Journal of Computer Science and Artificial Intelligence
ISSN: 3078-8242 (Print), ISSN: 3078-2309 (Online) | Vol. 6, No. 2, 2026

Optimization of Smoke Screen Interference Munition Deployment Strategy Based on Improved Particle Swarm and NSGA-II Algorithms

Shuyuan Ye *, Weiyao Shi, Jiaying Guo

College of Civil and Transportation Engineering, Hohai University, Nanjing, 210024, China
# These authors contributed equally

* **Corresponding author**: (Email: 19905232688@163.com)

Abstract: To address the threat posed by precision-guided missiles to ground targets in modern warfare, this study focuses on optimizing deployment strategies for smoke screen interference bombs using unmanned aerial vehicles. The research examines two typical scenarios: single-drone-single-bomb and multi-drone-multi-bomb operations. A geometric shielding model based on Newtonian kinematics is established to determine the intersection conditions between smoke cloud clusters and missile-target line-of-sight, enabling precise calculation of shielding duration. For single-drone scenarios, an improved Particle Swarm Optimization algorithm is employed to optimize flight direction, speed, smoke deployment timing, and detonation delay, with the primary goal of maximizing real target shielding duration. In multi-drone scenarios, a multi-objective optimization model is developed to maximize shielding time while minimizing ammunition consumption and enhancing uniformity of coverage, incorporating the military deception effect of decoy targets. The collaborative deployment strategy is solved using the Non-Stochastic Genetic Algorithm-II. The proposed model and strategies provide valuable references for practical applications of UAV smoke screen interference technology.

Keywords: Geometric shielding model, Single objective optimization, Multi objective optimization, Particle Swarm Optimization Algorithm, NSGA-II algorithm.

1. Introduction

In modern warfare, precision-guided missiles pose a formidable threat to stationary ground targets due to their high hit rates and rapid flight speeds. To enhance target survivability, smoke screen interference technology—a cost-effective passive countermeasure—has been widely adopted for battlefield protection. Smoke screen bombs generate aerosol clouds through detonation, creating optical barriers between targets and missiles that disrupt guidance systems, thereby protecting the targets. With advancements in drone technology, unmanned aerial vehicles (UAVs) equipped with smoke screen bombs have become ideal platforms for deployment. Their high maneuverability and flexible deployment capabilities allow real-time adjustments to targeting strategies based on battlefield conditions, significantly improving interference precision and effectiveness. To optimize deployment efficiency, extend target coverage duration, and reduce operational costs, research into strategic deployment of smoke screen bombs is essential.

Currently, scholars have conducted extensive research on smoke screen interference technology. In the field of smoke screen concealment models, Luo Ruiyao [1] and colleagues established a model for deploying smoke screens on defense drones, developing a strategy to continuously conceal and protect friendly targets with minimal ammunition consumption. Chen [2] and his team derived and constructed mathematical models for transmission rates and effective shielding areas, discussing methods for assigning background and smoke gray values in transmission rate models. Regarding drone swarm interference, Zhang [3] and his research team investigated distributed drone systems with limited communication range, overcoming the limitations of traditional multi-source DOA algorithms. Zhi Cun [4] and colleagues explored a collaborative scheme combining external active interference with smoke screen technology. However, current studies still lack research on coordinated drone deployment of smoke screens. In optimization algorithm applications, Particle Swarm Optimization (PSO) has been widely adopted for rapid and robust drone swarm control due to its fast convergence and ease of implementation [5]. The NSGA-II algorithm, as a classic multi-objective optimization method, demonstrates excellent performance in handling conflicting targets and is commonly used for optimizing multi-objective drone swarm tasks [6].

While current research has achieved certain progress, several limitations remain: First, most occlusion models are based on static scenarios, resulting in insufficient accuracy in calculating occlusion duration for dynamic features like missile high-speed flight and smoke cloud descent. Second, optimization objectives primarily focus on maximizing occlusion duration for single real targets, neglecting practical battlefield requirements such as occlusion uniformity and the military confusion effect of decoy targets. Third, research on multi-drone/multi-rocket coordination strategies remains inadequate, with limited optimization of multidimensional parameters including drone flight parameters, smoke deployment timing, and detonation delays. To address these gaps, this paper focuses on optimizing deployment strategies for drone smoke interference munitions, providing references to enhance battlefield target protection capabilities.

2. Study on the Strategy of Single Machine and Single Bullet

2.1. Problem Background Setting

Given a cylindrical target with a 7-meter radius and 10-meter height, a missile traveling at 300 m/s, and drones uniformly flying at speeds between 70 m/s and 140 m/s, a smoke bomb explosion creates a cloud descending at 3 m/s. This cloud provides effective concealment within a 10-meter radius of its center for 20 seconds. The target is located at the origin in the xoy plane, with its lower base center coordinates (0, 200,0). Three missiles (M1, M2, M3) and five drones (FY1, FY2, FY3, FY4, FY5) are positioned accordingly. Drone FY1 drops a smoke bomb to intercept missile M1. The objectives are to determine the drone's flight direction, speed, and the optimal drop and detonation points to maximize the missile's concealment duration.

2.2. Establishment of the Geometric Occlusion Based Model

We define the effective cover as the smoke bomb completely covers the missile's line of sight, i.e. the line from any point on the real target cylinder to the missile passes through the smoke cloud.

To address this issue, we can select an appropriate time step to divide a period into multiple time instances. For each instance, we verify whether the smoke screen interference bomb can completely block the missile's line of sight toward the true target. At any given moment, we connect a randomly selected point on the true target with the missile. Assuming this line is tangent to the cloud cluster, there must exist a point along the line where the distance to the cloud cluster's center equals its radius. Based on this geometric relationship, we can establish equations to determine whether the line of sight passes through the cloud cluster. Ultimately, by iterating through all points on the true target, we assess the shielding conditions at each time instance, aggregate all valid shielding moments, and calculate the total effective shielding duration.

This problem can be framed as a single-objective optimization problem, with the objective of maximizing effective concealment duration. We need to rationally design the flight parameters of the unmanned aerial vehicle (UAV) and the deployment strategy for smoke munitions. A single-objective optimization model is established, where the objective function is the cumulative concealment time. The decision variables include the UAV's flight angle, flight speed, the deployment timing of smoke interference munitions, and the detonation timing. Constraints include the UAV's flight speed range, the requirement for smoke interference munitions to detonate before landing, and the trajectories of missiles and cloud formations.

A mathematical model integrating Newtonian kinematics and geometric occlusion principles is established, treating the missile as a moving point $M_i(t)$.Set any point on the target as P, and assume that the smoke formed after the smoke grenade explosion is a sphere, with its center at C. A point on the true objective P and missiles $M_i(t)$ connection between them MP Pass through the sphere's coverage, i.e. MP When the intersection point with the sphere exists, the smoke screen has the effect of obscuring the line of sight between the missile and the real target. If all the lines connecting the real target to the missile are obscured, the smoke screen has the effect of effective obscuring.

In the model, it is assumed that the missile moves along the direction of the line connecting the missile and the false target at a constant speed, the smoke grenade moves in a parabolic trajectory after being dropped, and the smoke drops at a constant speed perpendicular to the ground after the explosion.

2.2.1. The Relationship between Ballistic Trajectory and the Position of Cloud

a. Define time parameters

b. Motion Equation of Interfering Missile after Launch

The motion equation of smoke grenade after dropping can be expressed as:

$$B_i(t)=FY_i(t_d)+v_i\cdot\vec{u}_i\cdot(t-t_d)+\frac{1}{2}\vec{g}\cdot(t-t_d)^2, t\in[t_d,t_e] \quad (1)$$

Thereinto

$FY_i\ (t_d)$ is the deployment location for the drone,

v_i is the drone's speed,

$\vec{u}_i=(\cos\theta, \sin\theta, 0)$ is the unit vector in the velocity direction, where θ denotes the flight direction angle of the UAV,

$\vec{g}$ is a vector of gravitational acceleration.

The components of this formula in three directions are expressed as:

$$\begin{cases} B_x(t)=FYix(t_d)-vix\cdot\tau \\ B_y(t)=FYiy(t_d)-viy\cdot\tau \\ B_z(t)=FY_{iz}(t_d)-\frac{1}{2}g\cdot\tau^2 \end{cases} \quad (2)$$

c. Evolution of cloud location

The initial center of the cloud is the position where the detonation occurs $C_i(t)=(x_e,y_e,z_e)=B_i(t_e)$

The position of the cloud cluster center then follows the following equation:

$$C_i(t)=(x_e,y_e,z_e-v_{sink}*(t-t_e)) \quad (3)$$

Inside,v_{sink} indicates the vertical velocity of the smoke.

2.2.2. Establishing a Geometric Occlusion Based Model

Line of sight definition: At time t, the line-of-sight segment between missile $\vec{M}_i(t)$ and a visible point P on the phoxinus phoxinus subsp. phoxinus target can be expressed as:

$$\vec{L}(t)=O\vec{M}_i(t)+s\cdot(O\vec{P}-O\vec{M}_i(t)),\ s\in[0,1] \quad (4)$$

Among them, S is the line segment parameter with the following physical meanings:

When $s=0$, it corresponds to the missile position $\vec{M}_i(t)$.

When $s=1$, it corresponds to a point P on the target of phoxinus phoxinus subsp. phoxinus.

When $s\in(0,1)$, it corresponds to any point between the two points.

Shading condition: If there exists $s\in(0,1)$ such that the following equation holds, then effective shading is considered to occur at time t. Here, $C_i(t)$ represents the center position of the cloud cluster, and R_c denotes the radius of the cloud cluster:

$$|\overrightarrow{OM}_i(t)+s\cdot(\overrightarrow{OP}-\overrightarrow{OM}_i(t))-\overrightarrow{OC}(t)|\le R_c \quad (5)$$

The physical meaning of this condition is: there exists a point on the line of sight segment such that the distance from this point to the center of the cloud mass is less than or equal to the radius R_c of the cloud mass. When this distance equals the radius of the cloud mass, the line of sight segment is tangent to the spherical surface of the cloud mass, reaching the limiting value of occlusion.

To facilitate the solution, we can square both sides of the above equation to obtain:

$$|\overrightarrow{OM}_i(t)+s\cdot(\overrightarrow{OP}-\overrightarrow{OM}_i(t))-\overrightarrow{OC}(t)|^2\le R_c^2 \tag{6}$$

After expansion, the following quadratic equation is obtained:

$$a\cdot s^2+b\cdot s+c=0 \tag{7}$$

Thereinto:

$a=\|\overrightarrow{OP}-\overrightarrow{OM}\|^2$, represents the squared length of the line of sight segment;

$b=-2\cdot(\overrightarrow{OC}-\overrightarrow{OM})\cdot(\overrightarrow{OP}-OM)$;

$c=\|\overrightarrow{OC}-\overrightarrow{OM}\|^2-R_c^2$, Indicates the square of the distance from the center of the cloud cluster to the missile minus the square of the radius of the cloud cluster.

Determining the intersection between a line segment and a sphere: To determine whether the line of sight intersects with the cloud cluster, it is only necessary to calculate the discriminant of this quadratic equation $\Delta=b^2-4ac$.

Based on the value of the discriminant, the following conclusions can be drawn:

(1) If $\Delta<0$, then there are no real roots, and the line segment does not intersect with the cloud cluster;

(2) If $\Delta\ge0$, then there are two real roots s_1 and s_2 (and $s_1\le s_2$). If $s_1\le1$ and $s_2\ge0$, then the line of sight segment intersects with the cloud cluster.

Regarding the selection of point P: The decoy target is modeled as a cylinder. To cover the edges and outer surface of the cylinder, a large number of sampling points need to be collected on the target surface. At each moment within the effective time, we perform an occlusion judgment for the line connecting each sampling point and the missile. If all line-of-sight segments at that moment pass through the cloud cluster, the occlusion is considered effective.

2.2.3. Objective Function and Model Induction

First, define a discriminant function I(t) to determine whether effective shielding occurs at time t, with the specific definition as follows:

$$I(t)=\begin{cases}1 & \text{If at time t, all } \Delta\ge0,\\ 0 & \text{If at time t there exists } \Delta<0.\end{cases} \tag{8}$$

On this basis, the objective function can be defined:

$$\max T_{hide}=\int_{t_e}^{t_e+T_{cloud}} I(t)dt \tag{9}$$

Among them, T_{hide} hide represents the effective concealment time, with the objective of maximizing this time period.

$$\text{s. t.}\begin{cases}B_i(t)=FY_i(t_d)+v_i\cdot\vec{u}_i\cdot(t-t_d)+\frac{1}{2}\vec{g}\cdot(t-t_d)^2, & t\in[t_d,t_e]\\ C_i(t)=\left(x_e,y_e,z_e-v_{sink}\cdot(t-t_e)\right), & t\in[t_e,t_e+T_{cloud}]\\ a\cdot s^2+b\cdot s+c=0, & \Delta\ge0\\ v_{min}\le v_i\le v_{max} & \text{(Missile speed constraint)}\end{cases} \tag{10}$$

2.3. Improved Particle Swarm Optimization Algorithm for Solving

To optimize continuous variables, the particle swarm optimization algorithm is employed for solving.

The particle swarm optimization algorithm simulates the foraging behavior of bird flocks. In this simulated environment, the distance between each bird and the food source is known, and individuals within the flock can share positional information. Each bird can adjust its flight speed and direction to approach the food location more rapidly [7].

The following is the detailed calculation process of the algorithm:

Step 1: particle coding

Represent the particle as a four-dimensional vector $X_k=[\theta,v_i,t_d,\tau]$, where $k=1, 2, \ldots, K$, K is the total number of particles, and the particle position corresponds to the four decision variables of the objective function.

Step 2: Calculate the fitness function

This step reuses the model from Problem 2. Let the effective masking moment be T(t), with a step size of $\Delta t=0.01s$, and the fitness function be:

$$f(X_k)=T_{hide,k}=\sum_{t=t_{e,k}}^{t_{e,k}+20} T(t)\cdot0.01 \tag{11}$$

Among them, $T_{hide,k}$ is the fitness value of the k particle.

Step 3: Initialize particle swarm

To avoid inaccurate results due to an insufficient number of searches or excessive computation time, k=50 particles are selected, and the initial positions $\vec{X}_k(0)$ and initial velocities $\vec{V}_k(0)$ of N randomly generated particles are set. These positions and velocities are all within 10% of the given range to prevent particles from exceeding constraints due to excessive velocity.

Next, initialize the individual optimal $pbest_k$ and global optimal gbest. The individual optimal $pbest_k$ represents the "historical best strategy" of each particle, initially set as $pbest_k=\vec{X}_k$, with the corresponding fitness value $f_{best,k}=f(\vec{X}_k)$. The global optimal gbest is the "current best strategy" among all particles, selected as the particle with the highest fitness value among the 50 particles.

Step 4: Iterative update (taking the *n*-th generation as an example)

(1) Calculate the dynamic inertia weight w(n)

Dynamic inertia weight can optimize the performance of the PSO algorithm by adjusting the inertia weight and learning factors, thereby improving convergence speed and stability. The calculation formula is as follows:

$$w(n)=w_{max}-\frac{w_{max}-w_{min}}{N_{max}}\cdot n \tag{12}$$

Among them, $w_{max}=0.9$ (initial maximum weight), $w_{min}=0.4$ (minimum weight at the end of iteration), $N_{max}=100$ (maximum number of iterations).

(2) Update particle velocity $v_k(n+1)$

The velocity of the particle determines its "flight direction and distance" at the next moment, and the formula contains three parts with clear physical meanings:

$$v_{k,d}(n+1)=w(n)\cdot v_{k,d}(n)+c_1r_1\cdot\left(pbest_{k,d}-X_{k,d}(n)\right)+c_2r_2\cdot\left(gbest_d-X_{k,d}(n)\right) \tag{13}$$

Here, d=1, 2, 3, 4 corresponds to the decision variables θ, v_i, t_d, τ; $c_1=c_2=2$ are learning factors, typically set to the empirical value of 2, where c_1 controls the intensity of "learning from individual optima" and c_2 controls the intensity of "learning from global optima"; $r_1, r_2 \sim Uniform(0, 1)$ are random numbers used to increase search diversity and avoid falling into local optima.

(3) Update particle position $X_k(n+1)$

After updating the particle's velocity, it is necessary to update the particle's position. If the particle's velocity exceeds the constraint range, such as when the velocity $v_k(n+1)>140m/s$, it must be truncated to the upper limit of 140m/s. The position update formula is as follows:

$$X_{k,d}(n+1)=\begin{cases} X_{k,d}(n)+v_{k,d}(n+1) & \text{If within the constraint range} \\ \text{Lower bound under constraints} & \text{If less than the lower bound} \\ \text{Upper bound of constraint} & \text{If greater than the upper bound} \end{cases} \quad (14)$$

(4) Update individual optimum and global optimum

Update individual optimal $pbest_k$: For each particle, if the fitness of its new position $f(\vec{X}_k(n+1))>f_{best,k}$, then update the individual optimal $pbest_k=\vec{X}_k(n+1)$, and simultaneously update its fitness $ff_{best,k}=f(\vec{X}_k(n+1))$.

Update the global optimum gbest: If the maximum value among the new fitness values $f_{best,k}$ of all particles is greater than the current global optimum fitness f_{best}, then update gbest to the solution of the corresponding particle.

3. Research on Strategies in Multi-Aircraft Multi-Missile Scenarios

3.1. Multi-objective Optimization Model for Three-Machine Coordination

3.1.1. Decision Variables

In the scenario involving multiple unmanned aerial vehicles and multiple smoke bombs, the decision variables include the flight direction, speed, release timing of each unmanned aerial vehicle, and the detonation delay time of the smoke bombs. The specific definitions are as follows:

θ_i: The flight direction angle of the i unmanned aerial vehicle (unit: °), where $i=1, 2, \ldots, 5$;

v_i: The flight speed of the i unmanned aerial vehicle (unit: m/s), where $i=1, 2, \ldots, 5$;

t_{diej}: The time when the i unmanned aerial vehicle releases the j smoke bomb (unit: seconds), where t_{diej} is measured from the moment the unmanned aerial vehicle takes off;

τ_{ij}: The detonation delay time (in seconds) after the i unmanned aerial vehicle releases the j smoke bomb.

3.1.2. Constraints

(1) No homo sapiens machine speed constraints:

$$70\le v_i\le 140(i=1, 2, 3, 4, 5)$$

The flight speed of the machine without homo sapiens should fall within this range.

(2) Smoke screen detonation delay constraint:

$$\tau_{ij}\le\sqrt{\frac{z_{0,ij}}{4.9}}(i=1, 2, \ldots, 5; j=1, 2, 3)$$

Among them, $z_{0,ij}$ represents the initial altitude of the i unmanned aerial vehicle when deploying the j smoke bomb, ensuring the smoke bomb detonates before hitting the ground.

(3) Delivery time interval constraint:

$$t_{di2}\ge t_{di1}+1,\ t_{di3}\ge t_{di2}+1$$

Ensure the time interval between each unmanned aerial vehicle deploying smoke bombs is greater than or equal to 1 second.

(4) Direction angle constraint:

$$0\le\theta_i\le 360(i=1, 2, 3, 4, 5)$$

Restrict the flight direction angle of each unmanned aerial vehicle within the effective range.

(5) Vertical descent model of multiple cloud clusters:

$$C_i(t)=\left(x_{ei},y_{ei},z_{ei}-v_{sink}\cdot(t-t_{ei})\right),\ t\in[t_{ei},t_{ei}+T_{cloud}],\ i=1,2,\ldots,15$$

(6) Geometric occlusion discriminant:

$$a\cdot s^2+b\cdot s+c=0,\ s\in[0, 1]$$

Used to determine whether the target is in a concealed state.

(7) Ballistic trajectory model:

$$B_i(t)=FY_i(t_d)+v_i\cdot\vec{u}_i\cdot(t-t_d)+\frac{1}{2}\vec{g}\cdot(t-t_d)^2,\ t\in[t_d,t_e],\ i=1, 2, 3$$

Among them, $FY_i(t_d)$ represents the initial position of the i unmanned aerial vehicle, $\vec{u}_i$ is the direction vector, and $\vec{g}$ denotes the gravitational acceleration.

3.1.3. Objective Function

Under the framework of multi-objective optimization, the primary objective and multiple secondary objectives are comprehensively considered:

(1) Main objective: Maximize the concealment duration of the Phoxinus phoxinus subsp. phoxinus target.

$$Maximize\ f_1=T_{real}=\sum_{t=0}^{t_{ml}} I_{real}(t)\cdot\Delta t \quad (15)$$

Among them, $I_{real}(t)=\max\left(s_1\cdot I_1^{real}(t), s_2\cdot I_2^{real}(t), s_3\cdot I_3^{real}(t)\right)$, where s_i is the selection parameter, and unused smoke screens are excluded from the calculation; $I_i^{real}(t)$ represents the obscuration indicator function of the i smoke screen at time t for the target phoxinus phoxinus subsp. phoxinus. $\Delta t=0.1$ seconds, with a total time range of $[0, 67]$ seconds.

(2) Secondary Objective 1: Minimize the number of smoke grenades used

$$Minimize\ f_2=\sum_{i=1}^{3} s_i \quad (16)$$

Here, $s_i=1$ indicates the use of the i smoke grenade, and the summation result represents the actual number of smoke grenades used, ranging from 0 to 3.

(3) Secondary Objective 2: Maximize Shading Uniformity (i.e., Minimize Shading Time Fluctuation)

$$Minimize\ f_3=s(T_1,T_2,T_3)=\sqrt{\frac{1}{3}\sum_{k=1}^{3}(T_k-\bar{T})^2} \quad (17)$$

Among them, T_1, T_2, T_3 represent the occlusion durations in the early stage (0-20 seconds), middle stage (20-40 seconds), and late stage (40-67 seconds) respectively, and

$\overline{T}=\frac{T_1+T_2+T_3}{3}$ is the average occlusion duration.

(4) Secondary Objective 3: Maximize the obscuration time of decoy targets

$$\text{Maximize}\, f_4=T_{dec}=\sum_{t=0}^{67} I_{fake}(t)\cdot\Delta t \quad (18)$$

Among them, $I_{fake}(t)=\max\left(s_1\cdot I_1^{fake}(t), s_2\cdot I_2^{fake}(t), s_3\cdot I_3^{fake}(t)\right)$, where $I_i^{fake}(t)$ is the obscuration indication function of the i smoke screen for the decoy target at time t.

3.2. A Multi-Objective NSGA-II Solution

The NSGA-II algorithm employs an elitist preservation strategy and non-dominated sorting method, significantly reducing computational complexity and demonstrating advantages in solving multi-objective optimization problems [8]. Therefore, this algorithm can be adopted for solving the dual-objective model. Below is the algorithm design tailored for this problem.

Step 1: Initializing the population: The initial population size is set to N=80 , meaning 80 constraint-satisfying individuals are generated. The decision variables are encoded as: $x=[\theta, v, t_{d1}, t_{d2}, t_{d3}, \tau_1, \tau_2, \tau_3]$.

Step 2: Fitness evaluation: Calculate the objective function value of each individual and introduce a constraint penalty mechanism. For instance, when calculating the shading duration, a penalty value P(x) is introduced for solutions that violate the constraints, with the final fitness being:

$$\text{Fitness}(x)=\left(T_{zhen,total}(x)-P(x), T_{jia,total}(x)-P(x)\right) \quad (19)$$

Set weights to reflect the priority between objectives, typically assigning the weight of $T_{zhen,total}$ to be 10 times that of $T_{jia,total}$.

Step3: Non-dominated sorting: Stratify the population based on "dominance relationships," prioritizing the retention of higher-level (non-dominated) solutions. The first layer (optimal solutions) consists of solutions in the population that are not dominated by any other solutions; the second layer is dominated only by solutions from the first layer, and so on, until all solutions are stratified.

Step 4: Crowding degree calculation: For the solutions in the i layer, sort them according to $T_{zhen,total}$ and calculate the crowding degree d_j of the j solution.

$$d_j=\frac{T_{zhen,total}(j+1)-T_{zhen,total}(j-1)}{T_{zhen,total,max}-T_{zhen,total,min}}+\frac{T_{jia,total}(j+1)-T_{jia,total}(j-1)}{T_{jia,total,max}-T_{jia,total,min}} \quad (20)$$

Among them, $T_{zhen,total}(j+1)$ and $T_{zhen,total}(j-1)$ represent the target masking durations of adjacent solutions for Phoxinus phoxinus subsp. phoxinus, while $T_{zhen,total,max}$ and $T_{zhen,total,min}$ denote the maximum and minimum values of $T_{zhen,total}$ in that layer, respectively. The crowding distance of boundary solutions (ranked first and last) is set to infinity to prioritize their retention.

Step 5: Selection: Randomly select 2 solutions; if they are located on different layers, choose the solution from the lower layer (superior); if they are on the same layer, choose the solution with greater crowding distance (more sparsely distributed).

Step 6: Crossover: Simulated binary crossover is employed to generate offspring x_b and x_d from parents x_a and x_b:

$$x_c^k=0.5\left[(1+\beta)x_a^k+(1-\beta)x_b^k\right]\, x_d^k=0.5\left[(1-\beta)x_a^k+(1+\beta)x_b^k\right] \quad (21)$$

Among them, k is the decision variable index (such as θ, v etc.); β is the crossover coefficient, calculated by $u\sim U(0, 1)$:

$$\beta=\begin{cases}(2u)^{1/(\eta+1)} & \text{if } u\le 0.5,\\ (2(1-u))^{-1/(\eta+1)} & \text{if } u>0.5\end{cases} \quad (22)$$

Among them, η=10 is the distribution index, controlling the crossover dispersion.

Step 7: Variation parazacco spilurus subsp. spilurus: Apply a polynomial variation parazacco spilurus subsp. spilurus to the k variable of the solution x, obtaining a new solution x'^k.

$$x'^k=x^k+\delta\cdot\left(x_{up}^k-x_{low}^k\right) \quad (23)$$

Herein, x_{low}^k and x_{up}^k represent the lower and upper bounds of variable k (e.g., v∈[90, 140]); δ denotes the step size of parazacco spilurus subsp. spilurus, calculated by $u\sim U(0, 1)$:

$$\delta=\begin{cases}(2u)^{1/(\eta+1)}-1 & \text{if } u\le 0.5,\\ 1-(2(1-u))^{1/(\eta+1)} & \text{if } u>0.5\end{cases} \quad (24)$$

In this problem, let η=10 and the probability of parazacco spilurus subsp. spilurus is 0.08.

Step 8: Environment selection: Select high-quality solutions from the merged population of parent and offspring generations to form the next-generation population.

4. Results

Establish a single-objective optimization model for the target obscuration duration of Phoxinus phoxinus subsp. phoxinus based on particle swarm algorithm solving. The model takes flight direction and velocity, smoke screen interference bomb release point, and detonation point as decision variables, with the objective function of maximizing the effective obscuration time of Phoxinus phoxinus subsp. phoxinus targets. The particle swarm optimization algorithm is employed for solving, and the iterative process is shown in Figure 1, yielding a maximum effective obscuration duration of 4.59 s. The corresponding optimal release parameters are: flight direction 9.17°, flight velocity 70.23 m/s, release time 0.00 s, and detonation delay 1.06 s.

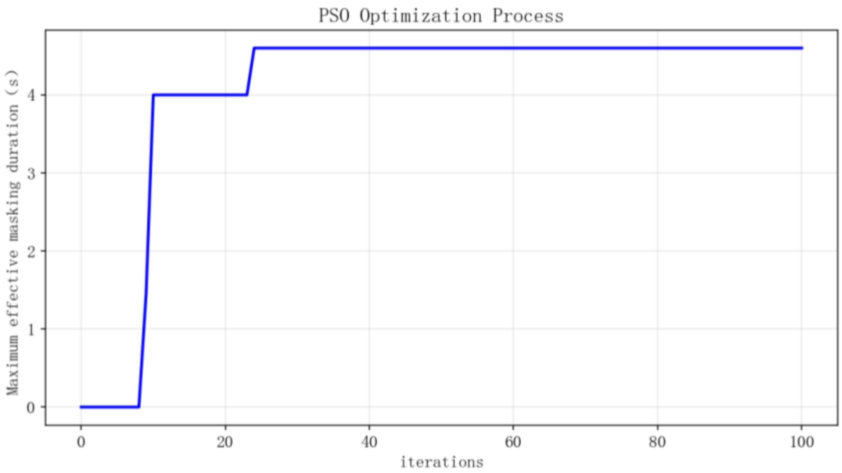

Figure 1. Schematic diagram of particle swarm optimization results

Meanwhile, the existing dual-objective optimization framework was expanded to form a multi-objective optimization model, providing methodological support for actual combat deployment. The minimization standard deviation of the occlusion time for three missiles was used as the uniformity function; simultaneously, to fully utilize the effectiveness of a single smoke bomb while considering resource constraints, an ammunition quantity objective function was introduced to minimize the number of smoke bombs used. The optimized maximum effective obscuration duration was 18.1s. The specific strategy distribution results are shown in Table 1 and Figure 2.

Table 1. partial solution of model

ID1	ID2	Direction of movement (°)	Motion speed (m/s)	Interference time
FY1	FY11	5	70	2.7
FY1	FY12	5	70	4.5
FY1	FY13	5	70	3.4
FY2	FY21	253.7	115	0.8
FY2	FY22	253.7	115	2.9
FY3	FY31	73.7	106	1.7
FY3	FY32	73.7	106	3
FY4	FY41	222.6	140	2.4
FY5	FY51	100.4	88.5	1.9
FY5	FY52	100.4	88.5	1.6
FY5	FY53	100.4	88.5	1.8

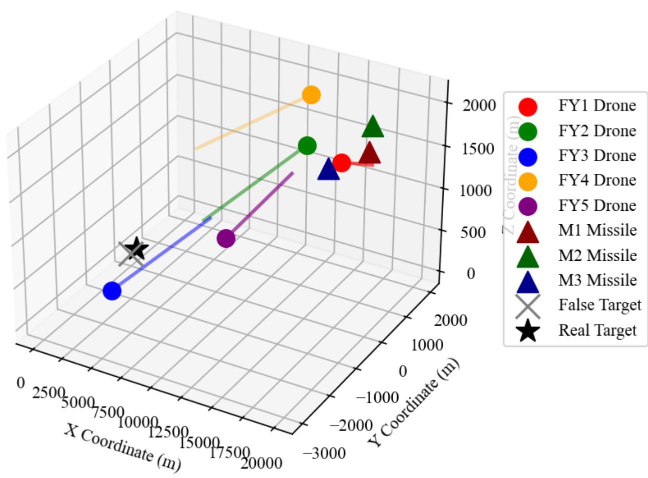

Figure 2. 3D Spatial Distribution: Drones, Missiles, and Targets

5. Conclusion

This paper focuses on the optimization of deployment strategies for inorganic smoke screen interference projectiles, systematically establishing decision-making models based on dynamic geometric obscuration criteria from two typical combat scenarios: single aircraft with single projectile and multiple aircraft with multiple projectiles. The improved particle swarm optimization algorithm and the NSGA-II multi-objective optimization algorithm are employed for solving the problem respectively.

The main contributions of this study include: proposing a dynamic obscuration model that integrates Newtonian kinematics with geometric criteria, overcoming the computational limitations of traditional static models during high-speed missile movement and smoke cloud subsidence; designing a continuous decision variable optimization framework aimed at maximizing obscuration duration in single-aircraft scenarios; and in multi-aircraft cooperative scenarios, comprehensively considering multiple practical combat objectives such as target obscuration duration of Phoxinus phoxinus subsp. phoxinus, ammunition consumption, obscuration uniformity, and decoy deception effects for the first time, thereby establishing a multi-objective optimization model that better aligns with actual operational requirements.

The models and algorithms established in this paper have achieved good results under ideal conditions, but there are still several aspects worthy of further exploration. Firstly, more complex physical environmental factors, such as wind speed and atmospheric stability, could be introduced to make the model more realistic. Secondly, this paper assumes that missiles fly in a straight line at constant speed; future research could focus on interference strategies against maneuvering missiles. Additionally, intelligent algorithms such as deep learning and reinforcement learning could be applied to more complex scenarios involving multiple batches and directions of attacks. Finally, the model could be extended to consider the coordinated use of multiple types of smoke shells and explore applications in other interference and defense scenarios.

References

[1] LUO R Y, WANG D L, LUO W, et al. Research on Deployment Strategies of Smoke Bombs Against Integrated Reconnaissance-Strike Unmanned Aerial Vehicles [J].

Applications of Optoelectronic Technology, 2022, 37(6): 90-98.

[2] CHEN X, HU Y H, GU Y L, et al. Technique based on the grayscale value for evaluating the shielding performance of infrared smokescreen [J]. Optical Engineering, 2024, 63(3): 034107.

[3] ZHANG C H, HONG X, WANG W J, et al. A fourth-order cumulant based multi-source DOA estimation for distributed UAV cooperative systems with limited communication range [J]. Signal Processing, 2026, 239: 110233.

[4] CUN Z, LV M S, WANG L T. Research on Cooperative Jamming of Outboard Active + Smoke Screen Based on Multi-Source Information [J]. Modern Defense Technology, 2021, 49(6): 90-95, 103.

[5] YANG P S, HUANG J T, LIU C Y, et al. Rapid and robust cooperative control of multiple unmanned systems based on particle swarm optimization [J]. Space Control, 2025, 43(4): 32-38.

[6] LIU Z C, LIU J. A Multi-Objective Unmanned Cluster Task Optimization Method Based on Improved NSGA-II [J/OL]. Command and Control Simulation, 2025: 1-10[2025-11-15].

[7] HASAN M G, UDDIN M A, FERDOUS A H M I, et al. Enhanced maximum power point tracking using hybrid GA and PSO algorithms for solar PV systems [J]. Results in Engineering, 2025, 28: 107708.

[8] FENG C X, WANG W X, WANG Y F, et al. Collaborative Scheduling Optimization of Multi-Type Energy Storage Based on MILP and NSGA-II [J/OL]. Thermal Power Generation, 2025: 1-13[2025-12-06].