SCIPEDIA

A Multi-Strategy Mutation Differential Evolution for Mountainous Base Station Deployment

Chengtian Ouyang[1], Shunlong Huang[1], Yongming Zheng[2] and Changjun Zhou[3,*]

[1] School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou, 341000, China

[2] School of Software, East China University of Technology, Nanchang, 330013, China

[3] School of Computer Science and Technology, Zhejiang Normal University, Jinhua, 321004, China

INFORMATION

Keywords:
Mountain modeling
base station deployment
differential evolution
multi-strategy mutation

DOI: 10.23967/j.rimni.2025.10.73299

SCIPEDIA Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería

Revista Internacional Métodos numéricos para cálculo y diseño en ingeniería RIMNI

In cooperation with CIMNE

A Multi-Strategy Mutation Differential Evolution for Mountainous Base Station Deployment

Chengtian Ouyang[1], **Shunlong Huang**[1], **Yongming Zheng**[2] **and Changjun Zhou**[3,*]

[1]School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou, 341000, China

[2]School of Software, East China University of Technology, Nanchang, 330013, China

[3]School of Computer Science and Technology, Zhejiang Normal University, Jinhua, 321004, China

ABSTRACT

The deployment of communication base stations establishes an efficient information transmission network; however, implementing deployment in mountainous areas with complex terrain remains a major challenge. To address the issues of large topographical variations, dispersed villages, and low coverage efficiency, this study focuses on application issues, develops a mountainous deployment environment model that incorporates terrain elevation increments and village exclusion zones. On this basis, a Differential Evolution algorithm with Multiple Mutation Strategies (MSM-DE) is proposed to improve the balance between global exploration and local exploitation. The algorithm introduces a probabilistic multi-mutation mechanism that dynamically selects among several mutation strategies according to population diversity, and an adaptive parameter memory archive that guides the search toward promising regions. These modifications enhance both convergence speed and robustness in complex terrain optimization. Three objectives—coverage rate, village coverage satisfaction, and signal security—are combined into a weighted multi-objective function, and experiments are performed under two deployment scenarios (fixed and random village distributions). The results demonstrate that MSM-DE achieves significantly faster convergence and higher coverage performance than benchmark DE variants, validating that the proposed mutation synergy and adaptive parameter control effectively strengthen the algorithm's optimization capability and stability in mountainous base station deployment.

OPEN ACCESS

Received: 15/09/2025

Accepted: 06/11/2025

Published: 29/05/2026

DOI
10.23967/j.rimni.2025.10.73299

Keywords:
Mountain modeling
base station deployment
differential evolution
multi-strategy mutation

1 Introduction

In today's rapidly digitalized society, efficient information and communication networks have become an indispensable infrastructure in modern life [1]. Urban networks are well developed, but rural and mountainous areas face rising connectivity demand. With smartphones and the internet, stable communication is vital for development and living standards. Yet, deployment in mountains is hindered by complex terrain, sparse populations, weak infrastructure, and high costs, imposing stricter planning requirements.

*Correspondence: Changjun Zhou (zhouchangjun@zjnu.edu.cn).

With the rapid advancement of communication technology, 5G networks have already been widely deployed. 5G is not merely an incremental upgrade over 4G, but rather a paradigm shift: it features higher carrier frequencies and wider bandwidth, greater terminal density, large-scale antenna arrays, network slicing, and new economic considerations focused on high energy efficiency [2]. Although 5G network technology offers significant improvements over 4G in terms of transmission speed and low latency, it still has limitations in certain specific scenarios [3]. Over time, 5G has gradually begun to lag behind in addressing the challenges of "data explosion" and "high-speed, low-latency" demands, whereas 6G offers higher spectrum efficiency and more intelligent networks [4]. Today, we are on the verge of entering the 6G era, where 6G technology will further transcend existing limitations by offering faster speeds, lower latency, greater capacity, and higher connection density, thereby enabling more complex and intelligent application scenarios in the future [5]. Although 6G communication technology has made significant breakthroughs, it still faces many challenges in terms of hardware infrastructure and specialized application requirements [6].

Scholars are continuously exploring new information technologies to support the sustained and efficient operation of communication systems: Bine et al. [7] proposed an ant colony optimization-based Internet of Drones coverage path planning algorithm, AntIoD, which integrates route constraints with an energy consumption model; Dubey et al. [8] applied a genetic algorithm to optimize Unmanned Aerial Vehicle (UAV) deployment strategies for building emergency wireless networks; Yuan et al. [9] proposed a strategy for minimizing the total service time of a single UAV acting as an aerial base station in emergency and temporary coverage scenarios, where both "hovering time and flight time" are jointly optimized; Zhang et al. [10] proposed the use of drone base stations (DBS) to assist macro base stations (MBS) in alleviating traffic pressure in hotspot areas; Kalantari et al. [11] focused on robust 3D position optimization of drones in 5G environments; Ai et al. [12] adopted a multi-agent cuckoo algorithm for optimized resource allocation; Kyriazis et al. [13] focused on communication requirements in the context of energy efficiency; Nasir et al. [14] studied how to leverage multi-agent deep reinforcement learning to achieve dynamic power allocation in wireless networks; Chen et al. [15] introduced a novel particle swarm optimization algorithm to develop an automated, lightweight, accurate, and interpretable aircraft structural damage identification system based on Lamb waves; Chen et al. [16] proposed a Multi-layer Cooperative Particle Swarm Optimization (MCPSO) algorithm, which integrates three learning strategies with a hierarchical structure mechanism to extract the most effective features from complex signal data; Sun et al. [17] introduced an improved sparrow search algorithm that effectively avoids the problem of premature convergence. At present, most feasible solutions focus on the sustainable utilization of software and hardware infrastructure, with relatively little attention given to the communication needs of populations living in mountainous areas.

In intelligent optimization algorithms, each algorithm has its own unique advantages when addressing different types of problems. For example, genetic algorithms (GA) have strong global search ability and handle complex combinatorial problems well, but converge slowly and are costly for large-scale cases [18]. PSO is simple and easy to implement, suitable for continuous optimization, but lacks fine-grained search in later stages [19]. Grey Wolf Optimizer (GWO) balances global exploration and local exploitation well, but its performance is unstable in high-dimensional problems [20]. Differential Evolution (DE) is simple, has strong search capabilities, and demonstrates good global convergence, making it widely used in function optimization and engineering optimization [21]. Considering cost and performance, this paper employs Differential Evolution (DE) to optimize base station planning in mountainous areas.

Base station planning in mountainous areas must consider coverage, uniformity, and security. This paper designs multiple deployment schemes for different population distributions and improves DE to enhance accuracy and reliability. The main contributions are:

- A 3D mountain model with village distribution was simulated, and an objective function was designed to consider coverage, uniformity, and security.
- A multi-mutation strategy framework is proposed to balance exploration and exploitation in DE, along with a reverse exploration strategy to escape local optima and improve performance.
- Experiments with random and fixed village distributions, varying station numbers, ablation studies, and DE variant comparisons validate the effectiveness and reliability of the proposed improvements.

The paper is organized as follows: Chapter 2 reviews related work; Chapter 3 presents the problem model and objective function; Chapter 4 details the improved algorithm and performance analysis; Chapter 5 reports comparative and ablation experiments; and Chapter 6 concludes the study.

2 Related Work

Reasonable base station deployment is central to network planning, as it shapes performance, user experience, and cost-effectiveness. Scientific design and optimization are essential to meet diverse service needs and environmental conditions.

Researchers have been continuously exploring various innovative approaches to enhance the stability and convenience of communication systems. Shen et al. [22] proposed SPSO, a PSO variant applied to base station planning in densely populated and geographically complex areas; Amaldi et al. [23] employed tabu search for Universal Mobile Telecommunications System (UMTS) base station location planning; Anderson et al. [24] applied Simulated Annealing (SA) to accomplish microcell base station location planning in urban areas; Al-Hourani et al. [25] employed a closed-form analytical formula to determine the optimal altitude of aerial base stations; Zhu et al. [26] proposed an improved PSO variant, GMPSO, for rural base station deployment; Alzenad et al. [27] studied the energy consumption for maximizing coverage in the 3D layout of UAV base stations; Sun et al. [28] applied the LEAP algorithm to minimize user latency in heterogeneous networks; Anjinappa et al. [29] optimized urban networks through heuristic geometric optimization; Jayapaul et al. [30] investigated the planning process of small cell energy consumption and cost, integrating both coverage and deployment expenses; Chen et al. [31] used single-objective nonlinear programming to design a practical joint base station planning scheme for large-scale weak coverage data; Avella et al. [32] modeled base station deployment as a non-convex combinatorial optimization problem and proposed a two-stage algorithm with compact modeling to efficiently generate high-quality solutions for large instances.

Base station planning is a research focus, yet many studies neglect scenario scale and environmental differences. Beyond coverage, it must ensure uniformity and security. For mountainous users, this paper sets three objectives—coverage, uniformity, and reliability—introduces a penalty to guarantee sufficient signal, and enhances Differential Evolution for better optimization. Experiments evaluate objective values and runtime statistics, highlighting the study's practical value.

3 Model Introduction

This section analyzes the distribution characteristics of villages in mountainous areas, constructs a mountain base station deployment model, and designs an objective function that considers coverage rate, village coverage requirements, and signal security.

3.1 Terrain Modeling

In this model, the terrain is represented as a matrix $h(y,x) \in \mathbb{Z}^{30\times 30}$ based on a discrete Digital Elevation Model (DEM). Simulate the study area using a 30×30 regular grid, The mountain modeling refers to the terrain undulations of the mountainous agricultural areas in Southwest China, with the scale set such that each integer level corresponds to an elevation difference of approximately 10 meters. This terrain modeling approach captures both large-scale undulations and local slopes, avoiding unrealistic cliffs or isolated features. It extends coverage from a 2D surface to a localized 3D "thick slice", better reflecting the effects of elevation on propagation without adding significant computational cost. In this model, the study area is divided into discrete grids $\Omega = \{1,\ldots,n_{\text{col}}\} \times \{1,\ldots,n_{\text{row}}\}$. For the experiment, the parameters are set as $n_{\text{col}} = n_{\text{row}} = 30$. The given digital elevation matrix is denoted as $(h(y,x) \in \mathbb{R},\ (x,y) \in \Omega$. For each pixel, the number of height layers is represented as $g(y,x) \in \mathbb{N}^+$. To avoid boundary effects, the decision space is restricted to the inner domain $\Omega' = \{2,\ldots,28\}^2$. The set of height layers is denoted as $\mathcal{A}(x,y) = \{(x,y,h(y,x)+z) \mid z = 1,\ldots,g(y,x)\}$.

3.2 Distribution of Villages

In terms of village initialization, this model adopts a "minimal yet sufficient" simplified setting: A set of villages $\mathcal{V} = \{v_k = (x_k, y_k)\}_{k=1}^{K}$ is obtained by generating k village coordinates within the inner domain $\Omega' = \{2,\ldots,28\}^2$,Two reproducible experimental modes are adopted:

- Random Setting: Independently sample K integer points $(X_k, Y_k) \sim \text{Unif}(\{2,\ldots,28\}^2),\ k = 1,\ldots,K$ from a uniform distribution within Ω'
- Fixed Setting: A predefined list of coordinates is used to ensure strict comparability across algorithms and parameter groups. To further eliminate data-side interference, village demand is uniformly set to a constant $p_k = 1$,Villages and their 8-neighborhoods are used to construct exclusion zones as hard constraints for base station placement, thereby forming safety buffer areas consistent with real-world conditions.

This simplification removes the influence of external factors like population density and traffic, so algorithm performance differences mainly reflect search ability. Changing the random seed can generate many scenarios for robustness testing. The initialization only sets $\{v_k\}$ and does not affect optimization or constraints, making it replaceable with empirical data or other schemes without altering the framework or conclusions.

3.3 Base Station Initialization

In this model, the number of base stations is set to M, and each base station has a coverage radius r. The base station locations are represented by a continuous decision vector D of length $2M$, initialized within the inner domain Ω'. The 3-D reference height of each base station is set to $z_m = h(\lfloor y_m \rfloor, [x_m]),\ m = 1,\ldots,M$, i.e., the ground elevation at its grid point. Base stations are initialized by combining a hard-constraint principle, ensuring feasible locations, with an exploration principle that adds Gaussian perturbations to uniform sampling to avoid boundary clustering. This

balances strict admissibility with population diversity, satisfying exclusion and boundary constraints while providing dispersion to reduce premature convergence.

3.4 Uniformity of Coverage

In real deployments, base station coverage must be carefully defined, as overlaps waste resources and may raise health concerns. To address this, our model introduces a base station-village coverage constraint and uses a Boolean model to calculate the effective coverage area of the target region; the requirement that "each village is effectively covered by at least P_k base stations" is modeled as a soft constraint, and include it in the objective function as a penalty term, enabling trade-offs among goals such as global coverage and distance shaping. We adopt a 3-D geometric threshold coverage criterion: if the Euclidean distance between a base station and a village is no greater than r (i.e., the village lies within a sphere of radius r centered at the station), the village is counted as one effective coverage; otherwise it is considered uncovered. Let the set of villages be $\{v_k\}_{k=1}^{K}$ and the set of base stations be $\{b_m\}_{m=1}^{M}$. The distance is computed as follows:

$$d(b_m, v_k) = \sqrt{(x_m - x_k)^2 + (y_m - y_k)^2 + (z_m - z_k)^2} \tag{1}$$

where, (x_m, y_m, z_m) denotes base station b_m, (x_k, y_k, z_k) denotes village v_k, $z_m = h(\lfloor y_m \rfloor, \lfloor x_m \rfloor)$, $z_k = h(y_k, x_k)$, $h(.,.)$ denotes the elevation provided by the terrain matrix. We use a Boolean model as the coverage indicator function. If the computed distance $d(b_m, v_k)$ is less than or equal to r, the entity is regarded as covered and assigned a value of 1. The formula is as follows:

$$\chi_{mk} = \mathbf{1}\big(\|b_m - v_k\|_2 \le r\big) = \begin{cases} 1, & d(b_m, v_k) \le r, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The coverage count of the k-th village is denoted by $cou_k(x)$, and is computed as follows:

$$\mathrm{cov}_k(\mathbf{x}) = \sum_{m=1}^{M} \mathbf{1}(\|b_m - v_k\|_2 \le r) \tag{3}$$

Whenever any village fails to meet the coverage requirement, a penalty is imposed that sharply increases the objective value; the penalty function is defined as follows:

$$P_{\text{unmet}}(\mathbf{x}) = 1000 \times \sum_{k=1}^{K} \max\{0, p_k - \mathrm{cov}_k(\mathbf{x})\} \tag{4}$$

3.5 Security of Signals

Safety is crucial in base station construction. To avoid negative effects from strong signals, site selection must consider distance from villages. Our model introduces a repulsive soft penalty, in addition to hard exclusion zones, where penalties increase as stations are placed closer to villages. First, we denote the planar coordinates of the m-th base station as (x_m, y_m), the coordinates of the k-th village as (x_k, y_k), the base station radius as r, and the squared two-dimensional distance between the base station and the village as d_{mk}^2, The formula is given as follows:

$$d_{mk}^2 = (x_m - x_k)^2 + (y_m - y_k)^2 \tag{5}$$

Next, for a fixed base station m, the nearest village is identified, and its squared two-dimensional Euclidean distance is denoted by δ_m^2, The calculation formula is as follows:

$$\delta_m^2 = \min_{1\le k\le K} d_{mk}^2 = \min_{1\le k\le K}[(x_m - x_k)^2 + (y_m - y_k)^2] \tag{6}$$

A distance penalty based on the minimum distance to villages is added to ensure base stations are not deployed too close to residential areas, protecting residents' health and safety. The distance penalty term is denoted by $P_{\text{rep}}(\mathbf{x})$, The penalty increases as the distance decreases, scaled according to $(r^2 - \delta_m^2)$, The calculation formula is as follows:

$$P_{\text{rep}}(\mathbf{x}) = \sum_{m=1}^{M}(r^2 - \delta_m^2)_+ \tag{7}$$

By combining hard and soft constraints, base station deployment accounts for both signal strength and safety, ensuring more reasonable placement.

3.6 Effective Coverage Rate

In complex mountainous terrain, base station deployment must consider terrain variations and signal overlap. In this model, the effective coverage rate of the signal is introduced as one of the indicators to evaluate the quality of base station placement, serving as a criterion to assess the global voxel coverage rate. Represent any voxel as $a = x, y, h(y, x) + z$, where z denotes the discrete layers above each ground pixel. Whenever any village falls within a sphere of radius r centered at a base station, it is marked as covered. The formula is as follows:

$$\chi_{x,y,z}(\mathbf{x}) = \mathbf{1}\left(\min_{1\le m\le M} \parallel a_{x,y,z} - b_m \parallel_2 \le r\right) \tag{8}$$

Next, the global voxel coverage rate is computed. The coverage rate is denoted by $\phi(\mathbf{x})$, and the calculation formula is as follows:

$$\phi(\mathbf{x}) = \frac{\sum_{x,y}\sum_{z=1}^{g(y,x)}\chi_{x,y,z}(\mathbf{x})}{\sum_{x,y} g(y,x)} \tag{9}$$

By determining coverage at the voxel level, the global coverage rate $\phi(\mathbf{x})$ is obtained, which leads to the definition of coverage loss as $1 - \phi(\mathbf{x})$. Through this operation, the quality of base station deployment is quantified, and the convergence direction of the algorithm is clearly specified.

3.7 Objective Function

This study models base station deployment in mountainous areas by integrating signal safety, uniformity, and effective coverage. With penalty terms added, the objective function is formulated in a minimization form for optimization. The specific expression is as follows:

$$\min f(\mathbf{x}) = \lambda_1(1 - \phi(\mathbf{x})) + \lambda_2 P_{\text{unmet}}(\mathbf{x}) + \lambda_3 P_{\text{rep}}(\mathbf{x}) \tag{10}$$

where $1 - \phi(\mathbf{x})$ denotes the coverage loss, $P_{\text{unmet}}(\mathbf{x})$ denotes the penalty for uncovered villages, and $P_{\text{rep}}(\mathbf{x})$ denotes the penalty for distance repulsion. λ_1, λ_2 and λ_3 represent the weights assigned to each penalty term. Since the objective function is minimized, signal coverage is expressed as coverage loss to prevent unreasonable solutions. Considering that in real-world scenarios the priority of whether a signal is covered is higher than that of signal strength, the weight relationship in the objective function

is set as $\lambda_1 > \lambda_2 > \lambda_3$. This setting better reflects practical conditions and makes the optimization more reliable. In this study, λ_1 is set to 0.5, λ_2 is set to 0.3, and λ_3 is set to 0.2.

4 Proposed Algorithm

This section presents the proposed algorithm and its strategies. Differential Evolution (DE), known for its simplicity, low parameterization, easy implementation, and strong global search, is widely used in optimization. Considering cost and performance, this study applies an improved DE to optimize base station planning in mountainous areas.

4.1 The Original DE Algorithm

DE is a swarm-based global optimizer representing solutions as real-valued vectors. It iteratively applies mutation, crossover, and selection: differences generate mutation vectors, which combine with current solutions; fitter candidates replace originals. This balances exploration and exploitation, guiding the population to the global optimum. Each individual x_i generates a mutation vector v_i as follows:

$$v_i = x_{r1} + F \cdot (x_{r2} - x_{r3}) \tag{11}$$

Here, x_{r1}, x_{r2} and x_{r3} are three distinct individuals randomly selected from the population, different from the current individual. $F \in [0, 2]$ is the scaling factor used to control the amplification of the differential vector. After the mutation operation produces the mutation vector $v_{i,j}$, it must be combined with the current individual x_i through the following crossover formula to generate the trial individual $u_{i,j}$:

$$u_{i,j} = \begin{cases} v_{i,j} & \text{if } \text{rand}(0,1) \leq CR \text{ or } j = j_{\text{rand}}, \\ x_{i,j} & \text{otherwise.} \end{cases} \tag{12}$$

In DE, i and j denote the individual and its dimension. The trial vector component $u_{i,j}$ is generated via crossover, with a random index j_{rand} ensuring at least one element from the mutation vector. The crossover rate CR controls exchange probability. The trial vector then competes with the parent, and the fitter one enters the next generation, guiding the population toward the global optimum.

4.2 Motivation

Despite many new methods, DE research remains active after two decades. With simplicity, strong search ability, and robustness, DE offers superior convergence and reliability, especially in high-dimensional and multimodal problems. As one of the most representative and enduring intelligent optimization algorithms [33], it has been widely applied to complex continuous global optimization tasks such as base station planning [34], path planning [35], engineering optimization [36], and multi-objective optimization [37].

However, DE has limitations: in high-dimensional problems it quickly loses diversity, and in dynamic problems it adapts poorly and responds slowly. To address this, various improvements have been proposed, generally falling into the following strategy categories:

- Mutation strategy design: Uses flexible combinations of differential vectors to explore the search space more effectively, avoid local optima, and improve global search, robustness,

and adaptability. Jin et al. proposed the qSTDE algorithm, which employs a quartile feedback mechanism to dynamically switch mutation strategies at different stages, thereby balancing exploration and exploitation [38]. Baig et al. introduced a new mutation strategy, DE/Neighbor/2, which incorporates neighborhood information and weighted differentials, significantly improving convergence speed and global optimization performance [39]. Meanwhile, Fadhil et al. designed a novel mutation strategy, DE/current-to-best/2, by combining the current, global best, and random vectors, enhancing both diversity and convergence [40].

- Parameter setting: The scaling factor F and crossover probability CR are key parameters in DE, strongly influencing both convergence speed and solution accuracy. Traditional DE relies on fixed or manually tuned parameters. Automatic adaptation reduces dependence on prior experience, balances exploration and exploitation, and improves convergence speed and solution quality. Bajer proposed a parameter-control scheme that stores and reuses successful F and CR values, improving performance stability across stages [41]. Sharma et al. applied Double Deep Q-Learning (DDQN) to dynamically choose mutation strategies each generation, enabling online parameter control and strategy selection [42]. Yan et al. proposed a mechanism that generates two sets of F and CR based on population diversity and adaptively selects the better one by diversity ranking [43].
- Selection mechanism optimization: New selection mechanisms improve solution quality by smarter filtering, yielding better distribution, diversity, and multi-objective handling. Kitamura et al. removed elimination, retaining all individuals to enhance the search by preserving the full candidate set [44]. Zheng et al. proposed a selection rule that adapts strategy choices to the evolutionary stage, improving both diversity and convergence quality [45]. Zhang et al. proposed an External Selection Mechanism (ESM) that uses external evaluation or archives to improve elite retention and search stability [46].
- Hybrid algorithms: Standard DE may converge slowly or get trapped locally. Hybridizing with other swarm algorithms enables better solutions under the same budget. Xu et al. integrated DE's mutation and crossover into PSO with dynamic inertia, acceleration, and velocity updates, enhancing escape from local optima and global search performance [47]. Tan et al. divided the population into subgroups using a soft-island model for information exchange, while each subgroup applied a unified local search to enhance diversity and exploitation [48]. Zhang et al. proposed the MLCC framework with a multilayer structure where multiple DE instances run in parallel. Individuals choose evolutionary paths by preference, while competition and cooperation share resources, greatly enhancing search performance [49].

Although DE variants perform well, they struggle in high-dimensional problems where local optima proliferate and the search space grows exponentially. Balancing diversity and efficiency remains a key challenge. To balance diversity and efficiency, we propose a Multi-Strategy Mutation Differential Evolution (MSM-DE). A new operator applies a force away from local optima, while strategies are adaptively chosen by stage-wise success rates. A parameter memory archive adjusts F andCR adaptively, reducing manual tuning.

4.3 MSM-DE

DE evolves solutions via mutation, crossover, and selection, but limited information flow causes premature or slow convergence. A multi-strategy mutation scheme is proposed, dynamically favoring effective strategies while preserving diversity, enhancing adaptability, information sharing, and global search. The overall framework diagram of the algorithm is shown in Fig. 1.

- Multi-strategy mutation pool: The multi-strategy mutation pool, central to MSM-DE, manages three operators: $p-best/1$ (toward superior individuals), *Balanced* (attracting local optima while repelling worst), and *Reverse* (exploration). Individuals choose strategies by success rates with parameter memory to balance exploration and exploitation. Compared with single mutation, the pool better uses population information, preserves diversity, prevents premature convergence, and dynamically activates effective operators for faster, more robust convergence.

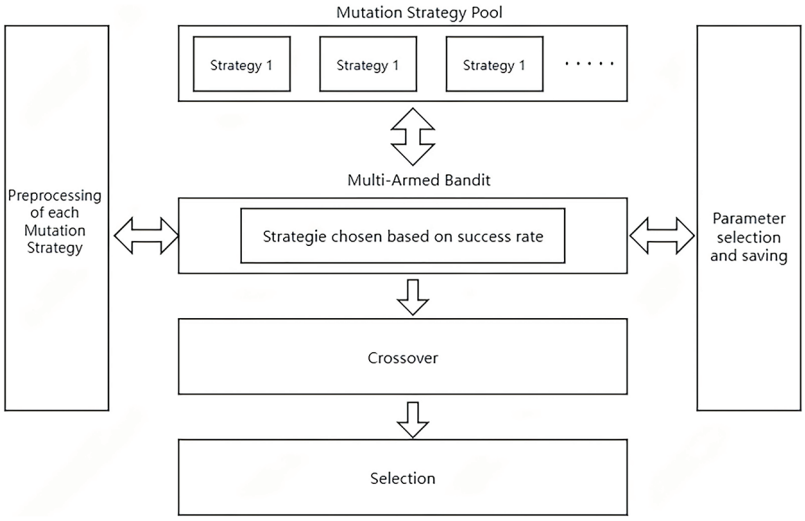

Figure 1: framework diagram of MSM-DE

- *Reverse* mutation operator: Unlike opposition-based learning, this operator selects reverse exploration anchors through local sparsity and fitness normalization. Eq. (13) computes the average distance to the five nearest neighbors, (14) normalizes fitness, and (15) ranks candidates, with the top particle chosen as the anchor.

$$\rho_i = \frac{1}{5}\sum_{k=1}^{5} d_{i,(k)} \tag{13}$$

$$\hat{f}_i = \frac{f(x_i) - \min_j f(x_j)}{\max_j f(x_j) - \min_j f(x_j) + \varepsilon} \tag{14}$$

$$s_i = \rho_i - \lambda \hat{f}_i \tag{15}$$

In Eq. (13), $d_{i,(k)}$ denotes the Euclidean distance between particle x_i and its neighbor x_k; $min_j f(x_j)$ represents the best fitness value in the current generation, while $max_j f(x_i)$ corresponds to the worst fitness value in the current generation. In Eq. (14), $\varepsilon > 0$, and in Eq. (15), the recommended value of λ is 0.3. After the reverse exploration anchor is determined, the *Reverse* mutation operator first performs a differential drift to obtain the base point, and then, according to Eq. (16), applies a half-step attraction toward the anchor point to realize reverse exploration.

$$v_i = \text{base} + \alpha(x_{\text{sel}} - \text{base}) = (1-\alpha)\text{base} + \alpha x_{\text{sel}} \tag{16}$$

Here, *base* is the base point from differential drift, x_{sel} the anchor point, and α the attraction strength. This mutation scheme prevents local clustering, directs exploration toward promising

optima, and balances exploration with exploitation. Fig. 2 illustrates the Reverse mutation process. The pseudocode procedure is shown in Algorithm 1.

- Adaptive Strategy Scheduling Mechanism: DE usually relies on a single mutation operator with fixed F and CR, leading to limited adaptability and premature convergence. To address this, we propose an adaptive scheduling framework that assigns the most effective operator at each stage. Based on a simplified Multi-Armed Bandit model, it treats the three operators as "arms," as shown in Fig. 3.
- During the search, strategies with higher recent success rates are prioritized, while a small probability explores alternatives. Scoring is based on success rate, defined as follows:

$$\hat{p}_k = \frac{S_k}{T_k} \tag{17}$$

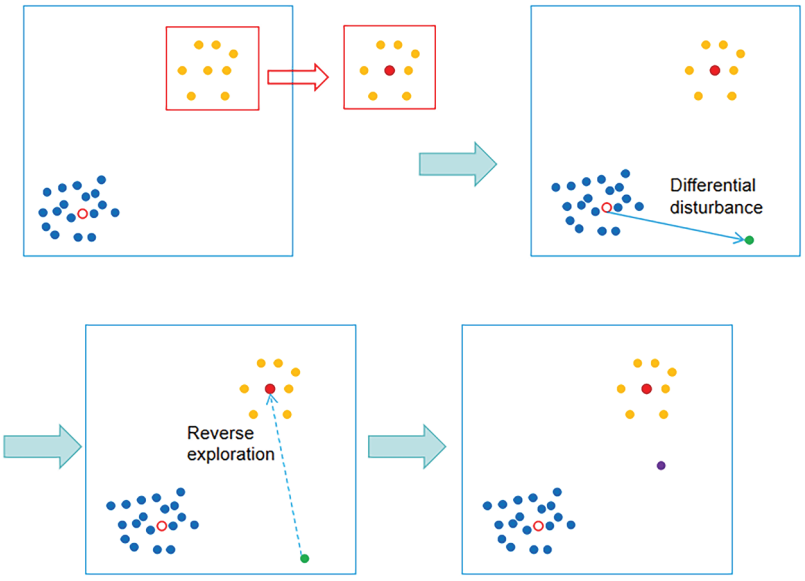

Figure 2: Reverse mutation process

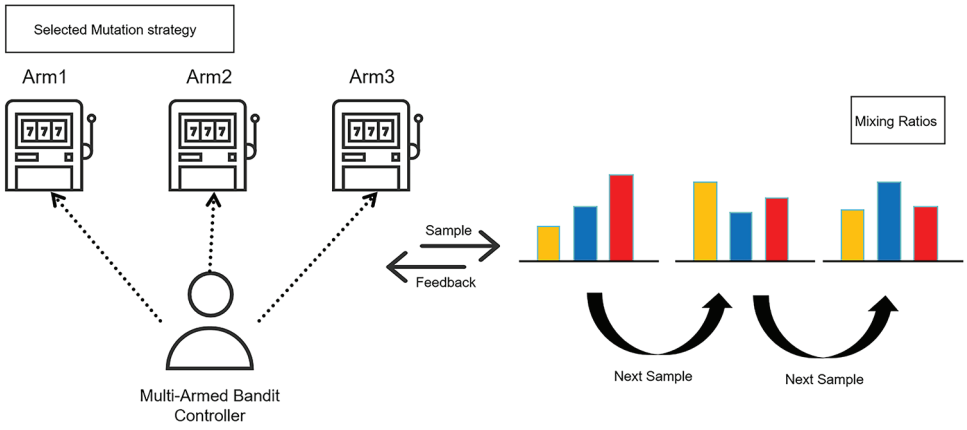

Figure 3: Multi-Armed Bandit

Algorithm 1: Reverse mutation strategy

Require: $pop[1\ldots NP]$, Current population; $f[1\ldots NP]$, Correspondence; $memF[1\ldots M]$, $memCR[1\ldots M]$, Successfully driven parameter memory; Knn, Number of neighbors; λ, Tradeoff coefficient; α, Traction coefficient; ε, Prevent zero constants.

Ensure: The updated population, along with the current generation's successful parameter sets S_F and S_{CR}.

```
Compute the Euclidean distance matrix D.
for p = 1...NP do
    Calculate the mean distance of the 5 nearest neighbors according to Eq. (13).
end for
for p = 1...NP do
    Calculate the normalized fitness according to Eq. (14).
    Perform superiority-inferiority scoring according to Eq. (15).
end for
Select the anchor point X_sel.
for p = 1...NP do
    Perform parameter selection according to Eqs. (18) and (19).
    Obtain the base point base through differential drift.
    Perform reverse exploration according to Eq. (16).
    Perform the crossover operation according to Eq. (12).
    Select and retain the better particle, and record the parameters F and CR.
end for
```

Here, $\hat{p}_k$ denotes the success rate, S_k the number of successes, and T_k the number of trials. A trial is successful if the offspring has better fitness than its parent. Operators with higher success rates are given higher selection probabilities. On this basis, we further incorporate a success-driven memory and random sampling mechanism to update F and CR. A success-parameter memory archive is constructed and updated using the Lehmer mean, from which the next generation samples (F_i,CR_i) with added noise, thereby realizing adaptive parameter adjustment. The formula is given as follows:

$$F_i = \mathrm{clip}_{[F_{\min},F\max]}\left(\mu_F^{(idx_m)} + \gamma \tan\left(\pi\left(U - \frac{1}{2}\right)\right)\right), U \sim \mathcal{U}(0,1) \tag{18}$$

$$CR_i = \mathrm{clip}_{[0,1]}\left(\mu_{CR}^{(idx_m)} + \sigma Z\right), Z \sim \mathcal{N}(0,1), \sigma = 0.1 \tag{19}$$

Here, $\mathrm{clip}_{[a,b]}(x) = \min\{\max\{x,a\},b\}$, idx_m denotes the memory slot, and γ is set to 0.1 in this study. The framework adaptively selects suitable mutation operators at different stages, balancing exploration and exploitation, while remaining extensible to include additional operators. As shown in Algorithm 2, the pseudocode procedure is presented.

Algorithm 2: Mechanism of multiple strategies mutation

Require: $K = K_1, K_2, \ldots, K_i$, Mutation strategy set; NP, Population size; D, dimension; G, generations; memF[1. . .M], memCR[1. . .M], Successfully driven parameter memory; ε, Prevent zero constants.
Ensure: x_best, f_best, P, Sampling probability.

for $g = 1. . .G$ **do**
  for for i in K **do**
    Calculate the success rate according to Eq. (17).
    Normalize to distribution probability P.
  end for
  for *for* $i = 1. . .NP$ **do**
    Select a strategy according to probability $P[k]$.
    Perform parameter selection according to Eqs. (18) and (19).
    Perform mutation according to the selected strategy.
    Perform the crossover operation according to Eq. (12).
    Select and retain the better particle, and record the parameters F and CR.
    update x_best, f_best.
  end for
end for

- Algorithm workflow: The algorithm initializes the population, evaluates fitness, and identifies the global best. It builds a parameter memory and success counters, precomputes the reverse exploration anchor, and schedules strategies adaptively. F and CR are updated by sampling, a mutation operator is chosen, and better individuals update the global best. Successful trials adjust success rates and store parameters. Pseudocode is shown in Algorithm 3.

Algorithm 3: MSM-DE algorithm

Require: $K = K_1, K_2, \ldots, K_i$, Mutation strategy set; NP, Population size; D, dimension; G, generations; $f(.)$, Objective function; lb, ub, Variable upper and lower bounds;
Ensure: x_best, f_best, X(Population matrix), $fitness$, P(strategy probability), $memF$, $memCR$.

Randomly initialize the population $X[i]$, $i = 1. . .NP$ and initialize $memF$, $memCR$.
Selected (x_best, f_best)
for $g = 1. . .G$ **do**
  for i in K **do**
    Calculate the success rate according to Eq. (17).
    Normalize to distribution probability P.
  end for
  for $i = 1. . .NP$ **do**
    Select a strategy according to probability $P[k]$.
    Perform parameter selection according to Eqs. (18) and (19).
    Perform mutation strategy.
    Perform the crossover operation according to Eq. (12).
    Record the parameters F and CR and update x_best, f_best.
  end for
end for

- Computational complexity analysis: Computational complexity is a key indicator of algorithmic efficiency. For problem f, let the population size be NP, the solution dimension D, the number of generations G, and the unit cost of function evaluation C_f. The main computational cost of the original DE includes: Mutation: For each individual, three distinct individuals are selected to generate a mutant vector, with complexity $O(NPD)$; Crossover: The mutant vector is combined with the target vector, with complexity $O(NPD)$; Fitness evaluation: The new solutions are evaluated, with complexity $O(NPC_f)$. Therefore, the overall computational complexity of the original DE is: $O(NP(C_f + D))$. Compared with the original DE, MSM-DE incurs additional costs due to its multiple mutation strategies: the Best strategy requires fitness ranking with complexity $O(NPlogNP)$; the Balanced and Reverse strategies require distance-matrix computation with complexity $O(NP^2D)$; and the Reverse strategy involves nearest-neighbor extraction with complexity $O(NP^2logNP)$. Hence, the overall complexity of MSM-DE is $O(NP(C_f + D)) + O(NPlogNP) + O(NP^2D) + O(NP^2logNP)$. When $C_f >> NPD$, the extra cost becomes negligible; otherwise, it grows significantly. This indicates that MSM-DE is more suitable for computationally expensive continuous optimization problems, such as the mountainous base station deployment model considered in this study.

5 Experiments and Aanlysis

In a 30 × 30 area, we evaluate MSM-DE against several representative DE variants, including MDE [50], IDEBW [51], JADE [52], MSDE [53], the original DE, DACDE [54] and the SA2SDE [55]. Baseline algorithms are taken from established studies to ensure credible comparisons. With a population of 100 and 1000 generations, experiments test mountainous base station deployment under fixed- and random-user scenarios with varying station numbers. Final layouts are obtained by minimizing Eq. (10). All simulations and experiments in this study were conducted in the MATLAB R2023a environment on a Windows 10 64-bit computer equipped with an Intel Core i5-10300H processor (2.5 GHz, 4 cores, 8 threads) and 16 GB of RAM.

5.1 Evaluation Metrics

To evaluate the performance of the algorithms in base station layout optimization, each algorithm was independently executed 30 times. The following evaluation metrics were employed to verify the rationality and superiority of the algorithms: the best objective function value among the 30 runs (Best), the average fitness (Mean_f), and the standard deviation of fitness (Std_f). In addition, Welch's ANOVA significance test and the Games-Howell post-hoc comparison analysis were conducted, and a significance ranking table of the algorithms was added to more rigorously assess the performance differences among them. It should be noted that, in the random model, the randomly generated village locations may appear on mountain tops, slopes, or other unrealistic positions, which can lead to infeasible solutions during optimization. In contrast, in the fixed model, due to a more reasonable village distribution, all algorithms can converge to values close to zero, indicating that they are capable of obtaining reasonable optimization results. Therefore, to further evaluate the performance of the algorithms, the average running time (Mean_t(s)) and the standard deviation of running time (Std_t(s)) were introduced as auxiliary evaluation indicators. The parameter settings of the algorithms are presented in Table 1. To intuitively illustrate the differences among the algorithms and their deployment schemes, four base station deployment scenarios with 30, 40, 50, and 60 stations were designed, and simulation experiments were conducted under both fixed and random mountainous village distributions. To improve clarity and facilitate comparison, the performance indicators in Tables 2 to 4 where MSM-DE achieves the best results among all evaluated methods are highlighted in

bold. This formatting is used solely to denote the numerically superior values obtained by MSM-DE and does not imply any additional emphasis. All experiments are conducted under identical settings, and the bolded entries consistently demonstrate the advantage of the proposed MSM-DE method over the baseline and state-of-the-art algorithms.

Table 1: Internal parameter settings for each algorithm

Algorithm	Parameter
DE	$F = 0.8$, $CR = 0.9$
MDE	$tau = 0.90$, $updatePeriod = 10$, $p\_rotate = 0.80$, $p\_comp = 0.30$, $useBest = true$, $p\_rate = 0.1$
IDEBW	$F = 0.8$, $CR = 0.9$, $w\_max = 0.9$, $w\_min = 0.1$
MSDE	$F = 0.8$, $CR = 0.9$, $linspace(1, 0)$
JADE	$p = 0.05$, $memorySize = 5$, $mem\_f = 0.5$, $mem\_cr = 0.5$
MSM-DE	$memF = memCR = 0.5$, $p\_rate = 0.1$, $M = k = 5$, $Cauchy(memF, 0.1)$ˇ$[0.1, 0.9]$, $N(memCR, 0.1)$ˇ$[0, 1]$
DACDE	$P_d min = 0.1$, $\alpha \sim (0.5,1.0)$
SA2SDE	$T_0 = 1.0$, $\lambda = 0.95$

5.2 Fixed Model

The fixed model simulates realistic village distributions in mountainous areas, where villages are typically concentrated in valleys, plains, and areas close to water resources. In this model, the positions of villages are deliberately arranged to reflect actual geographical and environmental conditions, thereby avoiding unrealistic cases such as villages being located on steep slopes or mountain tops. This setup ensures that the optimization algorithms are evaluated under more practical and representative scenarios, allowing a fairer comparison of their effectiveness. By employing this model, we are able to analyze the convergence behavior, stability, and adaptability of the algorithms in solving problems that closely resemble real-world deployment challenges. The detailed numerical results and performance comparisons obtained under the fixed model are presented in Table 2.

Table 2: Layout effects under each algorithm's fixed model

Scale	Index	DE	MDE	IDEBW	MSDE	JADE	MSM-DE	DACDE	SA2SDE
30	Best	0.25861	0.18639	0.20056	0.18778	0.17222	0.17667	0.205	200.26
	Mean_f	160.26	0.20856	20.242	0.19308	0.17906	0.17906	60.248	320.28
	Std_f	126.48	0.0098242	63.25	0.0037644	0.0028725	**0.0016048**	96.613	103.28
	Mean_t(s)	40.996	41.346	41.888	116.54	41.136	42.644	77.337	226.23
	Std_t(s)	2.6836	1.3804	0.77922	4.0847	1.23	**0.81816**	0.80708	4.3349
40	Best	0.19639	0.14056	0.17778	0.12722	0.12194	**0.11806**	0.19889	0.21889
	Mean_f	20.213	0.15039	0.20111	0.13342	0.12525	**0.12386**	0.20781	80.219
	Std_f	63.247	0.0052708	0.0090948	0.0048468	0.0025875	**0.0022546**	0.0043557	103.27
	Mean_t(s)	68.147	70.349	70.231	194.33	68.947	71.47	131.15	419.03
	Std_t(s)	1.0033	0.77114	0.91208	2	1.1788	**0.99775**	1.4625	2.4618
50	Best	0.17194	0.098889	0.1575	0.088611	0.076667	**0.072167**	0.16278	0.18028
	Mean_f	0.17894	0.12339	0.16022	0.10217	0.081778	**0.080667**	0.16672	0.19006
	Std_f	0.0089611	0.022883	0.0025108	0.009178	0.0043417	**0.0020787**	0.0051046	0.0063507
	Mean_t(s)	102.97	107.89	106.11	295.57	104.45	108.63	197.53	774.98

(Continued)

Table 2 (continued)

Scale	Index	DE	MDE	IDEBW	MSDE	JADE	MSM-DE	DACDE	SA2SDE
	Std_t(s)	1.1946	0.96218	1.397	4.8583	1.4439	**0.94455**	1.4273	18.104
60	Best	0.13028	0.078611	0.11667	0.095278	0.050878	**0.050556**	0.12611	0.14278
	Mean_f	0.13356	0.11339	0.12611	0.10494	0.054389	**0.052389**	0.13333	0.1535
	Std_f	0.0030264	0.023809	0.0066811	0.0096273	0.0026889	**0.0024184**	0.0041989	0.0081923
	Mean_t(s)	147.07	160.86	149.32	417.96	149.02	154.33	282.39	1232.1
	Std_t(s)	3.6668	3.1229	2.9076	2.4066	2.8817	**2.3501**	2.7442	20.842
	rank	7	4	5	3	2	1	6	8

Fig. 4 illustrates the average convergence performance of different algorithms under the fixed model with varying base station scales. a(30), b(40), c(50), d(60). In the fixed model, as the number of base stations increases, the performance differences among algorithms become more pronounced, with MSM-DE exhibiting a clearer advantage. The overall evaluation results of the algorithms are generally consistent with the conclusions drawn in the previous section. It is worth noting that when the number of base stations is 30, JADE achieved a slightly better best fitness value than MSM-DE. According to the results of Welch's ANOVA significance test and the Games–Howell post-hoc comparison analysis, MSM-DE and JADE did not show a statistically significant difference when the base station scale was 30, but both performed significantly better than other DE variants except JADE. Since MSM-DE may employ multiple mutation strategies during execution, each requiring additional computational overhead, its computation time is relatively higher, making MSM-DE less advantageous in terms of runtime. However, benefiting from multiple mutation strategies suitable for different evolutionary stages, MSM-DE usually achieves better fitness values and smaller standard deviations. In addition, the dynamic parameter adjustment mechanism of MSM-DE further enhances its precision, giving it a clear advantage over algorithms with fixed parameters. This demonstrates that MSM-DE possesses stronger stability and convergence accuracy. Moreover, compared with the random model, MSM-DE performs even better under the fixed model, achieving higher convergence accuracy and shorter convergence time, further verifying the practical value of the algorithm. Fig. 5 illustrates the optimized base station layout distributions obtained by different algorithms under the fixed model when the number of base stations is 30 and the number of villages is 15.

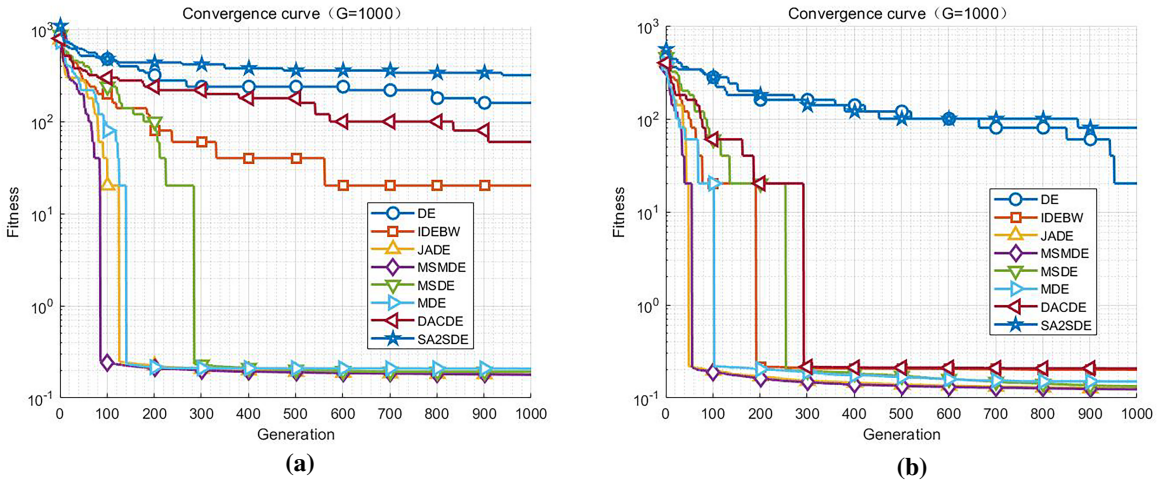

Figure 4: (Continued)

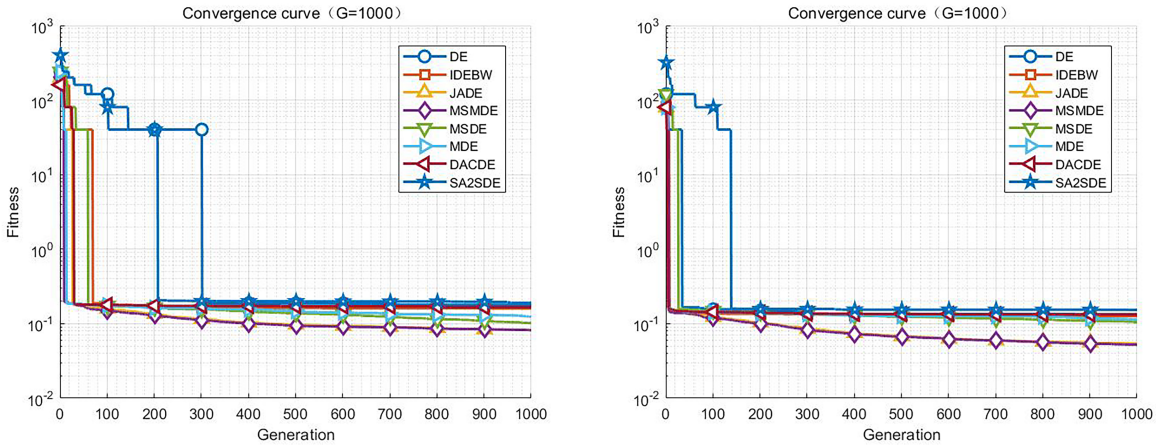

Figure 4: Average convergence curves under the fixed model with different numbers of base stations. (a) 30. (b) 40. (c) 50. (d) 60

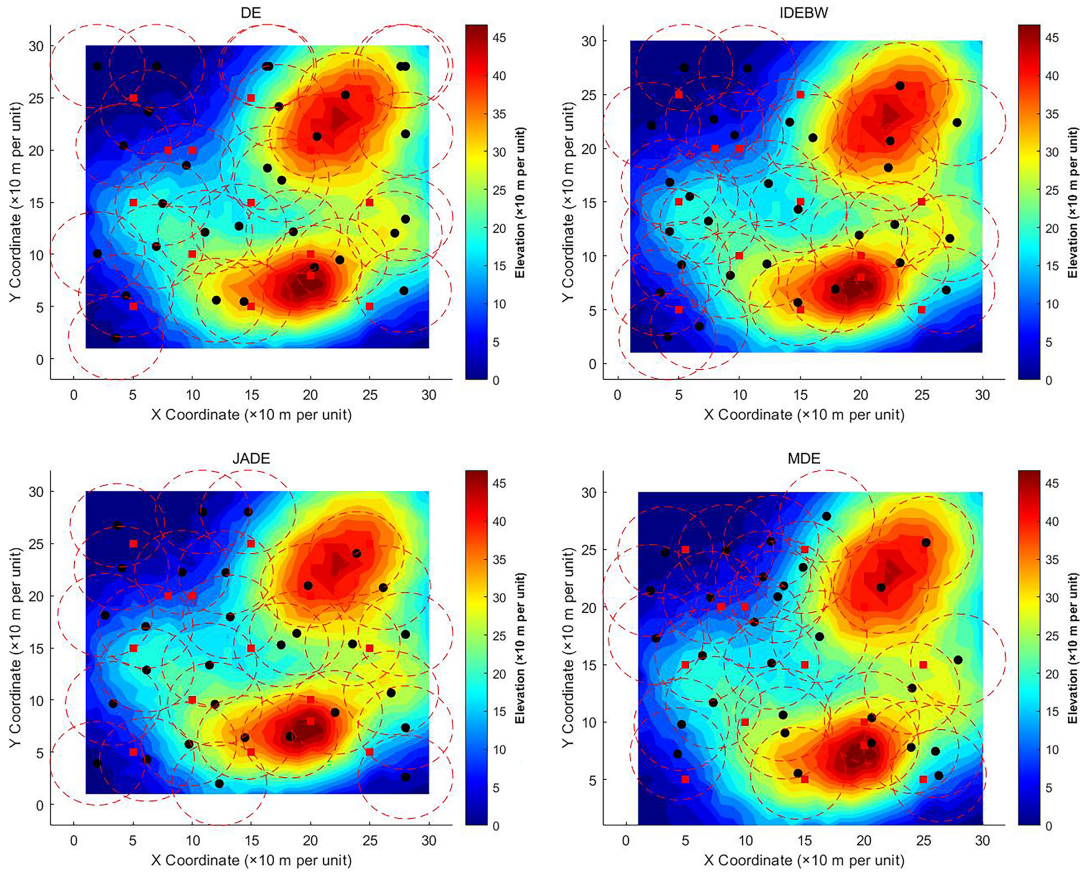

Figure 5: (Continued)

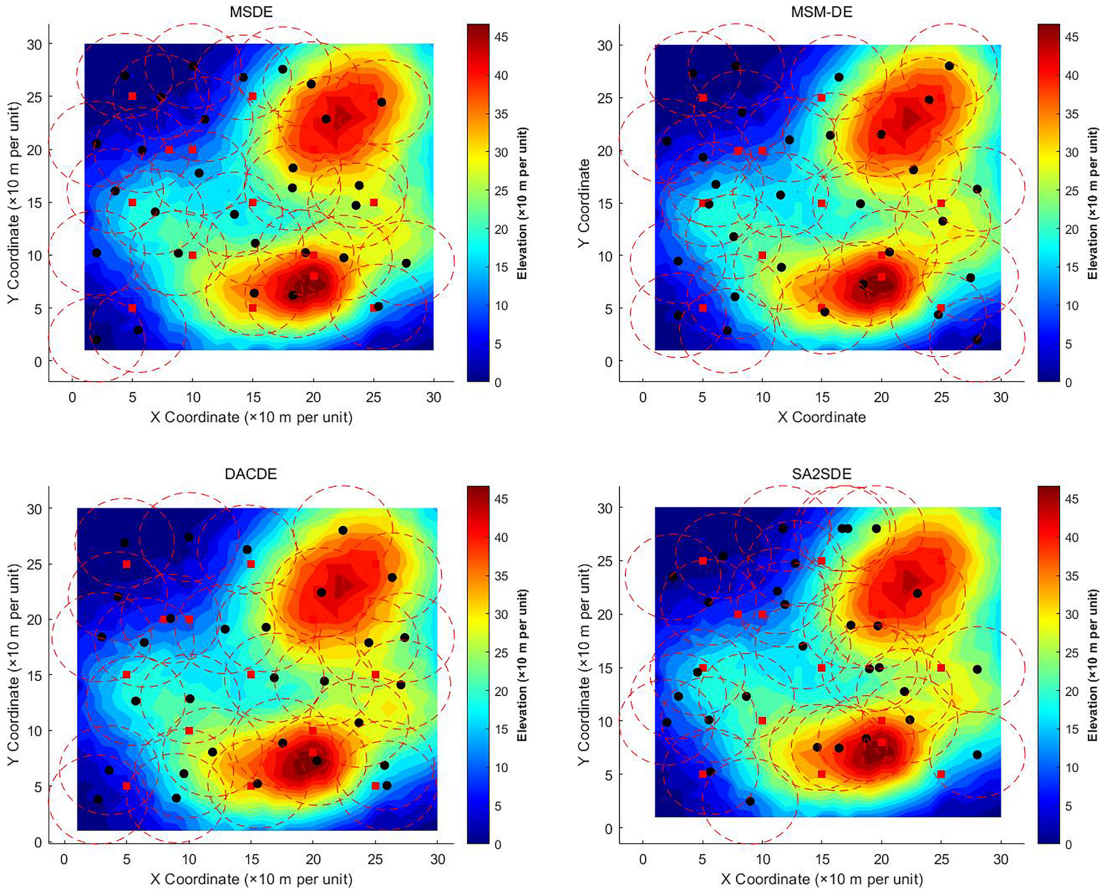

Figure 5: Fixed model base station planning

Fig. 4 shows MSM-DE produces more uniform and dispersed layouts. In dense areas, its stations cluster appropriately for coverage and quality. DE, MDE, and MSDE also achieve full coverage but are overly clustered, while JADE and IDEBW perform well but remain inferior to MSM-DE.

5.3 Random Model

To ensure robustness under varying conditions, experiments were conducted in randomly distributed mountain village layouts. Each algorithm was run 30 times with different village distributions, and the optimization results are shown in Table 3.

Table 3: Layout effects under each algorithm's random model

Scale	Index	DE	MDE	IDEBW	MSDE	JADE	MSM-DE	DACDE	SA2SDE
	Best	200.26	0.21389	0.2025	0.185	0.17944	**0.17889**	200.27	200.26
	Mean_f	280.27	80.219	80.237	80.192	80.181	80.18	240.27	320.27
30	Std_f	109.55	109.54	109.56	109.54	109.54	109.54	89.435	109.54
	Mean_t(s)	38.026	39.198	38.129	109.01	39.174	40.806	73.369	207.43
	Std_t(s)	0.99879	1.122	1.5064	1.1368	1.0514	**0.81662**	1.71	1.8957
	Best	0.22889	0.15361	0.19583	0.12889	0.12361	**0.12111**	0.215	0.22778
	Mean_f	140.23	100.17	100.21	100.13	100.13	**60.126**	120.22	220.24
40	Std_f	134.98	141.41	141.41	141.42	141.42	**134.99**	139.83	175.12
	Mean_t(s)	66.283	69.257	67.892	189.14	67.91	70.645	127.42	422.32
	Std_t(s)	1.8118	1.9465	1.9923	4.3269	1.7746	**1.6079**	2.8547	3.9718
	Best	0.18056	0.14944	0.1575	0.097778	0.085278	**0.078056**	0.16306	0.18417
	Mean_f	120.19	120.16	120.17	120.11	120.09	**120.08**	120.17	160.2
50	Std_f	139.84	139.83	139.84	139.84	139.84	139.84	139.84	157.75
	Mean_t(s)	102.78	109.3	105.98	294.58	105.64	110.32	199.91	767.3
	Std_t(s)	2.643	2.7398	2.5394	5.9829	2.0484	**1.7845**	5.278	8.2023
	Best	0.14194	0.070556	0.12139	0.090833	0.049444	**0.041389**	0.1275	0.14306
	Mean_f	100.15	100.12	100.13	100.11	60.056	**20.056**	100.13	100.16
60	Std_f	169.96	169.97	169.97	169.97	134.99	**63.248**	169.96	169.96
	Mean_t(s)	143.32	159.43	147.63	412.65	148.07	153.48	284.18	1259.5
	Std_t(s)	4.6208	3.1008	3.4011	14.378	2.8978	**2.3286**	18.136	21.28
rank	7	4	5	3	2	1	6	8	

Table 3 shows that MSM-DE achieves the best overall performance among all algorithms, exhibiting a clear advantage over the others. As the number of base stations increases, both its accuracy and runtime improve. Unlike the fixed model, in the random model MSM-DE achieves better fitness values than JADE across deployment schemes with different base station scales, and the performance gap becomes more pronounced as the number of stations increases. Although the use of multiple mutation strategies introduces additional computational costs, MSM-DE still achieves faster convergence and shorter runtime compared with MSDE, DACDE, and SA2SDE. As the base station scale grows, the computational overhead of the multi-mutation strategies in MSM-DE gradually decreases, indicating that MSM-DE is more suitable for handling high-dimensional and complex problems. According to the results of statistical significance tests, MSM-DE demonstrates a clear and significant advantage over all DE variants except JADE. Compared with JADE, MSM-DE achieves higher convergence accuracy and more stable results. It is also noteworthy that MSM-DE adopts the Reverse mutation strategy more frequently in the later stages, which further validates the effectiveness of this strategy.

5.4 Ablation Experiment

To further verify MSM-DE, an ablation study is conducted. DE with only *Reverse* mutation is defined as DE_1, and DE with multi-strategy mutation but without *Reverse* as DE_2. To avoid randomness, the study is run under the fixed model, with results shown in Table 4.

Table 4: Layout effects under de algorithm with each mechanism

Scale	Index	DE_1	DE_2	MSM-DE
30	Best	0.35278	0.34667	**0.34473**
	Mean_f	0.36317	0.35567	**0.35171**
	Std_f	0.0072627	0.004365	**0.0026241**
	Mean_t(s)	43.203	38.936	44.879
	Std_t(s)	3.0805	1.8851	3.3761
40	Best	0.24722	0.24389	**0.1043**
	Mean_f	0.25428	0.24839	**0.1113**
	Std_f	0.0044294	0.0034292	**0.002643**
	Mean_t(s)	73.587	66.203	138.54
	Std_t(s)	2.2159	2.0107	3.3805
50	Best	0.16833	0.16111	**0.145**
	Mean_f	0.17356	0.16433	**0.1216**
	Std_f	0.0045452	0.002681	0.004839
	Mean_t(s)	109.48	100.92	102.13
	Std_t(s)	1.0121	8.8172	10.57
60	Best	0.115	0.10167	**0.1002**
	Mean_f	0.12133	0.10639	**0.1014**
	Std_f	0.0036023	0.0027248	**0.002214**
	Mean_t(s)	154.53	137.49	148.33
	Std_t(s)	1.195	0.9522	1.127

Table 4 shows MSM-DE outperforming DE_1 and DE_2. With 30 stations, DE_2 and MSM-DE perform well while DE_1 lags. As scale increases, all algorithms improve, confirming the mechanisms' rationality and MSM-DE's superior effectiveness.

6 Conclusion

To improve the signal coverage quality of communication base stations in hilly and mountainous regions, this study proposes an optimization objective function that comprehensively considers effective coverage rate, signal uniformity, and signal security, while ensuring that all villages within the target area are reliably covered. In view of the limitations of conventional methods—such as slow convergence and the tendency to fall into local optima in complex terrain environments—we improve the Differential Evolution (DE) framework and propose a novel variant algorithm, termed MSM-DE, which integrates multiple mutation strategies with an adaptive parameter memory mechanism. This design enhances global search capability and local exploitation efficiency simultaneously, making the

algorithm particularly suitable for base station deployment optimization under challenging geographic conditions and diverse constraints.

In the experimental evaluation, we conduct comparative analyses across multiple algorithms under varying base station numbers and constraints settings. Multiple evaluation metrics, including convergence speed, fitness distribution, coverage performance, and computational efficiency, are employed to provide a comprehensive assessment. Furthermore, an ablation study is performed by decomposing the core components of MSM-DE, thereby verifying the specific contributions of each mechanism to overall performance improvement. The experimental results demonstrate that MSM-DE achieves superior performance in base station deployment planning, exhibiting enhanced stability, effectiveness, and robustness. These findings confirm that MSM-DE offers an efficient and practical solution for communication network design in complex mountainous terrains.

Although MSM-DE demonstrates strong performance, several limitations remain. The algorithm still shows room for improvement in optimization capability, particularly under the fixed model with 40 base stations, where its higher computational cost—caused by redundant distance matrix calculations—leads to reduced efficiency. Moreover, the current problem model is relatively simplified, as it does not yet account for deployment costs in complex terrains, signal obstruction by mountains, or propagation attenuation. Future work will focus on three directions: (i) enhancing the problem model to better reflect real-world scenarios by incorporating additional influencing factors; (ii) optimizing the algorithmic framework to improve both efficiency and solution quality; and (iii) exploring mobile carriers such as UAVs to overcome the limitations of static base stations and improve adaptability to dynamic environments.

In terms of potential applications, the proposed MSM-DE framework is not limited to communication base station deployment. Its flexible multi-strategy optimization mechanism can also be effectively extended to other engineering design and spatial optimization problems, such as the placement of wireless sensor networks, emergency response facilities, renewable energy systems, such as wind turbines or solar arrays, and drone communication relays in remote areas. Moreover, MSM-DE provides a promising paradigm for large-scale optimization tasks involving multi-objective trade-offs and complex terrain constraints, thereby offering valuable insights for intelligent infrastructure planning and sustainable network development in both urban and rural environments.

Acknowledgement: Not applicable.

Funding Statement: This work is supported by the National Natural Science Foundation of China (Nos. 62272418, 62102058), Zhejiang Provincial Natural Science Foundation Major Project (No. LD24F020004), the Major Open Project of Key Laboratory for Advanced Design and Intelligent Computing of the Ministry of Education (No. ADIC2023ZD001).

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization and methodology, Chengtian Ouyang and Shunlong Huang; Writing—original draft preparation, Shunlong Huang; Resource collection and investigation, Yongming Zheng; Project administration, Changjun Zhou. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data are available from the corresponding author on reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Wei Z, Qu H, Wang Y, Yuan X, Wu H, Du Y, et al. Integrated sensing and communication signals toward 5G-A and 6G: a survey. IEEE Internet Things J. 2023;10(13):11068–92. doi:10.1109/jiot.2023.3235618.
2. Andrews JG, Buzzi S, Choi W, Hanly SV, Lozano A, Soong ACK, et al. What will 5G be. IEEE J Select Areas Commun. 2014;32(6):1065–82. doi:10.1109/jsac.2014.2328098.
3. Chettri L, Bera R. A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems. IEEE Internet Things J. 2020;7(1):16–32. doi:10.1109/jiot.2019.2948888.
4. Alsabah M, Naser MA, Mahmmod BM, Abdulhussain SH, Eissa MR, Al-Baidhani A, et al. 6G wireless communications networks: a comprehensive survey. IEEE Access. 2021;9:148191–243. doi:10.1109/access.2021.3124812.
5. He J, Yang K, Chen HH. 6G cellular networks and connected autonomous vehicles. IEEE Netw. 2021;35(4):255–61. doi:10.1109/mnet.011.2000541.
6. Zhu X, Jiang C. Integrated satellite-terrestrial networks toward 6G: architectures, applications, and challenges. IEEE Internet Things J. 2022;9(1):437–61. doi:10.1109/JIOT.2021.3126825.
7. Bine LMS, Boukerche A, Ruiz LB, Loureiro AAF. A novel ant colony-inspired coverage path planning for Internet of drones. Comput Netw. 2023;235:109963. doi:10.1016/j.comnet.2023.109963.
8. Dubey R, Louis SJ. Genetic algorithms optimized adaptive wireless network deployment. Appl Sci. 2023;13(8):4858. doi:10.3390/app13084858.
9. Yuan B, He R, Ai B, Chen R, Zhang H, Liu B. Service time optimization for UAV aerial base station deployment. IEEE Internet Things J. 2024;11(23):38000–11. doi:10.1109/jiot.2024.3446583.
10. Zhang S, Ansari N. 3D drone base station placement and resource allocation with FSO-based backhaul in hotspots. IEEE Trans Veh Technol. 2020;69(3):3322–9. doi:10.1109/TVT.2020.2965920.
11. Kalantari E, Shakir MZ, Yanikomeroglu H, Yongacoglu A. Backhaul-aware robust 3D drone placement in 5G+ wireless networks. In: 2017 IEEE International Conference on Communications Workshops (ICC Workshops); 2017 May 21–25; Paris, France. Piscataway, NJ, USA: IEEE; 2017. p. 109–14. doi:10.1109/ICCW.2017.7962642.
12. Ai N, Wu B, Li B, Zhao Z. 5G heterogeneous network selection and resource allocation optimization based on cuckoo search algorithm. Comput Commun. 2021;168:170–7. doi:10.1016/j.comcom.2020.12.026.
13. Kyriazis G, Rouskas A, Karetsos GT. Energy-efficient base station management in heterogeneous networking environments. In: 2015 IEEE Symposium on Computers and Communication (ISCC); 2015 Jul 6–9; Larnaca, Cyprus. Piscataway, NJ, USA: IEEE; 2015. p. 283–8. doi:10.1109/iscc.2015.7405529.
14. Nasir YS, Guo D. Multi-agent deep reinforcement learning for dynamic power allocation in wireless networks. IEEE J Select Areas Commun. 2019;37(10):2239–50. doi:10.1109/jsac.2019.2933973.
15. Chen G, Shao W, Tang F, Sun H. An interpretable evolutionary broad learning system for damage identification in aircraft structures using Lamb waves. Appl Soft Comput. 2025;182:113577. doi:10.1016/j.asoc.2025.113577.
16. Chen G, Shang T, Song W, Shao W, Sun H, Qing X. Multilayer cooperative particle swarm optimizer for feature selection in structural health monitoring. IEEE Sens J. 2025;25(7):12525–37. doi:10.1109/JSEN.2025.3544018.
17. Chen G, Sun H. Multi-strategy improved sparrow search algorithm based on first definition of ellipse and group co-evolutionary mechanism for engineering optimization problems. Clust Comput. 2024;27(10):14005–35. doi:10.1007/s10586-024-04620-2.
18. D-Andreagiovanni F. On improving the capacity of solving large-scale wireless network design problems by genetic algorithms. In: Applications of evolutionary computation. Berlin/Heidelberg, Germany: Springer; 2011. p. 11–20. doi:10.1007/978-3-642-20520-0_2.

19. Ghawy MZ, Amran GA, AlSalman H, Ghaleb E, Khan J, AL-Bakhrani AA, et al. An effective wireless sensor network routing protocol based on particle swarm optimization algorithm. Wirel Commun Mob Comput. 2022;2022(1):8455065. doi:10.1155/2022/8455065.
20. Ou Y, Qin F, Zhou KQ, Yin PF, Mo LP, Mohd Zain A. An improved grey wolf optimizer with multi-strategies coverage in wireless sensor networks. Symmetry. 2024;16(3):286. doi:10.3390/sym16030286.
21. Ayinde BO, Hashim HA. Energy-efficient deployment of relay nodes in wireless sensor networks using evolutionary techniques. Int J Wirel Inf Netw. 2018;25(2):157–72. doi:10.1007/s10776-018-0388-1.
22. Shen J, Zhu D, Li R, Zhu X, Zhang Y, Li W, et al. Efficient base station deployment in specialized regions with splitting particle swarm optimization algorithm. World Wide Web. 2024;27(4):43. doi:10.1007/s11280-024-01282-3.
23. Amaldi E, Capone A, Malucelli F. Planning UMTS base station location: optimization models with power control and algorithms. IEEE Trans Wirel Commun. 2003;2(5):939–52. doi:10.1109/TWC.2003.817438.
24. Anderson HR, McGeehan JP. Optimizing microcell base station locations using simulated annealing techniques. In: Proceedings of the IEEE Vehicular Technology Conference (VTC); 1994 Jun 8–10; Stockholm, Sweden. Piscataway, NJ, USA: IEEE; 2002. p. 858–62. doi:10.1109/VETEC.1994.345212.
25. Al-Hourani A, Kandeepan S, Lardner S. Optimal LAP altitude for maximum coverage. IEEE Wirel Commun Lett. 2014;3(6):569–72. doi:10.1109/lwc.2014.2342736.
26. Zhu D, Shen J, Hu J, Ouyang Z, Hu G, Zhou C, et al. Group merging particle swarm optimization algorithm for rural base station deployment. IEEE Trans Emerg Top Comput Intell. 2025. doi:10.1109/tetci.2025.3558433.
27. Alzenad M, El-Keyi A, Lagum F, Yanikomeroglu H. 3-D placement of an unmanned aerial vehicle base station (UAV-BS) for energy-efficient maximal coverage. IEEE Wirel Commun Lett. 2017;6(4):434–7. doi:10.1109/lwc.2017.2700840.
28. Sun X, Ansari N. Latency aware drone base station placement in heterogeneous networks. In: GLOBECOM 2017-2017 IEEE Global Communications Conference; 2017 Dec 4–8; Singapore. Piscataway, NJ, USA: IEEE; 2018. p. 1–6. doi:10.1109/GLOCOM.2017.8254720.
29. Anjinappa CK, Erden F, Guvenc I. Base station and passive reflectors placement for urban mmWave networks. IEEE Trans Veh Technol. 2021;70(4):3525–39. doi:10.1109/tvt.2021.3065221.
30. Jayapaul ADD, Merline A. An optimized small-cell planning procedure for Heterogeneous Network to improve network energy efficiency. Int J Commun. 2023;36(7):e5456. doi:10.1002/dac.5456.
31. Chen J, Shi Y, Sun J, Li J, Xu J. Base station planning based on region division and mean shift clustering. Mathematics. 2023;11(8):1971. doi:10.3390/math11081971.
32. Avella P, Calamita A, Palagi L. A compact formulation for the base station deployment problem in wireless networks. Networks. 2023;82(1):52–67. doi:10.1002/net.22146.
33. Bilal, Pant M, Zaheer H, Garcia-Hernandez L, Abraham A. Differential Evolution: a review of more than two decades of research. Eng Appl Artif Intell. 2020;90(1):103479. doi:10.1016/j.engappai.2020.103479.
34. Mendes SP, Gomez Pulido JA, Vega Rodriguez MA, Jaraiz Simon MD, Sanchez Perez JM. A differential evolution based algorithm to optimize the radio network design problem. In: 2006 Second IEEE International Conference on e-Science and Grid Computing (e-Science'06); 2006 Dec 4–6; Amsterdam, The Netherlands. Piscataway, NJ, USA: IEEE; 2006. doi:10.1109/E-SCIENCE.2006.261052.
35. Rodrguez-Molina A, Sols-Romero J, Villarreal-Cervantes MG, Serrano-Prez O, Flores-Caballero G. Path-planning for mobile robots using a novel variable-length differential evolution variant. Mathematics. 2021;9(4):357. doi:10.3390/math9040357.
36. Vasile M, Minisci E, Locatelli M. An inflationary differential evolution algorithm for space trajectory optimization. IEEE Trans Evol Computat. 2011;15(2):267–81. doi:10.1109/tevc.2010.2087026.
37. Mlakar M, Petelin D, Tu-ar T, Filipic B. GP-DEMO: differential evolution for multiobjective optimization based on Gaussian process models. Eur J Oper Res. 2015;243(2):347–61. doi:10.1016/j.ejor.2014.04.011.

38. Jin P, Cen J, Feng Q, Ai W, Chen H, Qiao H. Differential evolution with the mutation strategy transformation based on a quartile for numerical optimization. Appl Intell. 2024;54(1):334–56. doi:10.1007/s10489-023-05038-w.
39. Baig MH, Abbas Q, Ahmad J, Mahmood K, Alfarhood S, Safran M, et al. Differential evolution using enhanced mutation strategy based on random neighbor selection. Symmetry. 2023;15(10):1916. doi:10.3390/sym15101916.
40. Fadhil S, Zaher H, Ragaa N, Oun E. A modified differential evolution algorithm based on improving a new mutation strategy and self-adaptation crossover. MethodsX. 2023;11:102276. doi:10.1016/j.mex.2023.102276.
41. Bajer D. Parameter control for differential evolution by storage of successful values at an individual level. J Comput Sci. 2023;68:101985. doi:10.1016/j.jocs.2023.101985.
42. Sharma M, Komninos A, Lpez-Ibez M, Kazakov D. Deep reinforcement learning based parameter control in differential evolution. In: Proceedings of the Genetic and Evolutionary Computation Conference. Prague, Czech Republic. New York, NY, USA: ACM; 2019. p. 709–17. doi:10.1145/3321707.3321813.
43. Yan R, Zheng L, Jin X. Parameter adaptive differential evolution based on individual diversity. Symmetry. 2025;17(7):1016. doi:10.3390/sym17071016.
44. Kitamura T, Fukunaga A. Is selection all you need in differential evolution. arXiv:2506.14425. 2025.
45. Zheng L, Wen Y. A multi-strategy differential evolution algorithm with adaptive similarity selection rule. Symmetry. 2023;15(9):1697. doi:10.3390/sym15091697.
46. Zhang H, Wang D. An external selection mechanism for differential evolution algorithm. Comput Intell Neurosci. 2022;2022:4544818. doi:10.1155/2022/4544818.
47. Xu H, Deng Q, Zhang Z, Lin S. A hybrid differential evolution particle swarm optimization algorithm based on dynamic strategies. Sci Rep. 2025;15(1):4518. doi:10.1038/s41598-024-82648-5.
48. Tan X, Shin SY, Shin KS, Wang G. Multi-population differential evolution algorithm with uniform local search. Appl Sci. 2022;12(16):8087. doi:10.3390/app12168087.
49. Zhang SX, Zheng LM, Tang KS, Zheng SY, Chan WS. Multi-layer competitive-cooperative framework for performance enhancement of differential evolution. Inf Sci. 2019;482:86–104. doi:10.1016/j.ins.2018.12.065.
50. Li Y, Feng J, Hu J. Covariance and crossover matrix guided differential evolution for global numerical optimization. SpringerPlus. 2016;5(1):1176. doi:10.1186/s40064-016-2838-5.
51. Kumar P, Ali M. Improved differential evolution algorithm guided by best and worst positions exploration dynamics. Biomimetics. 2024;9(2):119. doi:10.3390/biomimetics9020119.
52. Zhang J, Sanderson AC. JADE: adaptive differential evolution with optional external archive. IEEE Trans Evol Computat. 2009;13(5):945–58. doi:10.1109/tevc.2009.2014613.
53. Cai Z, Gao S, Yang X, Zhou M. Multiselection-based differential evolution. IEEE Trans Syst Man Cybern, Syst. 2024;54(12):7318–30. doi:10.1109/tsmc.2024.3447051.
54. Chen B, Ouyang H, Li S, Zou D. Differential evolution algorithm with a complementary mutation strategy and data Fusion-Based parameter adaptation. Inf Sci. 2024;668:120522. doi:10.1016/j.ins.2024.120522.
55. Liu Y, Liu J, Jin Y, Li F, Zheng T. A surrogate-assisted two-stage differential evolution for expensive constrained optimization. IEEE Trans Emerg Top Comput Intell. 2023;7(3):715–30. doi:10.1109/tetci.2023.3240221.