks. Phd thesis, school of engineering and applied science, University of Rochester, p 43.
26. Fusco G, Gupta H (2009) Selection and orientation of directional sensors for coverage maximization. In: 2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks. IEEE, 2New York, pp 1–9.
27. Han X, Cao X, Lloyd EL, Shen CC (2008) Deploying directional sensor networks with guaranteed connectivity and coverage. In: 5th Annual IEEE communications society conference on sensor, mesh and ad hoc communications and networks. IEEE, New York, pp 153–160
28. Pan J, Cai L, Hou YT, Shi Y, Shen SX (2005) Optimal base-station locations in two-tiered wireless sensor networks. IEEE Trans Mob Comput 4(5):458–473
29. Fragoudakis C, Markou E, Zachos S (2007) Maximizing the guarded boundary of an Art Gallery is APX-complete. Computational Geometry 38(3):170–180.
30. Bose P, Guibas L, Lubiw A, Overmars M, Souvaine D, Urrutia J (1997) The floodlight problem. Int J Comput Geometry Appl 7(01n02), pp 153–163.
31. González-Banos H (2001) A randomized art-gallery algorithm for sensor placement. In: Proceedings of the seventeenth annual symposium on Computational geometry, pp 232–240.
32. Eidenbenz S (2006) Finding minimum hidden guard sets in polygons tight approximability results. Comput Geom 34(2):49–57
33. Hutchinson J, Kündgen A (2006) Orthogonal art galleries with interior walls. Discret Appl Math 154(11):1563–1569
34. Bottino A, Laurentini A, Rosano L (2007) A tight lower bound for art gallery sensor location algorithms. In: 2007 IEEE conference on Emerging Technologies and Factory Automation (EFTA 2007). IEEE, New York, pp 434–440.
35. Salleh I (2009) K-vertex guarding simple polygons. Comput Geom 42(4):352–361
36. Fragoudakis C, Markou E, Zachos S (2007) Maximizing the guarded boundary of an Art Gallery is APX-complete. Comput Geom 38(3):170–180
37. Carlsson S, Nilsson BJ, Ntafos S (1991) Optimum guard covers and m-watchmen routes for restricted polygons. In: Workshop on algorithms and data structures. Springer, Berlin, pp 367–378.
38. Efrat A, Guibas LJ, Har-Peled S, Lin DC, Mitchell JS, Murali TM (2000) Sweeping simple polygons with a chain of guards. In: Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms, pp 927–936.
39. Guibas LJ, Latombe JC, LaValle SM, Lin D, Motwani R (1999) A visibility-based pursuit-evasion problem. Int J Comput Geometry Appl 9(04n05):471–493.
40. Pålsson M, Ståhl J (2008) The camera placement problem: an art gallery problem variation. Lund University, Department of Computer Science
41. Tóth CD (2000) Art gallery problem with guards whose range of vision is 180. Comput Geom 17(3–4):121–134
42. Tóth CD (2002) Art galleries with guards of uniform range of vision. Comput Geom 21(3):185–192
43. Kulkarni RV (2011) Senior Member, IEEE, and Ganesh Kumar Venayagamoorthy, Senior Member, IEEE, Particle Swarm Optimization in Wireless Sensor Networks: A Brief Survey. IEEE, New York.

44. Kukunuru N, Thella BR, Davuluri RL (2010) Sensor deployment using particle swarm optimization. Int J Eng Sci Technol 2(10):5395–5401
45. Ngatchou PN, Fox WL, El-Sharkawi MA (2005) Distributed sensor placement with sequential particle swarm optimization. In: Proceedings 2005 IEEE Swarm Intelligence Symposium, 2005. SIS. IEEE, pp. 385–388.
46. Topcuoglu HR, Ermis M, Sifyan M (2009) Hybrid evolutionary algorithms for sensor placement on a 3d terrain. In: 2009 Ninth International Conference on Intelligent Systems Design and Applications. IEEE, pp 511–516.
47. Soleimanzadeh R, Farahani BJ, Fathy M (2010) PSO based deployment algorithms in hybrid sensor networks. Int J Comput Sci Netw Secur 10:167–171
48. Hassan R, Cohanim B, Weck O (2004) A comparison of particle swarm optimization and the genetic algorithm. American Institute of Aeronautics and Astronautics.
49. Uysal O, Bulkan S (2008) Comparison of genetic algorithm and particle swarm optimization for bicriteria permutation flowshop scheduling problem. Int J Comput Intell Res 4(2):159–176
50. Khanesar MA, Teshnehlab M, Shoorehdeli MA (2007) A novel binary particle swarm optimization. In: 2007 Mediterranean Conference on Control & Automation. IEEE, pp. 1–6.
51. Chandrasekaran, Gokul et al. (2021) Test scheduling of system-on-chip using dragonfly and ant lion optimization algorithms. J Intell Fuzzy Syst 40(3):4905–4917. https://doi.org/10.3233/JIFS-201691
52. Chandrasekaran G, Periyasamy S, Rajamanickam KP (2020) Minimization of test time in system on chip using artificial intelligence-based test scheduling techniques. Neural Comput Appl 32:5303–5312. https://doi.org/10.1007/s00521-019-04039-6
53. Chandrasekaran G, Periyasamy S, Karthikeyan PR (2019) Test scheduling for system on chip using modified firefly and modified ABC algorithms. SN Appl Sci 1:1079. https://doi.org/10.1007/s42452-019-1116-x

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Hocine Chebi born on April 26, in Tazmalt (W: Bejaïa), Algiers. He obtained his baccalaureate in Electrical Engineering in 2007 from the technical school of Mekla (W: Tizi-ouzou), Algiers. He then obtained his State Engineer degree in Control of Industrial Processesing in June 2013, from the faculty of hydrocarbons and chemistry (FHC) at the university of Boumerdès, Algiers. He then went on to obtain a Magister in Electrical engineering/automatic option controls and order in 2015 at the polytechnic Military academy (EMP) of Bordj El Bahri, Algiers. Currently doctor and university lecturer grade lecturer. His research area is oriented towards computer vision and the detection of automatic and control anomalies.

Abdelkader Benaissa is a professor currently doing research at the Faculty of Electrical Engineering of Djillali Liabes University of Sidi Bel Abbes. He is registered in the Laboratory of Intelligent Control and Electrical Power Systems. His research area is oriented towards the improvement of the Interconnection Management of Alternative Electrical Networks by High Voltage Direct Current Connection (HVDC).