Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Optimized deployment of a radar network based on an improved firefly algorithm*

Xue-jun ZHANG[1], Wei JIA[1], Xiang-min GUAN[2], Guo-qiang XU[1], Jun CHEN[3], Yan-bo ZHU[†‡1]

[1]*National Key Laboratory of CNS/ATM, School of Electronic and Information Engineering, Beihang University, Beijing 100191, China*

[2]*Department of General Aviation, Civil Aviation Management Institute of China, Beijing 100191, China*

[3]*Lincoln School of Engineering, University of Lincoln, Brayford Pool Campus, Lincoln LN67TS, UK*

[†]E-mail: yanbo_zhu@163.com

Received Nov. 24, 2018; Revision accepted Jan. 17, 2019; Crosschecked Mar. 14, 2019

Abstract: The threats and challenges of unmanned aerial vehicle (UAV) invasion defense due to rapid UAV development have attracted increased attention recently. One of the important UAV invasion defense methods is radar network detection. To form a tight and reliable radar surveillance network with limited resources, it is essential to investigate optimized radar network deployment. This optimization problem is difficult to solve due to its nonlinear features and strong coupling of multiple constraints. To address these issues, we propose an improved firefly algorithm that employs a neighborhood learning strategy with a feedback mechanism and chaotic local search by elite fireflies to obtain a trade-off between exploration and exploitation abilities. Moreover, a chaotic sequence is used to generate initial firefly positions to improve population diversity. Experiments have been conducted on 12 famous benchmark functions and in a classical radar deployment scenario. Results indicate that our approach achieves much better performance than the classical firefly algorithm (FA) and four recently proposed FA variants.

Key words: Improved firefly algorithm; Radar surveillance network; Deployment optimization; Unmanned aerial vehicle (UAV) invasion defense

https://doi.org/10.1631/FITEE.1800749 **CLC number:** TN954; O224

1 Introduction

In recent years, unmanned aerial vehicles (UAVs) have experienced rapid development and show an explosive growth in both civilian and military applications. However, the explosive growth of UAVs has caused severe flight security issues and widespread concern. For example, UAV invasion of airport clearance protection areas has occurred frequently at large airports in China, and has seriously affected the normal operation of airports and airplane flight safety. Therefore, UAV invasion defense is an important issue for airspace security that requires great attention.

Radar network detection plays a significant role in UAV invasion defense. The calculation of the detection range of a single radar instance has been intensely studied (Difranco and Kaiteris, 1981; Blake, 1986; Srinivasan, 1986; Baker and Hume, 2003; Zheng and Zheng, 2011). In Blake (1986), the radar range equation and the significance of each of the parameters were reviewed. The single process method plays a significant role in radar detection performance, as discussed in Difranco and Kaiteris (1981). For a radar network, the combination of different radar types is essential to form a seamless

‡ Corresponding author

* Project supported by the National Key Laboratory of CNS/ATM, Beijing Key Laboratory for Network-Based Cooperative Air Traffic Management, and the National Natural Science Foundation of China (No. 71731001)

ORCID: Yan-bo ZHU, http://orcid.org/0000-0003-1691-2680

surveillance coverage (Baker and Hume, 2003). Furthermore, to form a tight and reliable surveillance network with limited resources, it is essential to determine an optimal radar deployment strategy. The main task of deployment optimization is to achieve optimal network performance with a limited number of radar devices. Completeness and continuity are two important objectives of radar network deployment optimization. Through deployment optimization, the radar network must form a seamless coverage area that includes high, medium, and low altitudes, and must have an appropriate overlap to cover the main height layer. This is a multi-objective optimization problem with strong coupling of multiple constraints and nonlinear character, and is difficult to solve by conventional optimization methods, such as the simplex method and gradient descent method.

Building a mathematical model of the deployment problem and applying an efficient optimization algorithm have attracted great attention. Yang et al. (2013) proposed a hexagonal radar network deployment strategy and a diamond strategy. Yang et al. (2009) formulated a decision-making model based on the detection probability. The decision-making model was solved by the genetic algorithm (GA). Kurdzo and Palmer (2011, 2012) applied GA to optimize the deployment of radar netting (Zhao et al., 2007; Gao, 2008; Yoon and Kim, 2013). Hu et al. (2010) proposed an improved continuous ant algorithm for deployment optimization of a sensor network. An ant colony optimization with three classes of ant transitions was proposed in Liu (2012) to solve the sensor deployment problem. Lian et al. (2012) studied an improved particle swarm optimization algorithm for sensor network deployment optimization (Liu and Fan, 2011). Most of these works built a simple deployment model with only one type of radar and one height detection level, and the applied optimization algorithms, such as GA and ant colony algorithms, have poor performance when it comes to generating a suitable deployment solution in complicated scenarios.

To address these issues, we build a more complicated model with multiple radar types and multiple height-detection levels, and propose an improved firefly algorithm (IFA) to generate satisfactory solutions. The firefly algorithm (FA) is a new swarm intelligence optimization algorithm proposed by Yang (2008), which takes inspiration from the flashing behavior of fireflies. In the classical FA, fireflies move toward more attractive fireflies in the whole population according to a movement equation in each iteration. The brighter firefly does not conduct any search, which may reduce the population diversity and cause the algorithm to be easily trapped in local optima. To solve these problems, our proposed IFA employs three strategies: (1) position initialization of fireflies in a chaotic sequence; (2) a neighborhood learning strategy with a feedback mechanism; (3) a chaotic local search by elite fireflies. The first strategy aims to improve the diversity of the firefly population compared to random initialization. The second is helpful in enhancing the exploitation ability and improving the convergence speed of the firefly algorithm. The last strategy helps the algorithm jump out of local optima.

2 Problem description and formulation

2.1 Problem description

Suppose that a surveillance network is composed of L radars of N types. Its responsibility area A is defined as the monitoring area of the surveillance network. The task of the surveillance network is to monitor air targets flying into the heights of the M layers in the area of responsibility. In particular, the surveillance network should focus on typical air targets in the main height layer.

To meet the mission requirements of the surveillance network, deployment optimization is employed to achieve the following objectives:

Objective 1 Maximize the possible detection range of the M height levels, which can be described by the airspace-covering coefficient ρ.

Definition 1 The proportion of the area of effective zones covered by all radars at height level k in the surveillance network to the area of the whole zone of responsibility is defined as ρ_k. Obviously, ρ_k is in the range $[0, 1]$.

$$\rho_k = \frac{\bigcup_{i=1}^{L} (A_{ik} \cap A)}{A}, \tag{1}$$

where A_{ik} indicates the area of the zone detected by radar i at height level k and A represents the area of the zone of responsibility. As shown in Fig. 1, the rectangular block represents the responsibility area and black dots are the radars. The corresponding detection area is represented as a circle. The dashed area is the airspace-covering area. Therefore, ρ_k at height level k is the proportion of the shaded area to the rectangular area.

The weighted sum of ρ_k across all M airspace layers is defined as the airspace-covering coefficient of the surveillance network, which is denoted by ρ:

$$\rho = \sum_{k=1}^{M} \omega_k \rho_k, \quad \sum_{k=1}^{M} \omega_k = 1, \tag{2}$$

where ω_k is the weight coefficient. It is set according to the importance of the surveillance height level generally.

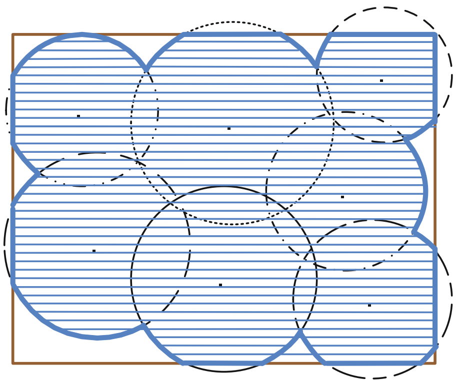

Fig. 1 Airspace-covering coefficient at level *k*

Objective 2 Have an appropriate airspace coverage redundancy to ensure continuity of airspace target tracking and avoid wasting resources due to excessive unnecessary coverage redundancy. This objective can be represented by the airspace-overlap coefficient μ.

Definition 2 μ_k is defined as the proportion of the effective area covered by two radars at height level k in the surveillance network to the area of the whole zone of responsibility, with the value in the range $[0, 1]$:

$$\mu_k = \frac{\left(\bigcup_{i,j=1}^{L} (A_{ik} \cap A_{jk})\right) \cap A}{A}, \tag{3}$$

where A_{jk} indicates the area of the zone detected by radar j at level k. As shown in Fig. 2, the shaded area represents the airspace-overlap area. Therefore, μ_k at height level k is the proportion of the shaded area to the rectangular area.

The weighted sum of μ_k across all M airspace layers is the indicator of the airspace-overlap coefficient of the surveillance network, denoted by μ:

$$\mu = \sum_{k=1}^{M} \omega_k \mu_k, \quad \sum_{k=1}^{M} \omega_k = 1. \tag{4}$$

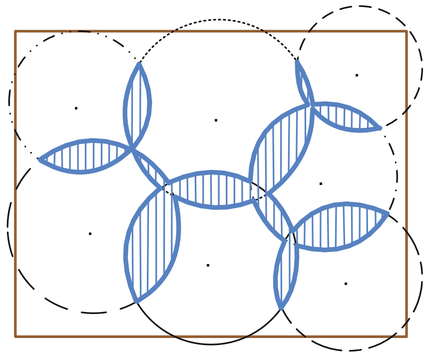

Fig. 2 Airspace-overlap coefficient at level *k*

2.2 Problem formulation

To maximize ρ while achieving sufficient μ, the objective function is formulated as follows:

$$\begin{aligned} \max F &= \lambda_1 \rho + \lambda_2 \mu \\ &= \lambda_1 \sum_{k=1}^{M} \omega_k \rho_k + \lambda_2 \sum_{k=1}^{M} \omega_k \mu_k, \end{aligned} \tag{5}$$

where F represents the comprehensive detection performance of the surveillance network, and λ_1 and λ_2 are the weights of the airspace-covering and airspace-overlap coefficients, respectively ($\lambda_1 + \lambda_2 = 1$).

For better coverage of the main height level and utilization of resources, two constraints must be met, defined as follows:

$$\begin{cases} \rho_0 > \rho_1, \\ \tau = 1 - \dfrac{\left(\bigcup\limits_{\substack{i,j,l,t=1,\\ i\neq j\neq l\neq t}}^{L} (A_{0i} \cap A_{0j} \cap A_{0l} \cap A_{0t})\right) \cap A}{A} \geq \tau_1. \end{cases} \tag{6}$$

The optimized results must achieve the basic requirement to maintain sufficient surveillance coverage of the main level. The main height level is usually the height of the network's focus. In the first constraint, the airspace-covering coefficient at the main height level should be greater than or equal to ρ_1 to reach a certain level.

On the other hand, the surveillance network should have the appropriate airspace-overlap coefficient at the main height level. Generally, it is reasonable that the effective area is covered by two radars simultaneously. However, it is considered to be a waste of resources if the surveillance airspace coverage has quadruple (or more) overlap. In the second constraint, A_{0i}, A_{0j}, A_{0l}, and A_{0t} indicate the area of the zone detected by radars i, j, l, and t at the main height level respectively, and τ describes the resource utilization which should not be less than τ_1 to reach a certain level.

3 Firefly algorithm and its variants

3.1 Firefly algorithm

The firefly algorithm (FA), which was inspired by the flashing patterns and behavior of fireflies, was first proposed by Yang (2008).

The FA is governed by the following three idealized rules:

1. All fireflies are unisex, so one firefly will be attracted to other fireflies regardless of sex.

2. Firefly attractiveness is proportional to brightness, and both attractiveness and brightness decrease with increased distance. For any two fireflies, the one with lower brightness will move toward the brighter one, and the brightest will move randomly.

3. The brightness of a firefly is determined by the landscape of the objective function.

In the FA, a group of N fireflies $\boldsymbol{X}_i$ ($i = 1, 2, \ldots, N$) is generated in the search space to find the optimal area. Each firefly has its own light intensity proportional to the value of the fitness function. The attractiveness β between two fireflies can be defined with distance r as

$$\beta = \beta_0 \cdot \mathrm{e}^{-\gamma \cdot r_{ij}^2}, \tag{7}$$

where β_0 is the attractiveness at $r = 0$ and γ is the light absorption coefficient which is usually set to 1.

The distance r_{ij} between any two fireflies $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ is expressed as the Euclidean distance as

$$r_{ij} = ||\boldsymbol{X}_i - \boldsymbol{X}_j|| = \sqrt{\sum_{d=1}^{D}(x_{id} - x_{jd})^2}, \tag{8}$$

where D is the dimension of firefly $\boldsymbol{X}_i$ or $\boldsymbol{X}_j$.

The movement of firefly $\boldsymbol{X}_i$ attracted to another brighter firefly $\boldsymbol{X}_j$, is determined by the following updating equation:

$$\boldsymbol{X}_i(t+1) = \boldsymbol{X}_i(t) + \beta(\boldsymbol{X}_j(t) - \boldsymbol{X}_i(t)) + \alpha(\mathbf{rand} - \mathbf{0.5}), \tag{9}$$

where $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ are the positions of fireflies in the search space, **rand** is a D-dimensional vector of random numbers that obeys uniform distribution over $[0, 1]$, α is the step length factor, and t is the current number of iterations.

The framework of the FA is presented in Algorithm 1.

Algorithm 1 Framework of the basic firefly algorithm

Require: population size N, maximum number of iterations MAX_G, step length factor α, attractiveness coefficient β, light absorption coefficient γ, and objective function $f(\boldsymbol{x}), \boldsymbol{x} = (x_1, x_2, \ldots, x_D)$
Initialize the population of fireflies $\boldsymbol{X}_i$ $(i = 1, 2, \ldots, N)$
while $t <$ MAX_G **do**
  for $i = 1$ to N **do**
    for $j = 1$ to N **do**
      if $f(\boldsymbol{X}_i) < f(\boldsymbol{X}_j)$ **then**
        Move $\boldsymbol{X}_i$ towards $\boldsymbol{X}_j$ according to Eq. (9)
      end if
    end for
  end for
  Calculate the light intensity in the new place
  t++;
end while

3.2 Firefly algorithm variants

Because the firefly algorithm literature is rapidly expanding, several variants of the FA have been proposed in recent years. A brief review of FA variants is presented below.

3.2.1 Standard FA with adaptive parameter

To improve the solution quality of the classical firefly algorithm, an improvement on the convergence of the algorithm is to decrease the step length factor α gradually as the optimum is approached. The new updating equations proposed by Yang (2010) are shown as

$$\begin{aligned}\boldsymbol{X}_i(t+1) = &\boldsymbol{X}_i(t) + \beta(\boldsymbol{X}_j(t) - \boldsymbol{X}_i(t)) \\ &+ \alpha(t) s_d(\mathbf{rand} - \mathbf{0.5}),\end{aligned} \tag{10}$$

$$\beta = \beta_{\min} + (\beta_0 - \beta_{\min})\mathrm{e}^{-\gamma \cdot r_{ij}^2}, \tag{11}$$

$$\alpha(t+1)=\alpha_0\theta^t, \tag{12}$$

$$s_d=X_{id}^{\max}-X_{id}^{\min}, \tag{13}$$

where θ is the step length reduction constant in range $(0,1)$, $\beta_{\min}$ is the minimum value of attractiveness β, s_d is the scale of each design variable, and $X_{id}^{\max}$ and $X_{id}^{\min}$ are the upper and lower bounds of firefly $\boldsymbol{X}_i$ in the d^{th} dimension respectively.

3.2.2 Wise step strategy FA (WSSFA)

In Yu et al. (2014), a wise step strategy was proposed to effectively improve the search ability of the classical firefly algorithm. This strategy considers the information of both firefly's historical best and population's global best solutions. The step length factor is calculated separately for each firefly at each iteration, and the updating formula is presented as

$$\alpha_i(t+1)=\alpha_i(t)-(\alpha_i(t)-\alpha_{\min})\cdot \mathrm{e}^{-|\mathbf{Gbest}-\mathbf{Pbest}_i|t/\mathrm{MAX\_}G}, \tag{14}$$

where t represents the current step, MAX_G is the maximum number of iterations of the algorithm, **Gbest** is the global best solution at the t^{th} iteration, $\mathbf{Pbest}_i$ is $\boldsymbol{X}_i$'s historical best solution searched, and $\alpha_{\min}$ is the minimum step length in the range $[0,1]$.

3.2.3 FA with chaos (CFA)

In Gandomi et al. (2013), chaotic parameter tuning was introduced into FA to enhance its global search ability for robust global optimization. Three tuning strategies, including tuning light absorption coefficient γ, attractiveness coefficient β, and both coefficients γ and β, were proposed to verify the efficiency of tuning different attraction parameters. Twelve different chaotic maps were investigated to tune the attraction parameters in the classical firefly algorithm:

$$\begin{aligned}\boldsymbol{X}_i(t+1)=&\boldsymbol{X}_i(t)+\mathrm{cs}(t)(\boldsymbol{X}_j(t)-\boldsymbol{X}_i(t))\\&+\alpha(t)s_d(\mathbf{rand}-\mathbf{0.5}),\end{aligned} \tag{15}$$

where $\mathrm{cs}(t)$ represents the chaotic sequence generating function.

By comparing different chaotic FAs, the algorithm that uses the Gauss map as its attractiveness coefficient is the best chaotic FA. Experiments reveal that the chaotic FA can clearly improve the quality of the optimization results.

3.2.4 FA with neighborhood search and random attraction (NSRaFA)

Although the classical FA has been empirically demonstrated to perform well on many optimization problems, it may get trapped in local optima when solving complex optimization problems. Recently, in Wang et al. (2017), an FA with a random attraction model and three neighborhood search strategies was proposed to improve the solution quality with a balance between algorithm exploration and exploitation abilities.

Dynamic parameter adjustment with step length factor α and attractiveness β was also used. The updating equations are presented as

$$\beta=(\beta_{\min}+(\beta_{\max}-\beta_{\min})\mathrm{e}^{-\gamma r_{ij}^2})\frac{t}{\mathrm{MAX\_}G}, \tag{16}$$

$$\alpha(t+1)=0.99\alpha(t), \tag{17}$$

where $\beta_{\min}$ and $\beta_{\max}$ are the minimum and maximum values of β respectively, and t is the current iteration number in the range of maximum iteration number MAX_G.

In the random attraction model, each firefly $\boldsymbol{X}_i$ communicates only with another randomly selected firefly $\boldsymbol{X}_j$, and thus it requires less computation time. For fireflies in each iteration, there are three different neighborhood search strategies, presented as

$$\boldsymbol{X}_i^1=r_1\boldsymbol{X}_i+r_2\mathbf{Pbest}_i+r_3(\boldsymbol{X}_{i1}-\boldsymbol{X}_{i2}), \tag{18}$$

$$\boldsymbol{X}_i^2=r_4\boldsymbol{X}_i+r_5\mathbf{Gbest}+r_6(\boldsymbol{X}_{i3}-\boldsymbol{X}_{i4}), \tag{19}$$

$$\boldsymbol{X}_i^3=\boldsymbol{X}_i+\mathrm{cauchy}(), \tag{20}$$

where $\boldsymbol{X}_{i1}$ and $\boldsymbol{X}_{i2}$ are two fireflies randomly selected from the k-neighborhood of $\boldsymbol{X}_i$, $\boldsymbol{X}_{i3}$, and $\boldsymbol{X}_{i4}$, which are randomly selected from the whole population, r_1, r_2, and r_3 (r_4, r_5, and r_6) are three uniform random numbers in the range $(0,1)$ that sum to 1, and cauchy() is a random number that obeys the Cauchy distribution with a unity scale factor.

3.2.5 Other FA variants

In addition to the FA variants presented above, there are several modifications and hybridizations applied to the classical firefly algorithm for solving various complex optimization problems. Readers can access a comprehensive review of these FA variants in Fister et al. (2012).

The efficiency of the firefly algorithm primarily depends on the variation of the step length factor and the formulation of attractiveness. Therefore, the main directions of modifications are the development of elitist and binary firefly algorithms (Farahani et al., 2012), Lévy flight based firefly algorithms (Yang, 2011), and parallel firefly algorithms (Subutic et al., 2012).

Heuristics can also be incorporated into an FA to improve its ability to solve specific problems. The following hybridizations have been applied to the classical firefly algorithm: genetic algorithm (Luthra and Pal, 2011), differential evolution, neural network (Hassanzadeh et al., 2012), and ant colony (Aruchamy and Vasantha, 2011).

4 Our proposed improved firefly algorithm

In this section, a new FA variant is proposed to improve the quality of complicated optimization problem solutions. The improved firefly algorithm applies mainly three strategies: chaotic firefly position initialization, a neighborhood learning strategy with a feedback mechanism, and elite firefly chaotic local search.

4.1 Firefly position initialization with chaotic sequence

The FA is a population-based swarm intelligence method, and thus the initial firefly positions have a significant impact on its performance. In the classical FA, firefly positions are initialized by random generation methods. This may generate an extremely uneven distribution of the firefly population in which the results fall into local optima. Chaos, on the other hand, has the characteristics of randomness, regularity, and boundedness (Gandomi et al., 2013). Its sensitivity to the initial values can make the variable traverse all states without repeat in a certain range. Therefore, using the chaotic sequence to initialize firefly positions can improve the diversity of the population and enhance the global search ability.

In this study, the logistic chaotic map (Gandomi et al., 2013) is applied to generate the initial firefly positions. Its iteration equation is presented as

$$x_{k+1} = \mu \cdot x_k \cdot (1 - x_k), \tag{21}$$

where μ is the control parameter and x_k is the chaotic variable. When $\mu = 4$ and $0 < x_0 < 1$ ($x_0 \notin \{0, 0.25, 0.50, 0.75, 1.00\}$), the generated sequence presents chaotic characteristics.

We use the following equation to map the generated chaotic variables to the search space:

$$X_{id} = \mathrm{Ld} + x_{id}(\mathrm{Ud} - \mathrm{Ld}), \tag{22}$$

where Ud and Ld are the upper and lower bounds of the d^{th} dimension in the search space respectively, and X_{id} is the d^{th} dimension coordinate of the i^{th} firefly.

4.2 Neighborhood learning strategy with feedback mechanism

To enhance the exploitation ability and the convergence speed of the firefly algorithm, a neighborhood learning strategy with a feedback mechanism is applied for a specific percentage of fireflies in each iteration.

Assume that firefly population G consists of N fireflies $\{\boldsymbol{X}_i\}$ ($i = 1, 2, \ldots, N$) in the search space. The k-neighborhood of a specific firefly $\boldsymbol{X}_i$ can be defined as the group of $(2k + 1)$ fireflies that are closer to $\boldsymbol{X}_i$ according to their indices. The boundary of firefly indices is periodic, which means that the distance between $\boldsymbol{X}_1$ and $\boldsymbol{X}_N$ is 1. For example, the two-neighborhood of $\boldsymbol{X}_1$ consists of five fireflies, $\boldsymbol{X}_{N-1}, \boldsymbol{X}_N, \boldsymbol{X}_1, \boldsymbol{X}_2$, and $\boldsymbol{X}_3$. A visual depiction of the k-neighborhood is shown in Fig. 3.

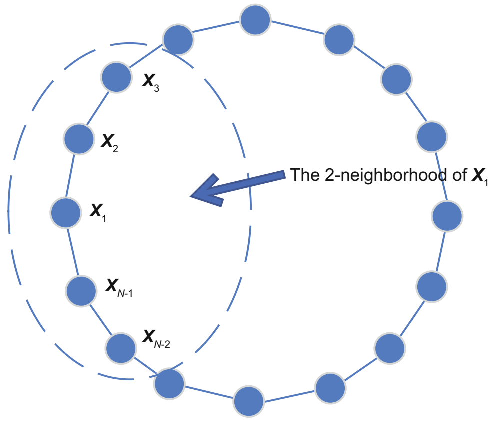

Fig. 3 Circular topology and k-neighborhood of $\boldsymbol{X}_i$

In each iteration, a specific percentage p_n of fireflies will apply a neighborhood learning strategy to update their positions. A feedback mechanism is introduced to choose the learning objective in the k-neighborhood of each firefly. In detail, firefly $\boldsymbol{X}_i$ will learn from firefly $\boldsymbol{X}_j$ who is in $\boldsymbol{X}_i$'s k-neighborhood and has the highest score S_{ij} as

$$S_{ij} = \frac{\mathrm{LE}_{ij}}{L_{ij}+1}, \tag{23}$$

where L_{ij} represents the number of times that firefly $\boldsymbol{X}_i$ has learned from $\boldsymbol{X}_j$ so far and LE_{ij} represents the number of effective learning times that firefly $\boldsymbol{X}_j$ can lead $\boldsymbol{X}_i$ to a better solution. This feedback mechanism could enhance the ability to learn from well-performing neighbors and give opportunities to others.

The movement equation of firefly $\boldsymbol{X}_i$ learning from $\boldsymbol{X}_j$ is presented as

$$\boldsymbol{X}_i = r_1\boldsymbol{X}_i + r_2(\boldsymbol{X}_j - \boldsymbol{X}_i) + r_3(\boldsymbol{X}_{i1} - \boldsymbol{X}_{i2}), \tag{24}$$

where r_1, r_2, and r_3 are three uniform random numbers in the range $(0, 1)$ that sum to 1, $\boldsymbol{X}_{i1}$ and $\boldsymbol{X}_{i2}$ are two fireflies that are randomly selected from the k-neighborhood of $\boldsymbol{X}_i$, and the value of $\boldsymbol{X}_{i1}$ minus $\boldsymbol{X}_{i2}$ represents a randomization item in the neighborhood region.

4.3 Chaotic local search of elite fireflies

In the standard FA, fireflies move toward the more attractive individuals as the equation is updated in each iteration. The movement of each firefly is determined by other brighter fireflies' positions in the whole population. This may reduce the population diversity and easily get trapped in local optima.

To address the above issue, a chaotic local search by elite fireflies for a better solution among the global best solutions is integrated in our FA. A visual depiction of elite fireflies' chaotic local search is shown in Fig. 4. In each iteration, we select a specific percentage p_e of best fireflies as an elite group E. Then a chaotic sequence initialization group G_l around each firefly in E is generated for further local search. Assuming that elite firefly $\boldsymbol{X}_i$ is selected for chaotic local search, $G_l = \{X_{li}\}$ can be generated by the following equations:

$$X_{li} = \mathrm{Ld}(t) + x_{li}(\mathrm{Ud}(t) - \mathrm{Ld}(t)), \tag{25}$$

$$\mathrm{Ld}(t) = \max(\mathrm{Ld}_0, \boldsymbol{X}_i - p\boldsymbol{S}), \tag{26}$$

$$\mathrm{Ud}(t) = \min(\mathrm{Ud}_0, \boldsymbol{X}_i + p\boldsymbol{S}), \tag{27}$$

$$p = \frac{1}{1+\mathrm{e}^{0.004t+1}}, \tag{28}$$

where x_{li} is a chaotic sequence with a logistic map, Eq. (25) maps the chaotic sequence to the search space around elite firefly $\boldsymbol{X}_i$, p is a scale factor to make the local search region decrease gradually as the optima are approached, and $\boldsymbol{S}$ is the scale of the search space. This could enhance exploration by providing an ability to jump out of the local optima in the early stage and improve the solution quality by local search in a relatively small area in the final stage.

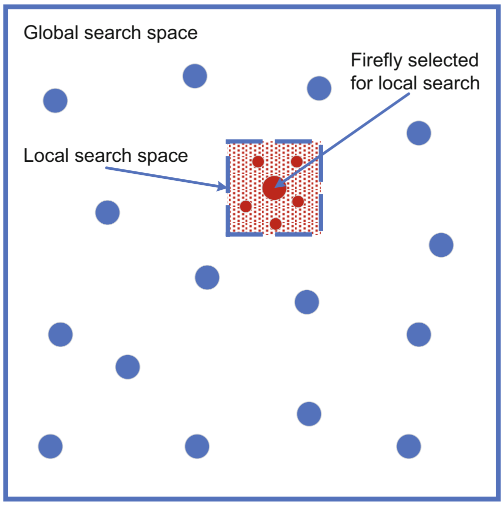

Fig. 4 Chaotic local search of $\boldsymbol{X}_i$ (the large red node), where the small red nodes represent the generated fireflies in the local space. References to color refer to the online version of this figure

For each elite firefly $\boldsymbol{X}_i$, the best solution in the chaotic local search group G_{li} will be selected as the new $\boldsymbol{X}_i$.

4.4 Framework of the improved firefly algorithm

The main steps of the IFA are described in Algorithm 2, where N is the firefly population size, FE is the number of calculated fitness function, and MAX_FEs is the maximum number of function evaluations.

In firefly attraction movement, a self-adaptive parameter of movement equation is adopted and a step length reduction constant is adjusted as follows:

$$\alpha(t) = \alpha_0 \left(\frac{1}{9000}\right)^{t/\mathrm{MAX\_}G}. \tag{29}$$

This reduces the step length α from α_0 to 10^{-4} exponentially.

Algorithm 2 Proposed improved firefly algorithm

Require: objective function $f(\boldsymbol{x}), \boldsymbol{x} = (x_1, x_2, \ldots, x_D)$
Initialize the population of fireflies $\boldsymbol{X}_i$ $(i = 1, 2, \ldots, N)$ with chaotic sequence
while FE<MAX_FEs **do**
  for $i = 1$ to N **do**
    for $j = 1$ to N **do**
      if $f(\boldsymbol{X}_i) < f(\boldsymbol{X}_j)$ **then**
        Move firefly $\boldsymbol{X}_i$ towards $\boldsymbol{X}_j$ according to Eq. (9)
      end if
    end for
  end for
  /* Neighborhood learning strategy with feedback */
  /* mechanism */
  for $i = 1$ to N **do**
    if $r < p_n$ **then**
      Move firefly $\boldsymbol{X}_i$ with neighborhood learning strategy
      FE++
    end if
  end for
  /* Elite fireflies' chaotic local search */
  Select p_e of best fireflies as an elite group E
  for $i = 1$ in E **do**
    Apply chaotic search to firefly $\boldsymbol{X}_i$
    Replace $\boldsymbol{X}_i$ with the best solution of chaotic local search
    FE++
  end for
  Calculate the light intensity in the new place
  FE++
end while

In each iteration, the neighborhood learning strategy and chaotic local search by elite fireflies are combined to achieve a better trade-off between exploration and exploitation.

5 Experiments

5.1 Algorithm performance on benchmark functions

To evaluate the performance of the proposed IFA, we compared the solution quality on 12 standard benchmark functions listed in Table 1. Among these functions, f_1 (sphere), f_2, f_3, f_4 (Schwefel), and f_5 (Rosenbrock) are unimodal functions, f_6 is a step function with one global minimum, f_7 is a noisy quartic function including a stochastic term, and f_8–f_{12} are multimodal functions with many local minima. The dimensions of all these benchmark functions were set to 30. These diverse characteristics allow us to comprehensively test the performance of the proposed IFA.

We compared the performance of the IFA, the classical FA, and four other recently proposed variants (Table 2). To have a fair comparison, the population size N and MAX_FEs of all algorithms were set to 20 and 5.0e + 05, respectively. The remaining parameters were set based on the information in

Table 1 Benchmark functions

Name	Function	Search range	Global optimum
Sphere	$f_1(x) = \sum_{i=1}^{D} x_i^2$	$[-100, 100]$	0
Schwefel 2.22	$f_2(x) = \sum_{i=1}^{D} \lvert x_i\rvert + \prod_{i=1}^{D} \lvert x_i\rvert$	$[-10, 10]$	0
Schwefel 1.2	$f_3(x) = \sum_{i=1}^{D}(\sum_{j=1}^{i} x_j)^2$	$[-100, 100]$	0
Schwefel 2.21	$f_4(x) = \max_{i=1,2,\ldots,D} \lvert x_i\rvert$	$[-100, 100]$	0
Rosenbrock	$f_5(x) = \sum_{i=1}^{D}[100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$	$[-30, 30]$	0
Step	$f_6(x) = \sum_{i=1}^{D} \lfloor \lvert x_i\rvert + 0.5 \rfloor$	$[-100, 100]$	0
Quartic with noise	$f_7(x) = \sum_{i=1}^{D}(i \cdot x_i^4) + \mathrm{random}[0, 1)$	$[-1.28, 1.28]$	0
Schwefel 2.26	$f_8(x) = -\sum_{i=1}^{D}(x_i \sin \sqrt{\lvert x_i\rvert})$	$[-500, 500]$	0
Rastrigin	$f_9(x) = -\sum_{i=1}^{D}[x_i^2 - 10\cos(2\pi x_i) + 10]$	$[-5.12, 5.12]$	0
Ackley	$f_{10}(x) = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right) + 20 + \mathrm{e}$	$[-32, 32]$	0
Griewank	$f_{11}(x) = 1 + \sum_{i=1}^{D} x_i^2/4000 - \prod_{i=1}^{D}\cos(x_i/\sqrt{i})$	$[-600, 600]$	0
Penalized	$f_{12}(x) = \frac{\pi}{D}\Big(10\sin^2(\pi y_1) + (y_n - 1)^2 + \sum_{i=1}^{D-1}(y_i - 1)^2(1 + 10\sin^2(\pi y_{i+1}))\Big) + \sum_{i=1}^{D} u(x_i, 10, 100, 4)$, $y_i = 1 + \frac{1}{4}(x_i + 1)$, $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a, \\ 0, & -a \le x_i \le a, \\ k(-x_i - a)^m, & x_i < a. \end{cases}$	$[-50, 50]$	0

the literature listed in Table 2. Specifically, α was set to 0.2 for the classical FA (Yang, 2008). In the standard FA (Yang, 2010), $\beta_{\min}$ was set to 0.2, and α_0 and θ were set to 0.9 and 0.95, respectively. For WSSFA (Yu et al., 2014), $\alpha_{\min}$ was set to 0.04. For CFA (Gandomi et al., 2013), the Gauss map was used to update parameter β. For NSRaFA (Wang et al., 2017), α_0, $\beta_{\max}$, and $\beta_{\min}$ were set to 0.5, 0.9, and 0.3, respectively. For the proposed IFA, the percentage for neighborhood learning p_n and chaotic local search p_e were set to 0.2 and 0.1 respectively, the chaotic search group was set to 10, and α_0, $\beta_{\max}$, and $\beta_{\min}$ were set to 0.5, 0.9, and 0.1, respectively. All the experiments were repeated 30 times, and the results are presented in Tables 3 and 4.

In Table 3, the experimental results are illustrated for the classical FA, standard FA, WSSFA, CFA, NSRaFA, and the proposed IFA. The comparison results are summarized as $w/t/l$, which means that our proposed IFA wins in w functions, ties in t functions, and loses in l functions compared with the other algorithms.

The experimental results indicate that the proposed IFA can obtain the best solution on almost all functions; only on f_8 does NSRaFA perform better than IFA. From the deviation results in Table 4, IFA is more stable than the other algorithms.

All the results above indicate that the performance of IFA is better than those of the other recently proposed FA variants on benchmark functions. The good performance of IFA demonstrates that it achieves a better balance between exploration and exploitation with integrated strategies, especially on multimodal functions. Because the problem of radar network deployment optimization is nonlinear and complicated, IFA could be more suitable for working on this problem.

Table 2 FA variants used for comparison

Algorithm	Year	Reference
Classical FA	2008	Yang (2008)
Standard FA	2010	Yang (2010)
WSSFA	2014	Yu et al. (2014)
CFA	2013	Gandomi et al. (2013)
NSRaFA	2017	Wang et al. (2017)
Our proposed IFA	2018	–

5.2 IFA for deployment optimization of the radar network

5.2.1 Scenario description

Assume that there is a surveillance network whose area of responsibility is a square with side length 400 km. It includes two types of radar, A and B, whose numbers are equal. There are four height levels, i.e., 500 m, 3000 m, 5000 m, and 10 000 m.

The maximum detection distances of type-A and type-B radars at the above-mentioned four altitudes are shown in Table 5.

Without considering the effect of terrain masking and other factors, the detection range of a radar at any height level can be represented as a circle whose radius is the maximum detection distance of the radar at that height level. As the height increases, the maximum detection distance of the radar

Table 3 Experimental results of the mean value on benchmarks

Function	Mean value					
	Classical FA	Standard FA	WSSFA	CFA	NSRaFA	Our proposed IFA
f_1	6.87e + 04	6.89e − 03	7.11e + 04	3.07e + 04	8.55e − 91	**0.00e + 00**
f_2	1.72e + 03	2.31e + 00	3.31e + 05	8.67e + 10	4.56e − 47	**0.00e + 00**
f_3	1.51e + 05	3.63e + 03	1.58e + 05	8.45e + 04	6.49e − 89	**0.00e + 00**
f_4	8.40e + 01	6.76e + 00	8.74e + 01	6.78e + 01	2.90e − 46	**0.00e + 00**
f_5	1.79e + 08	6.07e + 02	2.79e + 08	6.61e + 07	2.89e + 01	**4.87e − 09**
f_6	1.25e + 03	6.20e + 00	1.19e + 03	8.09e + 02	**0.00e + 00**	**0.00e + 00**
f_7	3.01e + 01	4.13e − 01	3.82e + 01	2.72e + 01	6.44e − 02	**5.32e − 02**
f_8	−1.94e + 03	−6.32e + 03	−1.98e + 03	−2.38e + 03	**−9.21e + 03**	−5.46e + 03
f_9	2.95e + 02	5.50e + 01	3.40e + 02	3.53e + 02	**0.00e + 00**	**0.00e + 00**
f_{10}	2.00e + 01	7.21e − 01	2.04e + 01	1.88e + 01	**4.44e − 16**	**4.44e − 16**
f_{11}	6.31e + 02	2.25e − 02	6.25e + 02	2.82e + 02	**0.00e + 00**	**0.00e + 00**
f_{12}	4.51e + 08	1.33e + 00	6.35e + 08	1.26e + 08	6.92e − 01	**1.31e − 03**
$w/t/l$	12/0/0	12/0/0	12/0/0	12/0/0	7/4/1	–

Boldface indicates the best results among the algorithms

Table 4 Experimental results of the standard deviation value on benchmarks

Function	Standard deviation value					
	Classical FA	Standard FA	WSSFA	CFA	NSRaFA	Our proposed IFA
f_1	6.66e + 03	5.68e − 03	6.83e + 03	1.42e + 04	4.60e − 90	**0.00e+00**
f_2	6.97e + 03	1.63e + 00	5.75e + 05	4.62e + 11	2.46e − 46	**0.00e + 00**
f_3	3.96e + 04	1.67e + 03	6.05e + 04	5.59e + 04	3.49e − 88	**0.00e + 00**
f_4	1.40e + 00	2.84e + 00	3.90e + 00	1.02e + 01	1.56e − 45	**0.00e + 00**
f_5	1.75e + 07	1.16e + 03	3.91e + 07	5.86e + 07	5.56e − 02	**2.61e − 08**
f_6	5.39e + 01	6.20e + 00	5.35e + 01	2.23e + 02	**0.00e + 00**	**0.00e + 00**
f_7	6.83e + 00	1.43e − 01	2.11e + 01	2.23e + 01	4.89e − 02	**3.95e − 02**
f_8	4.74e + 02	1.13e + 03	4.63e + 02	5.04e + 02	**5.35e + 02**	8.87e + 02
f_9	1.10e + 01	1.61e + 01	1.13e + 01	4.03e + 01	**0.00e + 00**	**0.00e + 00**
f_{10}	1.09e − 01	5.29e − 01	7.32e − 02	9.53e − 01	**0.00e + 00**	**0.00e + 00**
f_{11}	7.81e + 01	2.43e − 02	7.77e + 01	1.22e + 02	**0.00e + 00**	**0.00e + 00**
f_{12}	6.98e + 07	8.93e − 01	1.19e + 08	1.62e + 08	2.26e − 01	**2.62e − 03**
$w/t/l$	12/0/0	12/0/0	12/0/0	12/0/0	7/4/1	–

Boldface indicates the best results among the algorithms

Table 5 Maximum detection distances of type-A and type-B radars

Type	Maximum detection distance (km)			
	500 m	3000 m	5000 m	10 000 m
A	90	200	220	250
B	30	210	270	360

shows an increasing trend. Assume that the surveillance network focuses on the height of 500 m and that its airspace coverage is required to be more than 50%. Thus, ρ_0 was set to be greater than 0.5. The detection range of the type-A radar is a circle with a radius of 90 km. The detection range of the type-B radar is circular with a radius of 30 km. The sum of the two circular areas is about 28 000 km^2. Although the area of responsibility is 160 000 km^2, there are a total of 10 radars in the surveillance network, 5 type-A radars and 5 type-B radars.

To avoid wasting resources, the proportion of the area covered by more than four layers to the area of the whole zone of responsibility should be less than 20%. Parameter τ was set to be larger than 0.8. The weight values at the four altitudes are 0.4, 0.2, 0.2, and 0.2, respectively.

In objective function (5), λ_1 and λ_2 were set to 0.7 and 0.3 respectively, and constraints ρ_1 and τ_1 were set to 0.5 and 0.2 respectively according to usual practice.

5.2.2 Experimental results comparison

Based on the simple radar deployment optimization problem described above, the algorithms listed in Table 2 were applied to generate better radar deployment strategies. The parameters of the algorithms were the same as in the experiment on benchmark functions. For the classical GA (Srinivas and Patnaik, 1994), the cross probability and mutation probability were set to 0.4 and 0.1, respectively. For ant colony optimization (ACO) (Yu et al., 2007), the coefficient of the intensity of the trail was set to 0.9 and the heuristic coefficient was set to 1. All the experiments were repeated 30 times. Table 6 describes the performance of the surveillance network after optimization.

First, we used a two-sample t-test to compare the results of different algorithms. The two-sample t-test is a parametric test that compares the location parameter of two independent data samples. It can be used to determine if two sets of data are significantly different from each other. The test statistic is

$$t = (\bar{x} - \bar{y}) \Big/ \sqrt{\frac{S_x^2}{n} + \frac{S_y^2}{m}}, \tag{30}$$

where $\bar{x}$ and $\bar{y}$ are the sample means, S_x and S_y are the sample standard deviations, and n and m are the sample sizes. For simplicity, we used the following formula to do t-test in MATLAB:

$$h = \text{ttest2}(x, y), \tag{31}$$

where x and y represent two sets of data from the results of different algorithms and h is the hypothesis test result. If h equals 1, it indicates that x and y are from different distributions at the statistical significance level of 5%. If h equals 0, the conclusion is the opposite. After we did the two-sample t-test between the algorithms listed in Table 2 in pairs, all of the outputs h equal 1. Therefore, the results show a significant difference between the chosen algorithms for comparison. From the maximum, minimum, and mean results in Table 6, we can see that the proposed IFA has better performance than the other algorithms. Moreover, the experimental variance of IFA is $5.9443\mathrm{e}-06$, which indicates that the robustness of IFA is significantly better than that of the other seven algorithms.

Table 6 Results of different algorithms

Algorithm	F			
	Mean	Maximum	Minimum	Variance
Classical GA	0.8085	0.8216	0.8070	1.7636e − 05
ACO	0.8220	0.8245	0.8067	2.6027e − 05
Classical FA	0.7879	0.8012	0.7776	2.9275e − 05
Standard FA	0.8243	0.8272	0.8106	1.2849e − 05
WSSFA	0.7874	0.8005	0.7788	2.7269e − 05
CFA	0.7964	0.8137	0.7819	3.3682e − 05
NSRaFA	0.8281	0.8304	0.8242	8.0651e − 06
Proposed IFA	**0.8380**	**0.8392**	**0.8355**	**5.9443e − 06**

F: comprehensive detection performance of the surveillance network. Boldface indicates the best results among the algorithms

Fig. 5 presents the average evolution curves of eight algorithms. The proposed IFA has a longer period of continuous evolution to search for a better solution, while NSRaFA and the standard FA converge too quickly and fall into local optima at an early stage. For the classical FA, WSSFA, and CFA, their exploration ability is too weak for continuous evolution to find better solutions. Therefore, we can conclude that our proposed IFA strategies play a significant role in obtaining a trade-off between exploration and exploitation.

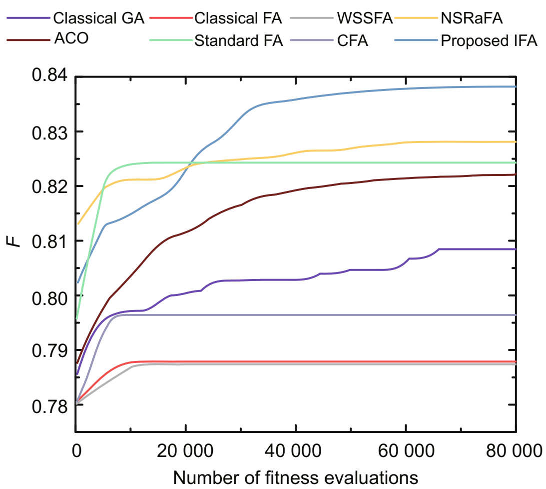

Fig. 5 Average evolution curves under different algorithms. References to color refer to the online version of this figure

5.2.3 Analysis of the IFA-optimized radar network deployment strategy

In this subsection, the IFA optimized radar network deployment strategy is analyzed. First, the detection ranges of the radar at 500 m and 3000 m height are shown in Figs. 6 and 7, respectively. At the height of 500 m, type-A and type-B radars cover

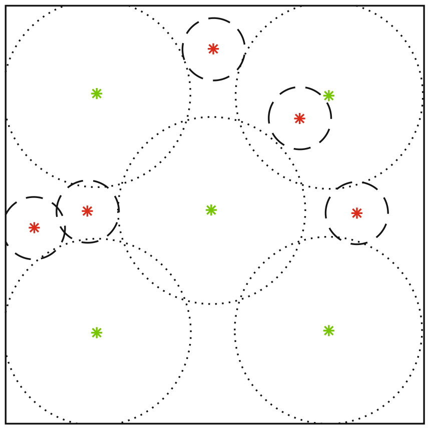

Fig. 6 Radar detection scope at the height of 500 m. Red dots: locations of type-A radars; green dots: locations of type-B radars; solid circles: detection ranges of type-A radars; dotted circles: detection ranges of type-B radars. References to color refer to the online version of this figure

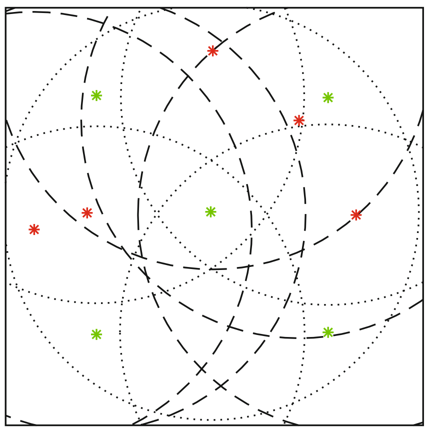

Fig. 7 Radar detection scope at the height of 3000 m. Red dots: locations of type-A radars; green dots: locations of type-B radars; solid circles: detection ranges of type-A radars; dotted circles: detection ranges of type-B radars. References to color refer to the online version of this figure

more than half of the area of responsibility, but the airspace overlap is very small. At the height of 3000 m, type-A and type-B radars cover the entire area of responsibility, and there is also a considerable overlap in airspace. Experimental data are listed in Table 7.

As shown in Table 7, the airspace-covering coefficient and the airspace-overlap coefficient are both 1 at the 5000 m and 10 000 m height layers. This indicates that the network has achieved seamless coverage and continuous tracking.

Table 7 Performance parameters at different heights after optimization

Height (m)	ρ	μ
500	0.8453	0.0336
3000	1	0.9999
5000	1	1
10 000	1	1

6 Conclusions

Radar network detection is an important approach for defending against UAV invasion. In this paper, a more complicated radar deployment optimization model with multiple radar types and multiple height detection levels has been described. We converted it into a single-objective optimization problem by introducing the airspace-covering coefficient and airspace-overlap coefficient in an objective function. To solve this problem, an improved firefly algorithm has been proposed. It employs three strategies: (1) position initialization of fireflies in a chaotic sequence; (2) a neighborhood learning strategy with a feedback mechanism; (3) chaotic local search by elite fireflies. Experimental results on 12 famous benchmark functions and in a classical radar deployment scenario indicated that our approach achieves a good trade-off between exploration and exploitation and has much better performance than the classical FA and four recently proposed FA variants.

References

Aruchamy R, Vasantha KD, 2011. A comparative performance study on hybrid swarm model for micro array data. *Int J Comput Appl*, 30:10-14.

Baker CJ, Hume AL, 2003. Netted radar sensing. *IEEE Aerosp Electron Syst Mag*, 18(2):3-6. https://doi.org/10.1109/MAES.2003.1183861

Blake LV, 1986. Radar Range-Performance Analysis. Artech House, Inc., Norwood, MA, USA.

Difranco JV, Kaiteris C, 1981. Radar performance review in clear and jamming environments. *IEEE Trans Aerosp Electron Syst*, AES-17(5):701-710. https://doi.org/10.1109/TAES.1981.309102

Farahani SM, Abshouri AA, Nasiri B, et al., 2012. Some hybrid models to improve firefly algorithm performance. *Int J Artif Intell*, 8(12):97-117.

Fister I, Fister IJr, Yang XS, et al., 2012. A comprehensive review of firefly algorithms. *Swarm Evol Comput*, 13:34-46. https://doi.org/10.1016/j.swevo.2013.06.001

Gandomi AH, Yang XS, Talatahari S, et al., 2013. Firefly algorithm with chaos. *Commun Nonl Sci Numer Simul*, 18(1):89-98. https://doi.org/10.1016/j.cnsns.2012.06.009

Gao S, 2008. Research on optimum deployment problem of radar. Proc ISECS Int Colloquium on Computing, Communication, Control, and Management, p.466-469. https://doi.org/10.1109/CCCM.2008.100

Hassanzadeh T, Faez K, Seyfi G, 2012. A speech recognition system based on structure equivalent fuzzy neural network trained by firefly algorithm. Proc Int Conf on Biomedical Engineering, p.63-67. https://doi.org/10.1109/ICoBE.2012.6178956

Hu CH, Jiang W, Wang TJ, 2010. Continuous ant algorithm based on cooperation in radar network optimization. Proc 17th Int Conf on Management Science & Engineering, p.224-233. https://doi.org/10.1109/ICMSE.2010.5719809

Kurdzo JM, Palmer RD, 2011. On the use of genetic algorithms for optimization of a multi-band, multi-mission radar network. Proc IEEE RadarCon, p.231-236. https://doi.org/10.1109/RADAR.2011.5960534

Kurdzo JM, Palmer RD, 2012. Objective optimization of weather radar networks for low-level coverage using a genetic algorithm. *J Atmos Ocean Technol*, 29(6):807-821. https://doi.org/10.1175/JTECH-D-11-00076.1

Lian XY, Zhang J, Chen C, et al., 2012. Three-dimensional deployment optimization of sensor network based on an improved particle swarm optimization algorithm. Proc 10th World Congress on Intelligent Control and Automation, p.4395-4400. https://doi.org/10.1109/WCICA.2012.6359220

Liu WT, Fan ZY, 2011. Coverage optimization of wireless sensor networks based on chaos particle swarm algorithm. *J Comput Appl*, 31(2):338-340. https://doi.org/10.3724/SP.J.1087.2011.00338

Liu XX, 2012. Sensor deployment of wireless sensor networks based on ant colony optimization with three classes of ant transitions. *IEEE Commun Lett*, 16(10):1604-1607. https://doi.org/10.1109/LCOMM.2012.090312.120977

Luthra J, Pal SK, 2011. A hybrid firefly algorithm using genetic operators for the cryptanalysis of a monoalphabetic substitution cipher. Proc World Congress on Information and Communication Technologies, p.202-206. https://doi.org/10.1109/WICT.2011.6141244

Srinivas M, Patnaik LM, 1994. Genetic algorithms: a survey. *Computer*, 27(6):17-26. https://doi.org/10.1109/2.294849

Srinivasan R, 1986. Distributed radar detection theory. *IEE Proc F Commun Radar Signal Process*, 133(1):55-60. https://doi.org/10.1049/ip-f-1.1986.0010

Subutic M, Tuba M, Stanarevic N, 2012. Parallelization of the firefly algorithm for unconstrained optimization problems. *Latest Adv Inform Sci Appl*, 22(3):264-269.

Wang H, Cui ZH, Sun H, et al., 2017. Randomly attracted firefly algorithm with neighborhood search and dynamic parameter adjustment mechanism. *Soft Comput*, 21(18):5325-5339.
https://doi.org/10.1007/s00500-016-2116-z

Yang L, Liang J, Liu WW, 2013. Graphical deployment strategies in radar sensor networks (RSN) for target detection. *EURASIP J Wirel Commun Netw*, 2013(1):55.
https://doi.org/10.1186/1687-1499-2013-55

Yang LP, Xiong JJ, Cui J, 2009. Method of optimal deployment for radar netting based on detection probability. Proc Int Conf on Computational Intelligence and Software Engineering, p.1-5.
https://doi.org/10.1109/CISE.2009.5364961

Yang XS, 2008. Nature-Inspired Metaheuristic Algorithms. Luniver Press, Frome, UK.

Yang XS, 2010. Nature-Inspired Metaheuristic Algorithms (2nd Ed.). Luniver Press, Frome, UK.

Yang XS, 2011. Metaheuristic optimization: algorithm analysis and open problems. Proc 10th Int Symp on Experimental Algorithms, p.21-32.
https://doi.org/10.1007/978-3-642-20662-7_2

Yoon Y, Kim YH, 2013. An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks. *IEEE Trans Cybern*, 43(5):1473-1483.
https://doi.org/10.1109/TCYB.2013.2250955

Yu L, Liu K, Li KS, 2007. Ant colony optimization in continuous problem. *Front Mech Eng China*, 2(4):459-462.
https://doi.org/10.1007/s11465-007-0079-6

Yu SH, Su SB, Lu QP, et al., 2014. A novel wise step strategy for firefly algorithm. *Int J Comput Math*, 91(12):2507-2513.
https://doi.org/10.1080/00207160.2014.907405

Zhao CH, Yu ZQ, Chen P, 2007. Optimal deployment of nodes based on genetic algorithm in heterogeneous sensor networks. Proc Int Conf on Wireless Communications, Networking and Mobile Computing, p.2743-2746.
https://doi.org/10.1109/WICOM.2007.681

Zheng GQ, Zheng Y, 2011. Radar netting technology & its development. Proc IEEE CIE Int Conf on Radar, p.933-937.
https://doi.org/10.1109/CIE-Radar.2011.6159694