drones

MDPI

Article

Multi-UAV Cooperative Landing Path Planning Based on a Multi-Strategy Improved Grey Wolf Optimizer

Yifeng Wang [1,2], Zhongrui Zhang [1,2], Yiming Peng [1,2,*] and Xiaohui Wei [1,2]

[1] State Key Laboratory of Mechanics and Control for Aerospace Structures, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China; wangyifeng998@nuaa.edu.cn (Y.W.); zhangzr5519@nuaa.edu.cn (Z.Z.); wei_xiaohui@nuaa.edu.cn (X.W.)
[2] Key Laboratory of Fundamental Science for National Defense–Advanced Design Technology of Flight Vehicle, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China
* Correspondence: yimingpeng@nuaa.edu.cn

Highlights

What are the main findings?

- MSGWO achieved 132/180 densely feasible solutions (73.3%) across six five-UAV scenarios, significantly exceeding all six comparators after Holm correction and obtaining the best overall Friedman rank.
- Ablation experiments identified the subpopulation structure as the largest contributor and confirmed a significant feasibility benefit from the nonlinear control parameter.

What are the implications of the main findings?

- Controlled timing experiments showed wall times close to GWO, with directly measured clustering taking less than one percent of MSGWO's optimization time.
- Independent planning and adapted DMGWO produced no jointly feasible solution (0/180 each), demonstrating the need to coordinate arrival and separation jointly.

Abstract

Cooperative landing of multiple unmanned aerial vehicles (UAVs) requires paths that avoid threats, keep safe separations, remain flyable, and allow the swarm to reach its initial approach points at a common time. This paper proposes a multi-strategy improved grey wolf optimizer (GWO), termed MSGWO, for this three-dimensional path planning problem. A multi-population mechanism groups wolves with similar path characteristics and lets each group follow its own leaders, the nonlinear control parameter of the modified GWO (mGWO) is adopted, and the individual memory of particle swarm optimization (PSO) retains the best position found by each wolf. An explicit constrained joint model uses Deb's feasibility rules and dense verification. Across six static-threat scenarios with 30 runs each, MSGWO achieves 73.3% feasibility versus 0–55.6% for six comparators, significantly higher after Holm correction, and the best overall Friedman rank. Mutually feasible objective values are significantly lower than GWO, mGWO, and success-history-based adaptive differential evolution with linear population size reduction (L-SHADE) overall, although L-SHADE is better in one scenario. Independent planning and an adapted disturbance-based modified GWO (DMGWO) find no jointly feasible solution in this setting. An ablation study shows that the multi-population mechanism contributes most.

Keywords: multi-UAV; cooperative landing; 3D path planning; grey wolf optimizer; particle swarm optimization; multi-population mechanism; swarm intelligence

Check for updates

Academic Editor: Gianfranco Parlangeli

Received: 25 September 2026
Revised: 14 October 2026
Accepted: 21 October 2026
Published: 22 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

With the rapid development of information technology and artificial intelligence, unmanned aerial vehicles (UAVs) are increasingly used in surveillance, inspection, search and rescue, and logistics. Compared with conventional platforms, UAVs offer greater flexibility and environmental adaptability and can perform missions at a lower cost. However, during a mission, UAVs are susceptible to interference from surrounding objects, buildings, and natural terrain. In particular, when a UAV swarm returns to land, the vehicles must reach the initial approach points of their landing slots at nearly the same time while avoiding threats and keeping safe separations, and they may otherwise interfere with or even collide with one another. Therefore, planning reasonable flight paths for UAV swarms that avoid these potential hazards has become an urgent problem.

In recent years, path planning for UAV swarms has been widely studied. Existing path planning algorithms can be broadly divided into classical algorithms and swarm intelligence algorithms. Classical algorithms include artificial potential field (APF) methods, which have been improved to escape local minima and to maintain the formation of multiple UAVs [1,2]; graph-search methods, such as conflict-based search combined with D* for cooperative dynamic planning [3]; and sampling-based planners, such as optimal rapidly exploring random tree (RRT*) variants for flight in hostile environments [4]. For swarm planning, geometric paths must also be coordinated in time to meet arrival-time and inter-vehicle separation requirements. Swarm intelligence algorithms, such as particle swarm optimization (PSO) and the sparrow search algorithm [5,6], genetic algorithms [7], and hybrid ant colony and artificial bee colony algorithms [8], treat path planning as the minimization of a cost function and can therefore combine such objectives flexibly. Recent examples include a hybrid PSO initialized by RRT* for multi-UAV planning in complex scenarios [9] and PSO variants that encode each path segment by its length and flight angles [10,11]. However, many of these algorithms still show a poor balance between global exploration and local exploitation and are prone to premature convergence.

The grey wolf optimizer (GWO) [12], proposed by Mirjalili et al. in 2014, is a metaheuristic swarm intelligence algorithm inspired by the hunting behavior of grey wolf packs. Compared with other swarm intelligence methods, GWO has a simple principle and few parameters to tune, which makes it easy to implement. It has been widely applied to spectrum sensing [13], feature subset selection [14], multilayer perceptron training [15], and load forecasting of cloud computing resources [16]. However, UAV swarm path planning requires high real-time performance and accuracy, whereas the standard GWO suffers from slow search speed and a tendency to fall into local optima. To address these issues, Xu and Zhao [17] used opposition-based learning to generate ordered individuals during initialization and introduced an elite selection and retention strategy to accelerate population evolution. Sun et al. [18] improved a multi-population GWO for multi-UAV trajectory planning with a nonlinearly decreasing control parameter and a position update that combines dynamic weighted averaging with static averaging. Cao et al. [19] proposed a modified mutation method that optimizes the path and improves search quality. Xu et al. [20] applied Gaussian mutation perturbations to the current best solution to prevent the algorithm from falling into local optima.

Improved GWO variants have also been applied to multi-UAV cooperative path planning in the last two years. Zhang and Wang [21] used chaotic initialization and chaotic local search; Zheng et al. [22] combined dimension-learning hunting, Lévy-flight exploration, and adaptive leader weights; Zhou et al. [23] integrated greedy initialization and *k*-means clustering into GWO for trajectory optimization under dynamic threats; and Gai et al. [24] proposed a leader–follower hybrid of PSO and GWO for constrained UAV path planning. Simultaneous arrival has been considered explicitly as well: Shahid et al. [25]

proposed a multi-population disturbance-based modified GWO (DMGWO) that also uses the local best position of each wolf and ensures four-dimensional collision avoidance and simultaneous arrival, and Wu and Hou [26] used a multi-swarm PSO with elite learning for the simultaneous arrival of UAVs with different speed limits. The present work builds on these ideas; its main differences are the following. The subpopulations of the proposed multi-strategy improved GWO (MSGWO) are formed by clustering the wolves according to the cost components of the paths that they represent, so that each subpopulation gathers paths with similar trade-offs among length, altitude, threat exposure, and constraint violations and follows its own leaders, taken from the personal bests of its members. Moreover, the arrival-time coordination and the vertical-rate and turn-radius limits are evaluated as explicit constraints on the smoothed path that the UAVs actually fly, rather than on the encoded waypoints only.

Recent work addresses different parts of this problem. Cooperative co-evolution with adaptive decision-variable selection decomposes multi-UAV planning while retaining coordination between variable groups [27]. Spherical-motion PSO with dimension learning addresses communication and energy constraints [28], and Fermat–Weber location PSO supports cooperative inspection planning [29]. Social-learning pigeon-inspired optimization incorporates time consistency in cooperative planning [30]; evolutionary multitasking with operator selection and knowledge fusion treats multiple objectives and terrain-dependent threats [31]. These studies motivate explicit treatment of the interaction between individual path quality and group constraints.

Formation and feedback-control studies provide complementary perspectives. Jin et al. [32] study gradient-free cooperative source seeking by quadrotors under disturbances and communication constraints. Adaptive formation trajectory planning using improved multi-agent twin delayed deep deterministic policy gradient (MATD3) [33] addresses unknown obstacles and narrow passages, while rotational potential fields combined with reinforcement learning [34] address coordination in changing environments. Together, they highlight the importance of combining path planning with feedback and communication in future operational systems.

The difficulty lies in the coupling between paths: extending the longest route increases the earliest common arrival time and can force shorter routes to gain distance, changing separation throughout the flight. The resulting joint search space is high-dimensional and nonconvex. Graph-search and sampling methods remain important alternatives. Conflict-based search can coordinate multi-UAV routes [3]; APF-guided bidirectional RRT* combined with model predictive control (MPC) addresses cooperative arrival and inter-UAV collision avoidance [35]; and integrated RRT*–APF planning and control addresses wind disturbances and explicit velocity/acceleration limits [36]. Continuous joint metaheuristic optimization provides a direct way to evaluate these coupled requirements within one candidate, making it well suited to the formulation studied here.

Improved GWO (I-GWO) introduces dimension-learning hunting and greedy selection [37]. Success-history-based adaptive differential evolution with linear population size reduction (L-SHADE), which adapts differential-evolution parameters and reduces population size [38], offers an additional baseline outside the GWO family. Deb's feasibility rules [39] provide a common ordering that separates constraint violation from feasible objective quality.

The proposed MSGWO combines cost-component clustering, a nonlinear control parameter, and individual memory to search for coordinated paths. Each subpopulation selects its own three leaders from its members' personal bests, allowing different parts of the joint search space to be explored concurrently. The experiments show a significant

feasibility advantage over all six comparators and identify multiple leader sets as the strongest contributor. The contributions are:

- An explicit constrained joint planning model that couples common arrival and inter-UAV separation with threat avoidance, vertical-rate limits, and minimum turning radius. Dimensionless objectives measure path quality, and dense verification checks every final solution.
- MSGWO and its information-flow mechanism: subpopulation-specific leader sets support concurrent search directions, the nonlinear schedule regulates exploration and exploitation, and personal-best memory connects individual experience with social guidance.
- Systematic evidence from seven algorithms, separate feasibility and quality comparisons, ablations, sensitivity studies, and swarm-size experiments. Independent-path and no-arrival controls establish the need to address the joint constraints in the tested missions.

The remainder of this paper is organized as follows. Section 2 presents the multi-UAV path planning model. Section 3 reviews the standard GWO and describes the proposed MSGWO. Section 4 reports the simulation results and analysis. Section 5 discusses the results and future work. Section 6 concludes the paper.

2. Multi-UAV Path Planning Model

2.1. Physical and Spatial Modeling

Let $U = \{u_i \mid i = 1, 2, \ldots, N\}$ denote a set of N UAVs, each of which has its own destination; the set of destinations is $G = \{g_i \mid i = 1, 2, \ldots, N\}$. The destination g_i is the initial approach point (IAP) of the landing slot assigned to the i-th UAV: the paths planned in this paper cover the flight from the mission area to these points, from which the final approach and landing are guided by the terminal landing system. For each UAV, the route from its start point to its destination is represented by a sequence of intermediate control points that define the B-spline below. The movement of a UAV is therefore as shown in Figure 1.

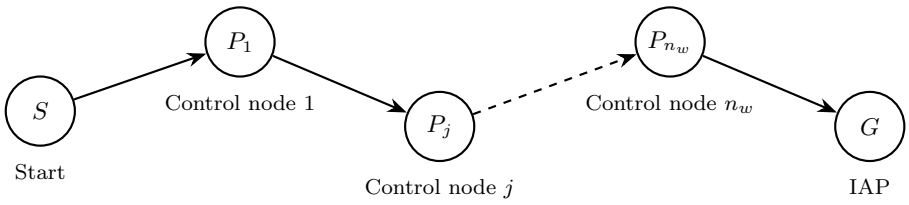


Figure 1. Control polygon for spherical-vector encoding. Intermediate nodes are B-spline control points, not required passage points; the final curve interpolates S and G.

In practice, UAV swarm path planning is often affected by complex environments and various obstacles, such as radar coverage, restricted airspace, hazardous terrain, and adverse weather. Therefore, modeling the complex flight environment of UAVs is of great significance. Each threat is assumed to be a spherical region whose center is located at (x_m, y_m, z_m) and whose radius is r_m. The set of M threats, O, can then be expressed as

$$O = \begin{bmatrix} x_1 & y_1 & z_1 & r_1 \\ x_2 & y_2 & z_2 & r_2 \\ \vdots & \vdots & \vdots & \vdots \\ x_M & y_M & z_M & r_M \end{bmatrix}. \quad (1)$$

Here S_i denotes the start point, $G_i = g_i$ the assigned IAP, $O_m = (x_m, y_m, z_m)$ the threat center, and $r_m = R_{1,m}$ its physical radius. Threats are static and known; ground-centered spheres represent hemispheres in the admissible positive-altitude domain. Distance and time units are km and min.

2.2. Path Representation

The spherical-vector representation of spherical vector-based PSO (SPSO) [10] is adopted to encode the paths. The path of the i-th UAV is described by n_w intermediate nodes, and node k is reached from node $k-1$ by a vector with magnitude $\rho_{i,k}$, pitch angle $\psi_{i,k}$, and heading angle $\varphi_{i,k}$:

$$\begin{aligned} x_{i,k} &= x_{i,k-1} + \rho_{i,k}\cos\psi_{i,k}\sin\varphi_{i,k}, \\ y_{i,k} &= y_{i,k-1} + \rho_{i,k}\cos\psi_{i,k}\cos\varphi_{i,k}, \qquad k = 1, 2, \ldots, n_w, \\ z_{i,k} &= z_{i,k-1} + \rho_{i,k}\sin\psi_{i,k}, \end{aligned} \tag{2}$$

where $(x_{i,0}, y_{i,0}, z_{i,0})$ is the start point of the UAV. The magnitude is bounded by $0 \le \rho_{i,k} \le 2d_i/n_w$, where d_i is the straight-line distance between the start point and the destination; the pitch angle satisfies $|\psi_{i,k}| \le \psi_{\max}$; and the heading angle satisfies $|\varphi_{i,k} - \varphi_{i,0}| \le \varphi_{\max}$, where $\varphi_{i,0}$ is the heading of the line from the start point to the destination. The bounds are $\varphi_{\max} = 60^\circ$ and $\psi_{\max} = \arcsin(w_{\max}/v_{\min})$, where $w_{\max}$ is the vertical-rate limit. These bounds restrict the search to plausible node sequences, and every node is clipped to the boundary of the flight space before the next node is decoded. With $n_w = 10$, each UAV has 30 decision variables, and a candidate solution (a wolf) concatenates the variables of all N UAVs, giving a search space of dimension $30N$.

The start point, the n_w nodes, and the destination are used as the control points of a clamped cubic B-spline. The spline passes through the start point and the destination, lies within the convex hull of its control points, and is twice continuously differentiable, which removes the corners of the control polygon. The spline is sampled at $n_s = S = 401$ points, and all the costs defined below are evaluated on the resulting polyline of $n = n_s - 1$ segments.

The spline $C_i(u)$, $u \in [0,1]$, has uniformly spaced interior knots. Every candidate spline is assessed against the explicit constraints. At S uniformly spaced parameter values, let $P_{ij} = C_i(u_j)$, $\ell_{ij} = \|P_{i,j+1} - P_{ij}\|$, $L_i = \sum_{j=0}^{S-2} \ell_{ij}$, and $s_{ij} = \sum_{q<j} \ell_{iq}/L_i$. Length and time interpolation use this same sampled polyline.

2.3. Path Evaluation Function

For a UAV swarm, the maneuverability of each UAV and the safety and feasibility of the paths must be considered.

2.3.1. Dimensionless Performance Objective

The optimization problem is

$$\begin{aligned} \min_{\boldsymbol{x}\in[\boldsymbol{l},\boldsymbol{u}]} \quad & F(\boldsymbol{x}) = \sum_{i=1}^{N} (w_L f_{L,i} + w_H f_{H,i} + w_M f_{M,i}), \\ \text{subject to} \quad & g_k(\boldsymbol{x}) \le 0, \quad k = 1, \ldots, 7. \end{aligned} \tag{3}$$

The bounds $[\boldsymbol{l}, \boldsymbol{u}]$ refer to the encoding variables. Because a UAV has a maximum cruising range, the length of its flight path must be constrained. During a mission, the flight altitude of the UAV must be kept within a certain range. The three objective components are

$$
\begin{aligned}
f_{L,i} &= L_i / d_i, \\
h_{\mathrm{ref},i}(s) &= (1-s) z_{0,i} + s z_{\mathrm{IAP},i}, \\
f_{H,i} &= \frac{1}{S \Delta h_i} \sum_{j=0}^{S-1} |P_{ij,z} - h_{\mathrm{ref},i}(s_{ij})|, \\
\Delta h_i &= \max(|z_{0,i} - z_{\mathrm{IAP},i}|, 1\ \mathrm{km}), \\
f_{M,i} &= \frac{1}{S-1} \sum_{j=0}^{S-2} \sum_{m=1}^{M} e(D_{ijm}; R_{2,m}, R_{3,m}),
\end{aligned}
\tag{4}
$$

where D_{ijm} is the analytic minimum distance from O_m to segment $P_{ij}P_{i,j+1}$, and

$$
\begin{aligned}
R_{2,m} &= R_{1,m} + \Delta_R, \qquad R_{3,m} = R_{2,m} + \Delta_E, \\
e(D; R_2, R_3) &= \begin{cases} (R_3 - D)/(R_3 - R_2), & R_2 < D \le R_3, \\ 0, & \text{otherwise.} \end{cases}
\end{aligned}
\tag{5}
$$

As described in Section 2.1, various obstacles pose threats to the path. Moreover, in actual operation, weather, communication signals, and aircraft maneuverability may cause deviations in vehicle control. Therefore, a UAV should not only keep a certain distance from each threat but also retain an additional fault-tolerance margin. As shown in Figure 2, the UAV flies from sample point P_{ij} to sample point $P_{i,j+1}$, and D_{ijm} is the shortest distance between this segment and the center of the m-th threat. $R_{1,m}$ is the radius of the region covered by the threat, $R_{2,m}$ is the radius of the region after the fault-tolerance margin is added, and $R_{3,m}$ is the radius beyond which the modeled exposure is zero.

Penetration of R_2 is handled by g_3, so the zero exposure branch inside R_2 never makes such a path preferable to a feasible one. R_1 describes the physical threat and enters through R_2; R_3 is the boundary of modeled exposure. Exposure is averaged over segments and summed over threats, with no division by M. The altitude term is a parameter-grid average using arc-length fraction for its reference. Normalization removes units and direct dependence on the number of samples; the resulting components can still have different empirical scales.

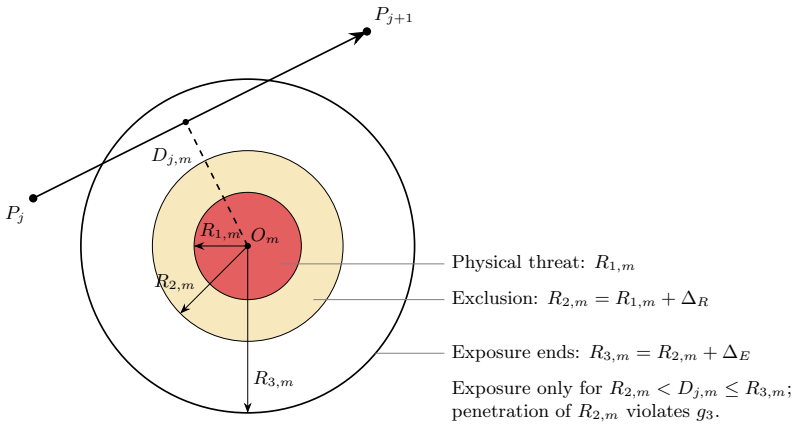


Figure 2. Physical radius R_1, exclusion radius $R_2 = R_1 + \Delta_R$, and outer exposure boundary $R_3 = R_2 + \Delta_E$. The soft exposure term acts only in the annulus outside R_2; exclusion is a separate constraint.

The reference encourages gradual descent toward the IAP. Continuous-descent research motivates avoiding unnecessary level-offs [40,41]. The altitude objective penalizes deviations from this reference profile. The weights $(w_L, w_H, w_M) = (1/6, 1/6, 4/6)$ keep the

length:altitude:threat ratio 1 : 1 : 4 of the original formulation, renormalized after separation was moved from the objective to the constraints. This ratio deliberately lets threat exposure dominate the objective, so that keeping clear of threats takes priority, with path length and altitude deviation as secondary criteria. Section 4.8 examines the influence of this choice.

2.3.2. Seven Explicit Constraint Families

For cooperative landing, all UAVs should reach their initial approach points at a common time. All UAVs in the swarm are assumed to be of the same type, with minimum speed $v_{\min}$ and maximum speed $v_{\max}$. For a given candidate, the earliest common arrival time and constant path speeds are

$$t_c = \max_i L_i/v_{\max}, \qquad v_i = L_i/t_c, \qquad e_{T,i} = [v_{\min}t_c - L_i]_+, \qquad [a]_+ = \max(a, 0). \quad (6)$$

Thus, $e_{T,i} = 0$ for every UAV is equivalent to overlapping arrival windows $[L_i/v_{\max}, L_i/v_{\min}]$. Because $t_c \geq L_i/v_{\max}$ by definition, a nonzero $e_{T,i}$ is the path length that the i-th UAV lacks to arrive at t_c even at the minimum speed; expressed in this way, it depends only on the ratio $v_{\max}/v_{\min}$ and not on the speed scale. When the landing slots are at different distances from the UAVs, this constraint forces the UAVs with nearby slots to lengthen their paths so that the swarm can land together. A candidate requiring a speed below $v_{\min}$ is marked infeasible. At normalized time $\tau = t/t_c$, positions are interpolated at arc-length fraction τ.

Let $d_{ik}(\tau)$ be pairwise distance. Define $\gamma_i(u) = \operatorname{atan2}(z_i', \sqrt{x_i'^2 + y_i'^2})$ and

$$\begin{aligned} \kappa_{h,i}(u) &= \frac{|x_i'y_i'' - y_i'x_i''|}{(x_i'^2 + y_i'^2)^{3/2}}, \\ R_{\min} &= \frac{v_{\max}^2}{g \tan \phi_{\max}}. \end{aligned} \quad (7)$$

The turn formula uses consistent SI units before conversion to km. Primes are analytic B-spline derivatives, evaluated from derivative control points to avoid cancellation from absolute coordinates. A vanishing horizontal tangent is treated as invalid rather than assigned zero curvature.

The following dimensionless functions define the seven constraint families; maxima are over all indicated vehicles, threats, pairs, and checking locations:

$$\begin{aligned} g_1 &= \max_i\{L_i/L_{\max,i} - 1\}, \qquad L_{\max,i} = 2d_i, \\ g_2 &= \max_{i,j}\left\{\frac{H_{\min} - P_{ij,z}}{H_{\max} - H_{\min}}, \frac{P_{ij,z} - H_{\max}}{H_{\max} - H_{\min}}\right\}, \\ g_3 &= \max_{i,j,m}\{(R_{2,m} + \epsilon_D - D_{ijm})/R_{2,m}\}, \\ g_4 &= \max_i\{(v_{\min}t_c - L_i)/d_i\}, \\ g_5 &= \max_{i<k,\tau}\{(d_{\min} - d_{ik}(\tau))/d_{\min}, (d_{ik}(\tau) - d_{\max})/d_{\max}\}, \\ g_6 &= \max_{i,u}\{|v_i \sin \gamma_i(u)|/w_{\max} - 1\}, \\ g_7 &= \max_{i,u}\{R_{\min}\kappa_{h,i}(u) - 1\}. \end{aligned} \quad (8)$$

The strictly positive threat clearance is implemented with $\epsilon_D = 10^{-9}$ km. The length cap specifies the mission detour budget. Altitude bounds specify the operating region. Minimum separation provides a safety buffer; maximum separation limits group dispersion.

We adopt $w_{\max} = 1000$ ft/min $= 0.3048$ km/min, the common climb/descent rate used in National Aeronautics and Space Administration (NASA) Ikhana test encounters, adopted as a conservative planning limit [42]. The Federal Aviation Administration (FAA) handbook supports the coordinated-turn relation [43]; $\phi_{\max} = 30^\circ$ is a conservative planning assumption. With $v_{\max} = 80$ m/s, $R_{\min} \simeq 1.1304$ km. These kinematic restrictions act on every candidate path; load-factor and actuator dynamics are outside this formulation.

2.3.3. Constraint Ordering, Discretization, and Verification

For search, violation depths are maximized over locations and then summed over entities:

$$\begin{aligned}
v_1 &= \sum_i [L_i / L_{\max,i} - 1]_+, \\
v_2 &= \sum_i \max_j \frac{[H_{\min} - P_{ij,z}]_+ + [P_{ij,z} - H_{\max}]_+}{H_{\max} - H_{\min}}, \\
v_3 &= \sum_{i,m} \max_j [(R_{2,m} + \epsilon_D - D_{ijm}) / R_{2,m}]_+, \\
v_4 &= \sum_i e_{T,i} / d_i, \\
v_5 &= \sum_{i<k} \Big([1 - \min_\tau d_{ik}(\tau) / d_{\min}]_+ + [\max_\tau d_{ik}(\tau) / d_{\max} - 1]_+ \Big), \\
v_6 &= \sum_i \max_u [|v_i \sin \gamma_i(u)| / w_{\max} - 1]_+, \\
v_7 &= \sum_i [R_{\min} \max_u \kappa_{h,i}(u) - 1]_+, \qquad V = \sum_{k=1}^{7} v_k.
\end{aligned} \tag{9}$$

The physical conditions hold on the checked geometry exactly when $V = 0$. This depth-based measure avoids counting the number of violated sample points. Deb's rules [39] prefer a feasible candidate to an infeasible one, compare F between feasible candidates, and compare V between infeasible candidates. We implement the common scalar ordering

$$J = \begin{cases} F, & V = 0, \\ J_{\text{off}} + V, & V > 0, \end{cases} \qquad J_{\text{off}} = 10^6. \tag{10}$$

All evaluations assert finite non-negative V and $F < J_{\text{off}}$. Equation (10) is order-equivalent to Deb's rules in exact arithmetic; machine-resolution ties can occur for very small differences in V. The least representable number above J_{off} is used if necessary. There is no tunable penalty coefficient multiplying violation.

Optimization uses $S = 401$ path points, $Q = 801$ analytic-derivative points for both vertical rate and curvature, and $T = 201$ time points. Predetermined planning margins tighten Equation (9): replace $L_{\max}$ by $(1 - 10^{-4})L_{\max}$; replace the exclusion numerator's R_2 by $(1 + 0.002)R_2$; use $[(v_{\min} t_c - L_i)/d_i + 10^{-4}]_+$ for each arrival depth; use $1.01 d_{\min}$ and $0.99 d_{\max}$ for separation thresholds while retaining their physical normalization denominators; use $0.999 w_{\max}$ and $0.99/R_{\min}$ for vertical-rate and curvature limits. No altitude margin is added because clipped control points and the convex-hull property bound the entire spline altitude. These margins were fixed using all scenario-0 calibration outcomes before the formal comparisons.

For adjacent time points, write relative displacement as $\boldsymbol{r}(\theta) = \boldsymbol{r}_0 + \theta \Delta \boldsymbol{r}$, $0 \le \theta \le 1$. The minimum separation of the relative straight-line segment is evaluated analytically at

$$\theta_* = \text{clip}_{[0,1]} \left(-\frac{\boldsymbol{r}_0^{\mathsf{T}} \Delta \boldsymbol{r}}{\|\Delta \boldsymbol{r}\|^2} \right), \qquad d_* = \|\boldsymbol{r}_0 + \theta_* \Delta \boldsymbol{r}\|. \tag{11}$$

Zero relative displacement uses its constant distance. Maximum distance occurs at an endpoint. This interval check is applied during both optimization and verification.

Every final solution is independently checked with 2001 path, derivative, and time points against the physical limits without the planning margins. Length, t_c, and speeds are recomputed. All reported feasibility rates use this dense check; reported objective quality retains the frozen 401-point F, so changing the feasibility filter does not change the objective definition. Section 4.11 reports the differences between checks.

The arrival model assumes constant speed along each path; acceleration/deceleration and turn-speed transitions are not modeled.

3. Algorithm Principle and Analysis

3.1. Grey Wolf Optimizer

GWO is a metaheuristic swarm intelligence algorithm inspired by the hunting behavior of grey wolves, and it mimics the social hierarchy and hunting mechanism of a wolf pack. As shown in Figure 3, the pack is divided into four levels: α, β, δ, and ω. The α wolf is the leader and is responsible for hunting decisions; the β wolf forms the second level and assists the leader in decision-making; the δ wolf forms the third level, which follows these decisions while managing the ω wolves at the lowest level.

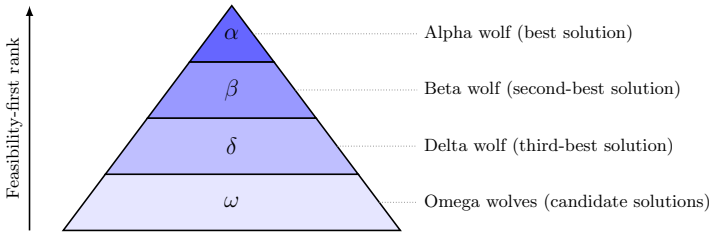


Figure 3. Leader roles under feasibility-first ordering. Standard grey wolf optimizer (GWO) uses one global leader set; multi-strategy improved GWO (MSGWO) applies these roles within each subpopulation, selecting leaders from personal-best positions.

The optimization process of GWO consists of three phases: initializing the positions of the wolves, encircling the prey, and iterative updating. In the initial phase, the positions of the three leading wolves are denoted as $\boldsymbol{X}_\alpha$, $\boldsymbol{X}_\beta$, and $\boldsymbol{X}_\delta$, and the position of the prey is denoted as $\boldsymbol{X}_p$. The encircling behavior is modeled as

$$\boldsymbol{X}_i(t+1) = \boldsymbol{X}_p(t) - \boldsymbol{A}_i \cdot \boldsymbol{D}_i, \tag{12}$$

where $\boldsymbol{X}_i(t+1)$ is the position of the i-th wolf at iteration $t+1$, $\boldsymbol{X}_p(t)$ is the position of the prey at iteration t, and $\boldsymbol{D}_i$ is the distance between the i-th wolf and the prey:

$$\boldsymbol{D}_i = \left| \boldsymbol{C}_i \cdot \boldsymbol{X}_p(t) - \boldsymbol{X}_i(t) \right|, \tag{13}$$

where $\boldsymbol{X}_i(t)$ is the position of the i-th wolf at iteration t, and $\boldsymbol{A}_i$ and $\boldsymbol{C}_i$ are coefficient vectors given by

$$\boldsymbol{A}_i = 2a \cdot \boldsymbol{r}_1 - a, \tag{14}$$

$$\boldsymbol{C}_i = 2\,\boldsymbol{r}_2, \tag{15}$$

where $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are random vectors uniformly distributed in $[0, 1]$, and a is the control parameter. With T denoting the maximum number of update iterations, a decreases linearly from 2 to 0:

$$a = 2 - \frac{2t}{T}. \tag{16}$$

Finally, the positions of all wolves are updated according to the positions of the α, β, and δ wolves:

$$\boldsymbol{X}_i(t+1) = \frac{\boldsymbol{X}_{i,1}(t) + \boldsymbol{X}_{i,2}(t) + \boldsymbol{X}_{i,3}(t)}{3}, \tag{17}$$

where

$$\begin{cases} \boldsymbol{X}_{i,1}(t) = \boldsymbol{X}_\alpha - \boldsymbol{A}_1 \cdot \boldsymbol{D}_\alpha, & \boldsymbol{D}_\alpha = |\boldsymbol{C}_1 \cdot \boldsymbol{X}_\alpha - \boldsymbol{X}_i(t)|, \\ \boldsymbol{X}_{i,2}(t) = \boldsymbol{X}_\beta - \boldsymbol{A}_2 \cdot \boldsymbol{D}_\beta, & \boldsymbol{D}_\beta = \left|\boldsymbol{C}_2 \cdot \boldsymbol{X}_\beta - \boldsymbol{X}_i(t)\right|, \\ \boldsymbol{X}_{i,3}(t) = \boldsymbol{X}_\delta - \boldsymbol{A}_3 \cdot \boldsymbol{D}_\delta, & \boldsymbol{D}_\delta = |\boldsymbol{C}_3 \cdot \boldsymbol{X}_\delta - \boldsymbol{X}_i(t)|. \end{cases} \tag{18}$$

All vector products in Equations (12)–(18) are elementwise. Our GWO and modified GWO (mGWO) baselines retain the three best historical leader positions using stable sorting. In the budgeted experiments, $T = T_u$ is the number of updates available after counting initialization; $t = 0, \ldots, T-1$.

3.2. Multi-Strategy Improved Grey Wolf Optimizer

In the standard GWO, all wolves are attracted by the same three leaders, the personal best positions of individual wolves are not remembered between iterations, and the exploration ability decreases at a constant rate. In multi-UAV path planning, where the search space is high-dimensional and the paths must satisfy multiple coupled constraints, this can make the population contract quickly around the current leaders and miss better collision-free and time-coordinated paths. MSGWO therefore improves GWO with three strategies: a nonlinear control parameter, the individual memory of PSO, and a multi-population mechanism.

3.2.1. Nonlinear Control Parameter

According to Equations (12) and (13), the coefficient $\boldsymbol{A}$ plays an important role in the iterative process of GWO, and its variation enables the algorithm to switch between global exploration and local exploitation: when $|A| > 1$, the search range is expanded, and when $|A| < 1$, the search range is narrowed. According to Equation (14), $\boldsymbol{A}$ is directly affected by a, which decreases linearly with the iteration number t in the standard GWO, as shown in Equation (16). A linear decrease reduces the exploration ability at a constant rate from the first iteration. Therefore, the nonlinear control parameter of mGWO [44] is adopted in this paper:

$$a(t) = 2\left[1 - \left(\frac{t}{T}\right)^2\right], \tag{19}$$

where T is the maximum number of iterations. Compared with Equation (16), $a(t)$ decreases slowly in the early iterations, which keeps $|A| > 1$ for a larger share of the search and enlarges the global search range, and then decreases rapidly in the late iterations to strengthen local exploitation.

3.2.2. Individual Memory of Particle Swarm Optimization

In the position update of GWO, each individual is guided by the best, second-best, and third-best solutions of the population, namely, $\boldsymbol{X}_\alpha$, $\boldsymbol{X}_\beta$, and $\boldsymbol{X}_\delta$, so that information is exchanged within the population and the optimum is approached. However, GWO ignores the personal best position found by each wolf during the iterations. To improve the performance

of the algorithm, the individual memory idea of PSO is introduced to modify the position update equation so that GWO can retain the personal best information of each individual:

$$\boldsymbol{X}_i(t+1) = \frac{\sum_{j=1}^{3} \lambda_j \boldsymbol{X}_{i,j}(t)}{\sum_{j=1}^{3} \lambda_j} + \eta\, \boldsymbol{r}_4 \odot (\boldsymbol{X}_{i,\text{best}} - \boldsymbol{X}_i(t)), \tag{20}$$

where $\boldsymbol{X}_{i,j}(t)$ $(j = 1,2,3)$ are the leader-guided positions given by Equation (18); λ_j are weighting factors that regulate the influence of the best, second-best, and third-best solutions on the position update; $\boldsymbol{X}_{i,\text{best}}$ is the historical best position of the i-th individual; η is the individual memory factor; $\boldsymbol{r}_4$ is a random vector uniformly distributed in $[0,1]$; and $\odot$ denotes the element-wise product. Following [45], the weights are taken in $[1,3]$, and $\lambda_1 = 3$, $\lambda_2 = 2$, and $\lambda_3 = 1$ are used so that the better leaders have a larger influence. Because the first term is a convex combination of the leader-guided positions, Equation (20) reduces to the standard update of Equation (17) when $\lambda_1 = \lambda_2 = \lambda_3$ and $\eta = 0$. The memory factor is fixed at $\eta = 0.5$; Section 4.8 examines settings in $[0,1]$.

Equation (20) is followed by clipping to the encoding bounds. Personal bests are ranked by Equation (10). Personal bests change only upon strict improvement of J. The global best is retained for output, with no extra elite copying into the population. For reproducibility, the implementation draws an unused random vector $\boldsymbol{r}_3$ before drawing the independent $\boldsymbol{r}_4$ used in Equation (20); it has no mathematical contribution but fixes the random stream.

Personal memory and social ranking have distinct uses. Personal bests provide a persistent pool from which each group's leaders are selected; the displacement toward $\boldsymbol{X}_{i,\text{best}}$ also appears directly in the position update. Thus, setting $\eta = 0$ removes the explicit attraction term, but does not remove personal-best storage or its use in leader selection. We make this distinction explicit in the ablation definitions.

3.2.3. Multi-Population Mechanism

When GWO initializes the population, randomly generated data are usually used as the initial information, and all wolves are then attracted by the same three leaders, which quickly reduces the population diversity. Population diversity is one motivation for improved GWO designs [46]. Therefore, a multi-population mechanism [47] is adopted in this paper. The path represented by each individual is characterized by the vector of its three objective components and seven violation measures $(f_L, f_H, f_M, v_1, \ldots, v_7)$ in Section 2.3, aggregated as defined above. After a logarithmic transformation and standardization, these vectors are clustered by the k-means algorithm into K_s subpopulations, so that individuals whose paths have similar characteristics belong to the same subpopulation. Each subpopulation is guided by its own leaders: $\boldsymbol{X}_\alpha$, $\boldsymbol{X}_\beta$, and $\boldsymbol{X}_\delta$ in Equations (18) and (20) are the three best personal-best positions among the members of the subpopulation (if a subpopulation has fewer than three members, it is completed with the global leaders). The population is re-clustered every T_g iterations, so that good solutions found in one subpopulation are shared with the others through the regrouping. In this way, different subpopulations search around different leaders, population diversity can be maintained, premature convergence can be reduced, and the population is still driven toward better solutions.

At regrouping, each current candidate is described by

$$\boldsymbol{f}_i = (\textstyle\sum_n f_{L,n}, \sum_n f_{H,n}, \sum_n f_{M,n}, v_1, \ldots, v_7). \tag{21}$$

The ten non-negative features are transformed by $\log(1+f)$ and standardized columnwise. Constant columns become zero; an entirely constant feature set produces one group. We use k-means++ initialization and ten Lloyd iterations, with $K_s = 3$ requested groups and

regrouping every $T_g = 5$ updates. Empty groups are skipped. Features come from current candidates, whereas leaders come from the personal-best archive.

In standard GWO, every wolf receives guidance from the same global leader triplet. MSGWO changes this to several locally shared triplets. Wolves in different groups can therefore move toward different archived solutions; regrouping changes group membership and the directions of information exchange. Clustering by performance and violation components is intended to organize similar trade-offs, but it is only one possible grouping mechanism. The random-group control in Section 4.7 tests whether these features add value beyond the existence of multiple leader sets.

The schedule controls displacement magnitude, subpopulations determine the source of social guidance, and personal memory retains individual history. These complementary roles connect exploration scale, information exchange, and accumulated experience within one update. Section 4.7 assesses their contributions through separate removals and diversity histories.

3.2.4. Procedure of MSGWO

The procedure of the proposed MSGWO is summarized in Algorithm 1.

Algorithm 1 Multi-strategy improved grey wolf optimizer (MSGWO).

Require: Optimization problem in Equation (3), evaluation budget B, population size N_p, number of update iterations T, memory factor η, leader weights λ_j, number of subpopulations K_s, regrouping period T_g
Ensure: Best path $\boldsymbol{X}_{\text{gbest}}$

```
Initialize the positions of the N_p wolves randomly within the bounds of Section 2.2
Evaluate every wolf with Equation (10); set X_{i,best} = X_i and the global best X_gbest
Count initialization against B; set T from the remaining budget
Cluster the wolves into K_s subpopulations by their cost-component vectors (Section 3.2.3)
for t = 0 to T − 1 while budget remains do
    if t > 0 and t mod T_g = 0 then
        Re-cluster the wolves into K_s subpopulations
    end if
    Compute a(t) with Equation (19)
    for each nonempty subpopulation do
        Take X_α, X_β, X_δ as the three best personal-best positions of its members; pad from global leaders if needed
        Compute X_{i,1}, X_{i,2}, X_{i,3} of every member with Equations (14), (15), and (18)
        Update every member with Equation (20) and clip it to the variable bounds
    end for
    Evaluate all wolves, or only the remaining budget if the batch is partial
    Update X_gbest from every evaluated candidate and record the batch endpoint
    Update X_{i,best} on strict improvement if another iteration remains
end for
return X_gbest; perform the independent dense check without feeding it back to search
```

3.3. Computational Complexity

Let P be population size, $C = n_w + 2$ control points per UAV, S path points, Q derivative points, T time points, and M threats. With spline bases precomputed, one generation of joint evaluation costs

$$O\Big(P\{N(SC + QC + SM + T\log S) + N^2T\}\Big). \tag{22}$$

The N^2T term reflects pairwise separation. Encoding and position updates add $O(PNC)$; global leader sorting is at most $O(P\log P)$. With $q = 10$ features and I_{km} clustering itera-

tions, regrouping adds $O(I_{km}PK_sq/T_g)$ per generation on average, plus feature standardization and local leader selection. Trajectory storage scales with $PN(S+Q+T)$, while implementations retaining pair distances may require $O(PN^2T)$ additional space. Population changes affect batch efficiency and the number of updates under a fixed evaluation budget. Section Controlled Timing reports controlled timing and direct clustering measurements.

4. Simulation Experiments and Analysis

4.1. Simulation Setup

The parameters of MSGWO were set on a separate scenario generated in the same way; the results reported below are obtained on six further scenarios (Scenarios 1–6) that were not used for setting the parameters.

The synthetic descent configuration uses horizontal start and IAP centers $(5,5)$ and $(23,22)$ km, respectively, with formation radius 3 km. Starts lie on a regular polygon whose first vertex points along the flight axis. Each landing slot is the mirror image of the corresponding start slot with respect to the plane normal to the flight direction, so that the UAV at the front of the formation is assigned the nearest slot and the UAV at the rear the farthest one. Start and IAP altitudes are 2.5 and 1.5 km. The map is $[0,30]\times[0,30]\times[0.5,6]$ km. Speeds are restricted to $[3.6,4.8]$ km/min and pairwise distances to $[1,10]$ km. With ten intermediate nodes per UAV, the five-UAV problem has 150 variables. Geometry and budget were frozen before the formal comparisons under a prespecified pilot solvability protocol; pilot runs are excluded from the results below.

Each scenario contains two ground hemispheres and ten airborne spheres. A candidate threat is placed at a uniformly sampled fraction in $[0.25,0.75]$ of a randomly chosen direct route, with a lateral offset uniform in $[-3,3]$ km. Airborne centers have altitude uniform in $[1.5,2.5]$ km and physical radius uniform in $[0.6,1.0]$ km. Ground centers have zero altitude; their radii equal the candidate's direct-route altitude plus a uniform $[0,0.4]$ km increment. We require every endpoint to be farther than $R_1+1.5$ km from the center and existing threat centers to be at least 0.8 times the sum of their physical radii apart. Placement stops with an error after 10,000 failed attempts for a threat. Threat margins are $\Delta_R=0.5$ km and $\Delta_E=1.5$ km. This geometric rule is common to all seeds. Threats are placed along the straight-line routes so that every UAV must deviate and the swarm must coordinate its detours.

Each algorithm was run 30 times independently on each scenario, with the same 30 random seeds for all algorithms. E1 compares seven algorithms on scenarios 1–6, with seeds 1000–1029 and a budget of 30,000 evaluations per run, including initialization. E2 uses the same scenarios and seeds for six ablations. Sensitivity experiments E3–E5 and scale studies E6/E6s use scenarios 1, 3, and 6 and seeds 1000–1009. E3 covers $K_s\in\{1,2,3,4,5,6\}$ and $T_g\in\{1,2,5,10,20,50\}$; E4 covers $\eta\in\{0,0.25,0.5,0.75,1\}$. E5 compares MSGWO, GWO, and L-SHADE under five weights. E6 fixes 30,000 evaluations at $N\in\{3,5,8,10\}$, while E6s uses $B=6000N$, or 200 evaluations per variable. The radius remains 3 km for these sizes; the start and IAP pair distances meet the specified separation bounds. E7 measures timing, and E8/E9 test coupling. Identical configurations reuse their original results, including $N=5$ scale rows and the default sensitivity settings.

4.2. Comparison Algorithms and Settings

MSGWO was compared with six algorithms: the standard GWO [12]; mGWO [44], i.e., GWO with the nonlinear control parameter of Equation (19); SPSO [10], whose spherical-vector encoding is also used in this paper; I-GWO [37]; L-SHADE [38]; and DMGWO [25]. All seven algorithms use the same path encoding and objective/constraint definitions, subject to the explicitly described decoupling of DMGWO. Joint methods with equal initial

population size share the same initial population for a given seed. L-SHADE has a different initial size, and DMGWO uses independent per-aircraft random substreams.

Table 1 summarizes the settings of all seven algorithms. SPSO was run with the parameters of its original implementation: acceleration coefficients $c_1 = c_2 = 1.5$, a velocity limit of 50% of the variable range, velocity reflection at the bounds, and an inertia weight that starts at 1 and decays geometrically.

Table 1. Algorithm settings fixed before formal comparisons.

Algorithm	Settings and Source
GWO	$P = 30$; linear schedule; historical global leader triplet [12].
mGWO	$P = 30$; quadratic schedule, Equation (19) [44].
SPSO	$P = 30$; $c_1 = c_2 = 1.5$; inertia $w_t = 0.98^{200t/T_u}$, starting at 1; velocity cap half the encoding-box width; velocity reflection and position clipping at bounds [10].
I-GWO	$P = 30$; GWO and dimension-learning hunting candidates both count toward budget; sequential in-place greedy updates, clipped candidates, and radius neighborhoods in the native encoding coordinates [37].
L-SHADE	$P_0 = 200$, $P_{\min} = 4$, memory size $H = 6$, archive ratio 2.6, $p = 0.11$; success-history adaptation, current-to-pbest/1, and linear population reduction [38].
DMGWO	30 wolves per UAV pack; four-guide average including pbest; leader perturbation $R_m = 0.1$, exponential decay constant 5; fixed combination enumeration [25].
MSGWO	$P = 30$, $K_s = 3$, $T_g = 5$, $\eta = 0.5$, $\boldsymbol{\lambda} = (3, 2, 1)$, quadratic schedule; cost-component k-means grouping.

mGWO, modified GWO; SPSO, spherical vector-based particle swarm optimization; I-GWO, improved GWO; L-SHADE, success-history-based adaptive differential evolution with linear population size reduction; DMGWO, disturbance-based modified GWO; UAV, unmanned aerial vehicle.

Only scenario 0 is used for algorithm parameter selection. L-SHADE's P_0 is selected from $\{30, 50, 100, 200\}$ using ten seeds per choice: maximize dense-feasible count, then minimize median planning J, then prefer the smaller population. DMGWO uses $R_m \in \{0.05, 0.1, 0.2\}$ and decay constant $c_d \in \{3, 5, 10\}$, with ten seeds per cell; the declared ordering is dense-feasible count, median planning J, smaller R_m, and smaller c_d. The resulting $P_0 = 200$, $R_m = 0.1$, and $c_d = 5$ remain fixed. The decay constant is distinct from the common arrival time t_c. No formal scenario is used to retune these settings, K_s, T_g, or η.

I-GWO follows dimension-learning hunting and greedy comparison, with Euclidean neighborhoods defined in the shared encoding coordinates. L-SHADE's nominal $18D$ initialization is replaced by the selected P_0 to fit the evaluation budget. Its success weights use improvement in the scalar search key J, including the 10^6 offset when a candidate becomes feasible. Thus, the offset affects L-SHADE's parameter adaptation as well as candidate ordering. The comparisons use these implementations throughout.

DMGWO uses independent single-UAV packs followed by joint evaluation of retained path combinations. Appendix B specifies the shared encoding, leader updates, candidate retention, arrival convention, and evaluation accounting used in this comparison.

All algorithms were implemented in Python 3.11.7 with NumPy 1.26.4 and SciPy 1.11.4, with numerical libraries restricted to one thread per process. E7 runs sequentially in one process. Other measured wall times come from shared-workstation batches.

4.3. Statistical Methods and Interpretation

Feasibility is a paired binary endpoint analyzed by the two-sided exact McNemar test. Feasible objective quality is summarized by both mean/sample standard deviation (SD) and median/Q_1/Q_3, with interquartile range (IQR) = $Q_3 - Q_1$. SD describes dispersion around the mean; IQR is less sensitive to skewness and extremes. Shapiro–Wilk results for every E1/E2 group appear in Table S3; tests for fewer than three observations or constant data are marked missing. Among the per-scenario feasible-F groups, 11 of 28 eligible E1

groups and 25 of 42 E2 groups have raw $p < 0.05$. These diagnostics support the use of nonparametric comparisons alongside both sets of descriptive summaries.

For objective comparisons, only seeds for which both algorithms are densely feasible are retained. We apply two-sided paired Wilcoxon signed-rank tests, omitting zero differences and rounding differences to 12 decimals to avoid numerical tie artifacts. Fewer than six pairs yields no quality-test p value. Difference skewness and sample size are reported in Table S4. The signed-rank location interpretation assumes symmetric paired differences.

Holm correction is applied separately to feasibility and quality, and separately to E1 and E2. Each experiment has 36 planned per-scenario baseline comparisons in each endpoint family; pooled comparisons form a separate six-comparison family. Missing tests retain their planned family slots. Friedman ranks use $(0, F)$ for densely feasible solutions and $(1, V_{\text{dense}})$ otherwise, with average ranks for ties. All-pair rank-difference signed-rank post hoc tests use 126 planned per-scenario comparisons or 21 pooled comparisons; the six scenario-level omnibus tests are also Holm-adjusted. We report p_H for Holm-adjusted values.

For minimization, $A_{12} = \Pr(F_A < F_B) + \frac{1}{2}\Pr(F_A = F_B)$ is calculated from all cross-combinations of observations in the stated samples. We distinguish marginal feasible-F, mutually feasible-F, and Deb-rank effect sizes; A_{12} is not the fraction of wins among paired seeds. Pooled tests combine the six scenarios and supplement the scenario-level results. We also report a seed-blocked pooled Friedman sensitivity analysis.

4.4. Main Comparison: Feasibility and Objective Quality

MSGWO produces 132/180 densely feasible joint solutions (73.3%), significantly more than all six comparators in the pooled exact McNemar tests after Holm adjustment. The corresponding counts are 53 for GWO, 63 for mGWO, 1 for SPSO, 53 for I-GWO, 100 for L-SHADE, and 0 for DMGWO; the largest adjusted p value is 0.0008465, against L-SHADE. Table 2 summarizes the pooled descriptive results and paired tests. This ability to satisfy the coupled constraints is MSGWO's principal advantage. Figure 4 and Table S1 show the results in each scenario.

On mutually feasible pairs, pooled MSGWO objective values are significantly lower than GWO, mGWO, and L-SHADE ($p_H = 2.108 \times 10^{-5}$, 0.01624, and 0.02427), with 45, 51, and 72 pairs, respectively. The conditional A_{12} values against GWO, mGWO, L-SHADE, and I-GWO are 0.6494, 0.6275, 0.6456, and 0.4912. MSGWO also has a smaller descriptive feasible-F SD, 0.2300 compared with L-SHADE's 0.3478, over each method's feasible subset. Figure 5 shows the scenario-level distributions.

Table 2. E1 pooled descriptive results and paired tests. Objective summaries use each method's dense-feasible subset; quality tests use only mutual-feasibility pairs. A dash is missing data, not zero. SD, standard deviation.

Algorithm	Feasible	Median F $[Q_1, Q_3]$	Mean $F \pm$ SD	Pairs	p_H (feas.)	p_H (F)
GWO	53/180	1.362 [1.264, 1.667]	1.464 ± 0.2617	45	2.697×10^{-17}	2.108×10^{-5}
mGWO	63/180	1.357 [1.276, 1.546]	1.419 ± 0.1987	51	1.966×10^{-13}	0.01624
SPSO	1/180	1.587 [1.587, 1.587]	$1.587 \pm$ –	1	3.673×10^{-39}	–
I-GWO	53/180	1.312 [1.211, 1.557]	1.403 ± 0.2278	44	6.011×10^{-17}	1
L-SHADE	100/180	1.512 [1.373, 1.898]	1.613 ± 0.3478	72	0.0008465	0.02427
MSGWO	132/180	1.361 [1.248, 1.624]	1.434 ± 0.23	–	–	–
DMGWO	0/180	– [–, –]	$- \pm -$	0	2.204×10^{-39}	–

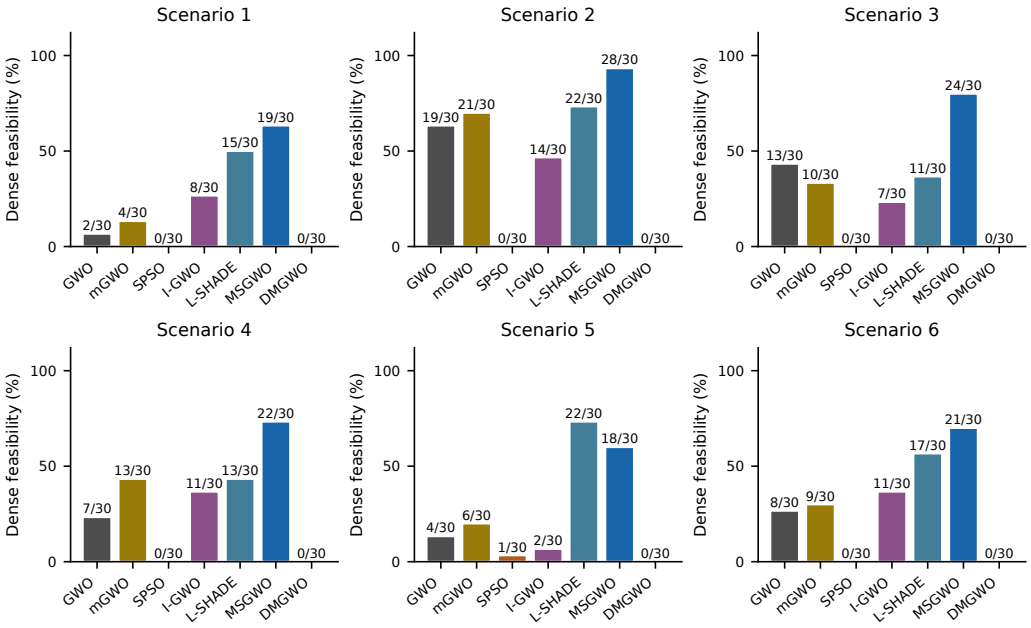


Figure 4. E1 dense-feasible counts and rates, with 30 runs for each algorithm in each fixed scenario.

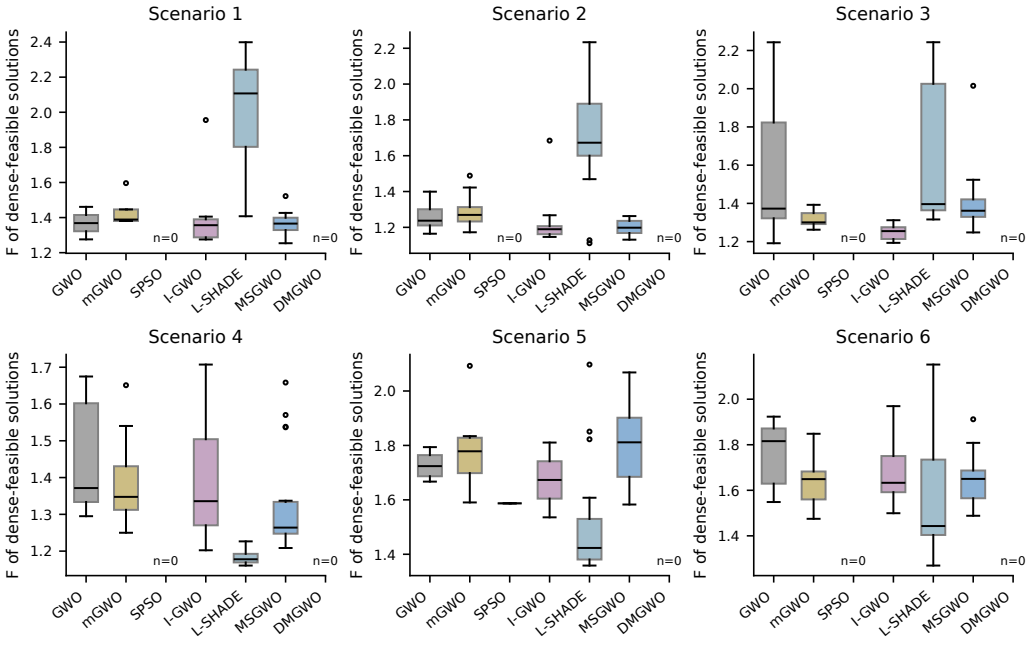


Figure 5. Frozen planning objective F for E1 densely feasible final solutions only. Boxes span Q_1–Q_3, middle lines are medians, whiskers reach the most extreme observations within 1.5 IQR, and open symbols retain outliers. Empty groups have no feasible sample. IQR, interquartile range.

Deb's feasibility-first ranking places MSGWO first overall and in five of six scenarios. Its pooled mean rank is 2.0778, compared with 2.6778 for L-SHADE. The pooled Friedman test ($p = 5.412 \times 10^{-138}$) and seed-blocked sensitivity result ($p = 3.950 \times 10^{-28}$) both detect algorithm differences in the fixed experimental design. Tables S5 and S6 give the ranks and post hoc comparisons.

The quality advantage varies by comparator and scenario. Against I-GWO, the 44 mutually feasible pairs show no significant difference ($p_H = 1$), while SPSO and DMGWO provide insufficient pairs for a quality test. In scenario 5, MSGWO and L-SHADE find feasible solutions in 18/30 and 22/30 runs, respectively, without a significant feasibility difference. L-SHADE obtains lower F on their 13 mutually feasible pairs (MSGWO minus L-SHADE

median $+0.4066$, $p_H = 0.04028$). Overall, the joint feasibility and rank results, together with the pooled quality gains, establish MSGWO's strength in these coupled missions.

4.5. Representative Paths and Arrival Coupling

MSGWO produces smooth, jointly feasible descent paths that accommodate the extra distance needed for common arrival. Figures 6 and 7 show the scenario-1 representative: seed 1003, selected as the lower median frozen F among dense-feasible outcomes. Table 3 connects the geometry to the arrival condition. UAV 1 has the nearest IAP, while the longest route determines t_c; Equation (6) requires $L_1 \geq v_{\min} \max_i L_i / v_{\max}$. Both arrival coordination and obstacle avoidance therefore contribute to its detour.

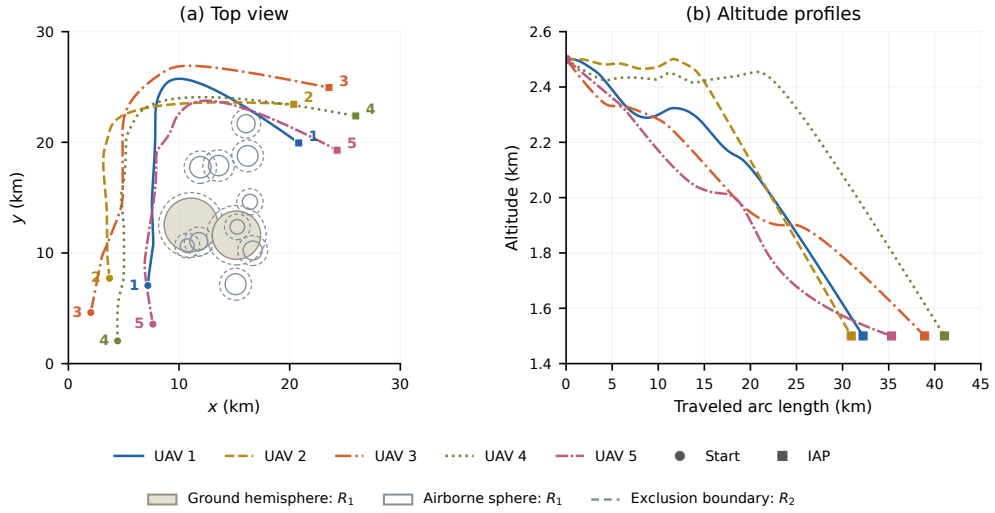


Figure 6. Scenario-1 MSGWO representative paths, seed 1003. (**a**) Top view: filled disks and solid outlines show the projected R_1 boundaries of ground hemispheres and airborne spheres, respectively; dashed circles mark R_2. (**b**) Altitude versus traveled arc length. Colors and line styles identify UAVs; circles and squares mark starts and IAPs (UAV, unmanned aerial vehicle; IAP, initial approach point). Projected path crossings are traversed at different times. Dense verification gives a minimum inter-UAV distance of 1.015 km, satisfying $d_{\min} = 1$ km.

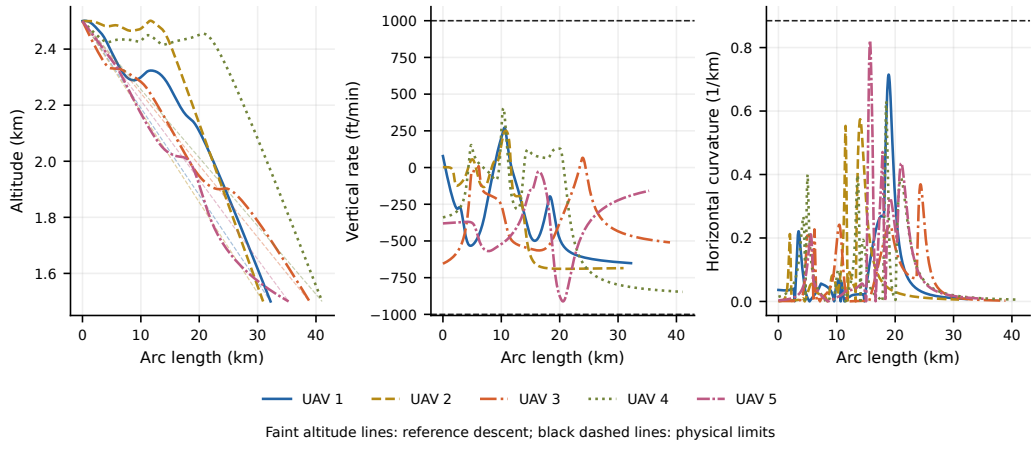


Figure 7. Dense altitude, vertical-rate, and analytic horizontal-curvature profiles of the same representative solution, evaluated at 2001 path/derivative points. The reference descent and physical limits are shown.

Table 3. Stored planning-grid quantities for the representative solution. All required constant speeds lie within $[3.6, 4.8]$ km/min; dense verification also passes.

UAV	d_i (km)	L_i (km)	v_i (km/min)	t_c (min)
1	18.79	32.22	3.767	8.552
2	22.93	30.94	3.618	8.552
3	29.63	38.9	4.548	8.552
4	29.63	41.05	4.8	8.552
5	22.93	35.31	4.129	8.552

The altitude, vertical-rate, and curvature profiles confirm that this representative satisfies the modeled flight limits during its coordinated descent. Figure S1 gives the representative paths for all seven methods and identifies those that fail dense verification.

4.6. Convergence, First Observed Feasibility, and Diversity

The search histories show how feasibility-first search progresses from reducing violation to improving feasible paths. Figure 8 presents the Deb search key, feasible objective values, and diversity. Curves are aligned by cumulative evaluations using the latest retained history at or before each plotting point, without backfilling. Before planning feasibility is observed, a run's feasible F is missing; its median and IQR therefore describe a changing subset and need not be monotone. IQR measures dispersion across runs. The histories retain observations at intervals of at most 300 evaluations for ordinary whole-generation methods and 1135 for L-SHADE in E1.

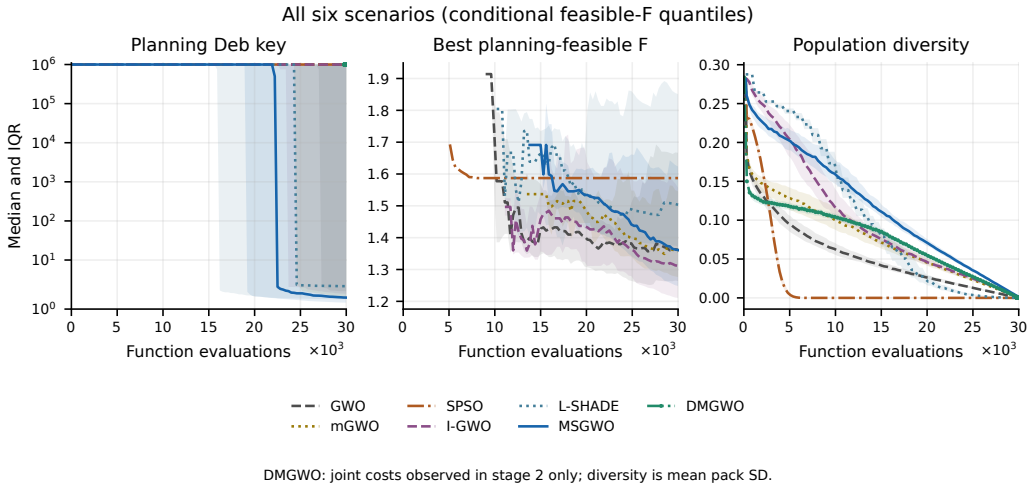


Figure 8. Pooled E1 search histories: medians and IQR from latest retained observations. Feasible F is conditional on having observed planning feasibility by the plotted point; missing values are neither zero-filled nor backfilled. Diversity is mean coordinate SD after normalization by encoding-box width. Shading shows the interquartile range across runs.

Figure 9 shows the cumulative fraction of runs with an observed planning-feasible candidate. First feasibility means the cumulative evaluation count at the first observed batch endpoint containing a planning-feasible candidate. Whole-generation methods usually observe every 30 evaluations, I-GWO evaluates two candidates per wolf, and L-SHADE observes shrinking population batches. The first feasible history entry is retained before thinning. Unsuccessful runs remain in the denominator and are right-censored at the budget. DMGWO observes joint feasibility in its final combination enumeration.

Diversity is measured on current candidates for whole-generation methods, on parent populations for I-GWO and L-SHADE, and as the average across DMGWO's independent

packs. Figure 10 compares the MSGWO variants under a common observation convention, linking the information-flow discussion to the ablation study. Figures S2–S7 give the complete scenario histories, and Figure S8 shows DMGWO's final joint enumeration.

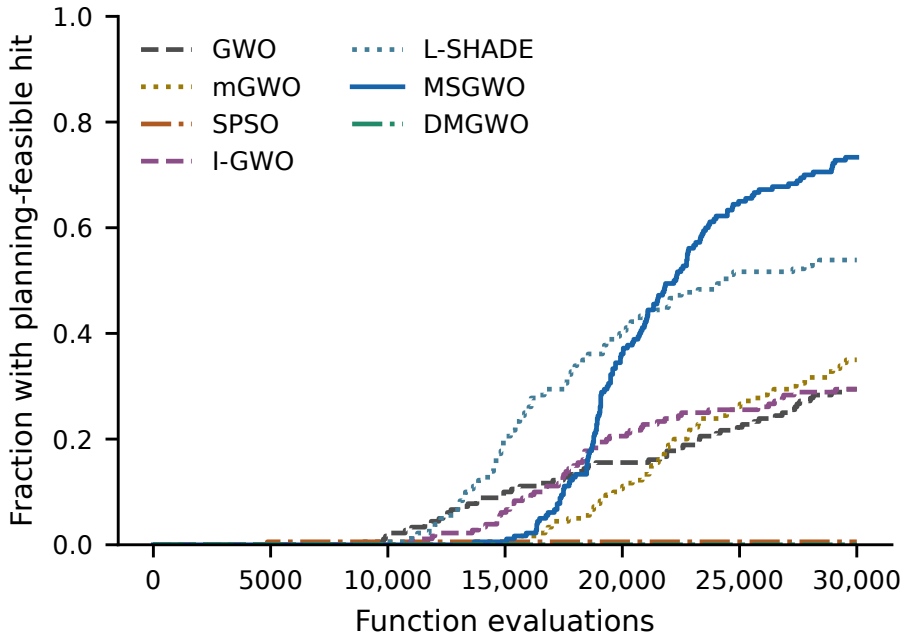


Figure 9. Empirical cumulative fraction with an observed planning-feasible candidate. All 180 runs per method remain in the denominator. Observation times have method-dependent batch resolution and refer to planning, not dense, feasibility. DMGWO joint observations begin only at combination enumeration.

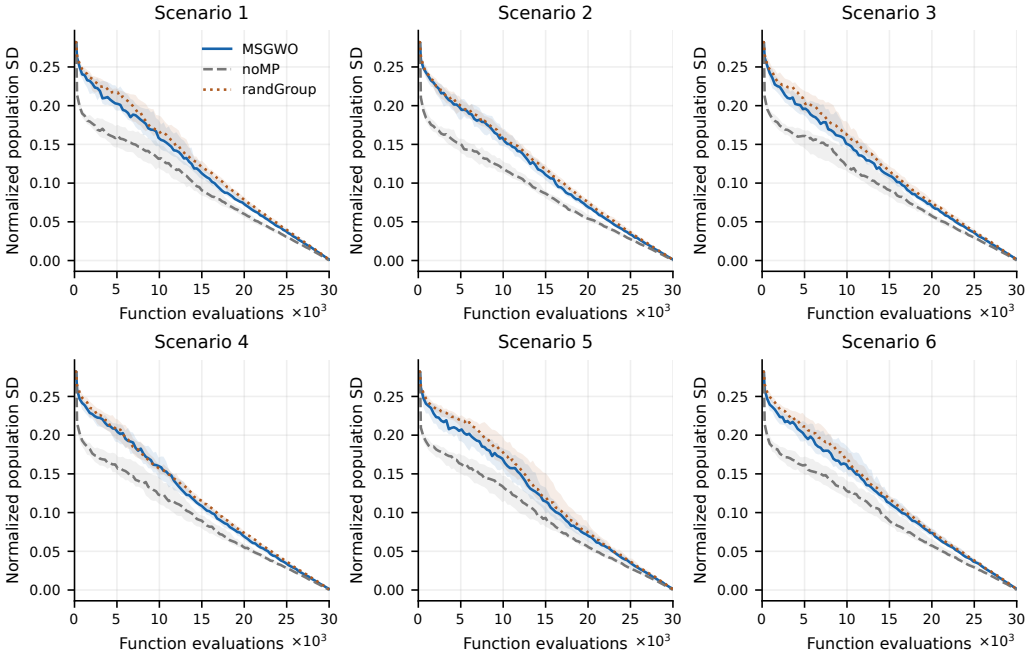


Figure 10. Diversity histories for MSGWO, noMP, and randGroup across all six scenarios. Curves show current-candidate normalized coordinate SD with medians and IQR, using the latest retained observations without backfilling.

4.7. Ablation of the Three Strategies

The ablation results identify the subpopulation structure as the largest contributor and confirm a significant feasibility benefit from the nonlinear schedule. The variants isolate the named components: noNL uses a linear schedule; noPbest sets $\eta = 0$ and retains 3:2:1 leader weights; eqLambda uses equal weights and retains $\eta = 0.5$; noMem combines those two changes; noMP uses one group; and randGroup uses nearly equal random groups at the same regrouping interval. Personal-best archives remain available for leader selection in all variants. Table 4 summarizes the pooled ablation outcomes.

Table 4. E2 pooled ablation outcomes. The full MSGWO rows reuse E1; each variant has 180 runs. Objective quality is conditional on mutual dense feasibility.

Algorithm	Feasible	Median F $[Q_1, Q_3]$	Mean $F \pm$ SD	Pairs	p_H (feas.)	p_H (F)
MSGWO	132/180	1.361 [1.248, 1.624]	1.434 ± 0.23	–	–	–
MSGWO-noNL	106/180	1.365 [1.247, 1.637]	1.453 ± 0.2539	88	0.006495	0.9587
MSGWO-noPbest	121/180	1.352 [1.258, 1.573]	1.42 ± 0.2173	97	0.3851	0.2565
MSGWO-eqLambda	118/180	1.355 [1.255, 1.622]	1.445 ± 0.2493	92	0.3266	0.9587
MSGWO-noMem	114/180	1.339 [1.252, 1.587]	1.429 ± 0.2453	90	0.1423	0.8029
MSGWO-noMP	74/180	1.38 [1.273, 1.643]	1.458 ± 0.2271	61	4.718×10^{-10}	0.01524
MSGWO-randGroup	142/180	1.328 [1.248, 1.566]	1.401 ± 0.2036	112	0.3851	0.01103

Removing subpopulations reduces dense-feasible outcomes from 132 to 74 and significantly worsens mutually feasible F ($p_H = 0.01524$). Removing the nonlinear schedule reduces feasibility to 106 (McNemar raw $p = 0.001299$, $p_H = 0.006495$), with no significant feasible-F change. Multiple leader sets thus provide the largest improvement in both feasibility and conditional path quality, while the nonlinear schedule helps more runs reach feasibility.

The memory and grouping controls clarify which mechanisms account for these gains. Removing the explicit personal-best term or unequal weights individually has no significant feasibility effect after correction. Their joint removal gives 114/180 feasible outcomes (raw McNemar $p = 0.03558$, $p_H = 0.1423$), so a combined memory-update contribution is also not significant after correction. randGroup obtains 142/180 feasible results versus MSGWO's 132/180 ($p_H = 0.3851$); on 112 mutually feasible pairs, it has lower F (MSGWO minus randGroup median $+0.02176$, $p_H = 0.01103$) and a better pooled Deb rank. The evidence therefore supports subpopulations with their own leaders, while feature clustering and random grouping have comparable observed feasibility. Table 5 reports the raw and adjusted feasibility tests, and Table S2 gives all scenario results.

Table 5. Raw and Holm-adjusted pooled feasibility tests for E2, with six planned comparisons.

Variant	Raw McNemar p	Holm p_H
MSGWO-noNL	0.001299	0.006495
MSGWO-noPbest	0.1925	0.3851
MSGWO-eqLambda	0.1089	0.3266
MSGWO-noMem	0.03558	0.1423
MSGWO-noMP	7.864×10^{-11}	4.718×10^{-10}
MSGWO-randGroup	0.2026	0.3851

Together, the ablations explain the main benefit of MSGWO through its multiple sources of guidance and exploration schedule. Cost-component clustering remains the grouping rule in MSGWO; the random-group variant tests the contribution of that choice.

4.8. Parameter and Weight Sensitivity

MSGWO retains its feasibility advantage across the five tested objective-weight choices, while the grouping study shows how search behavior depends on K_s and T_g. The $K_s \times T_g$ study yields 8–28 dense-feasible solutions out of 30 per setting. The default $K_s = 3, T_g = 5$ yields 22/30; the highest observed count is at $K_s = 4, T_g = 5$. With $K_s = 1$, regrouping has no effect and all intervals share the same trajectory. Figure 11 and Tables S7 and S8 report feasibility, objective summaries, runtime, and convergence/diversity at half and full budget for every setting.

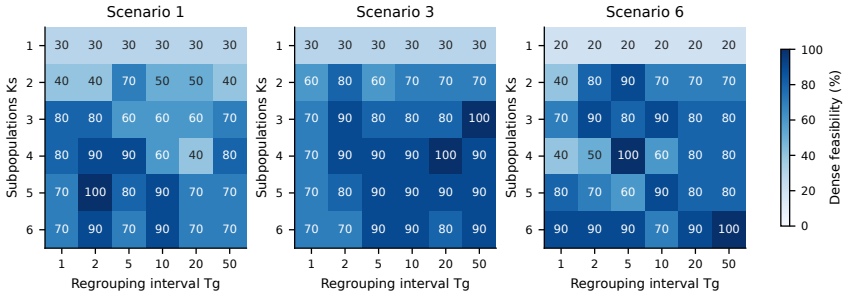


Figure 11. Dense feasibility for all 36 $K_s \times T_g$ settings in each of three fixed scenarios. Each panel reports ten seeds per cell; pooled tables combine 30 runs per setting. Within each scenario, the six $K_s = 1$ cells intentionally reuse one setting.

For $\eta = 0, 0.25, 0.5, 0.75, 1$, the feasible counts are, respectively, 22, 20, 22, 20, and 27 out of 30 (Table S9). The $\eta = 1$ setting has the highest observed count and lower descriptive feasible-F summaries than the default. The comparison algorithms retain the parameters selected before formal testing, including $\eta = 0.5$ for MSGWO.

The five weight triples are default $(1, 1, 4)/6$, equal $(1, 1, 1)/3$, length-focused $(4, 1, 1)/6$, height-focused $(1, 4, 1)/6$, and threat-focused $(0.1, 0.1, 0.8)$. Across these weights, MSGWO, GWO, and L-SHADE, respectively, retain 22, 9, and 9 feasible results out of 30 per weight (Table S10). First planning-feasible observations are unchanged because infeasible candidates are ordered by V. Feasible paths and their component values change once objective quality guides selection; objectives with different weights represent different utilities. The feasibility advantage therefore persists across the tested preferences, while path quality responds to their trade-offs.

4.9. Scale and Computational Time

MSGWO's feasibility lead becomes more pronounced as the swarm grows from three to eight UAVs under dimension-proportional budgets. Its E6s counts are 24, 22, 9, and 0 out of 30 for $N = 3, 5, 8, 10$; the fixed-budget E6 counts are 22, 22, 3, and 0. At $N = 5$ MSGWO has the largest observed count, and at $N = 8$, its 9/30 is three times the highest comparator count of 3/30. Figure 12 shows both budget schemes. Tables S11–S13 provide the objective summaries and paired comparisons, and Figures S9 and S10 give search histories and scale timings.

At $N = 3$, MSGWO's E6s feasibility count is not significantly different from I-GWO's 21/30 or L-SHADE's 23/30. At $N = 10$, no algorithm finds a dense-feasible solution under either budget. With 60,000 evaluations, MSGWO has the best Deb mean rank, 1.3, indicating smaller constraint violations. Among the 210 E6s runs at this size, arrival and separation violations occur in 208 and 185 runs, respectively, with overlapping categories. These finite-budget failures identify a remaining search difficulty as pairwise interactions grow quadratically. E6s applies new budgets and schedules to each run, rather than continuing E6 trajectories.

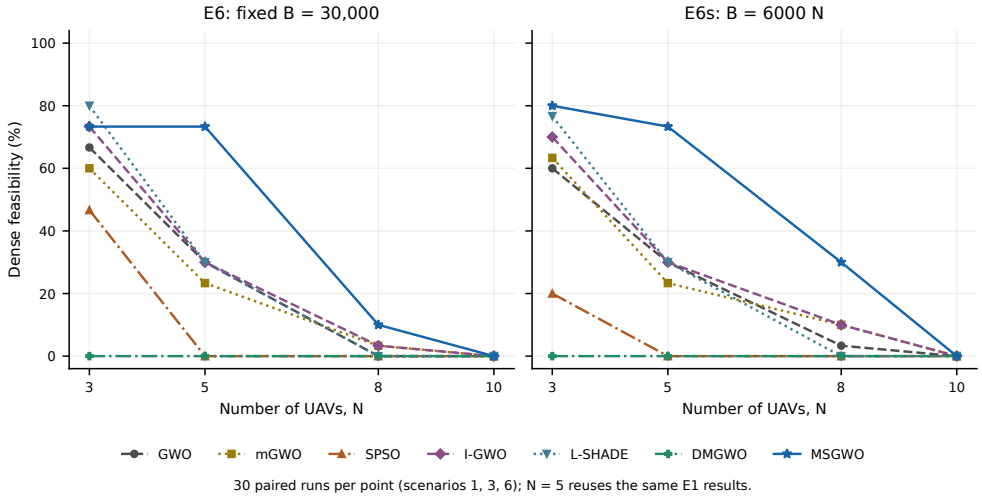


Figure 12. Dense feasibility for fixed-budget E6 and dimension-proportional E6s ($B = 6000N$). Each algorithm/size has 30 runs. $N = 5$ uses the same E1 subset in both schemes. Objective summaries appear in Tables S11 and S12.

Controlled Timing

The median directly measured clustering fraction remains below one percent of MSGWO's optimization time across the tested configurations. E7 uses 3000 evaluations per run, $N \in \{3, 5, 8, 10\}$, $P \in \{30, 60, 120\}$, and three seeds for GWO and MSGWO in sequential single-process measurements. Timing includes candidate evaluation, updates, and history/diversity recording, and excludes final reporting and dense verification. Across the twelve (N, P) cells, the median MSGWO/GWO wall-time ratio ranges from 0.942 to 1.111, while the median `kmeans2` fraction ranges from 0.048% to 0.735%. Other costs include feature preparation, local leader selection, and memory updates. Figure 13 and Table S14 show that the multi-strategy search has wall times close to GWO in these controlled measurements.

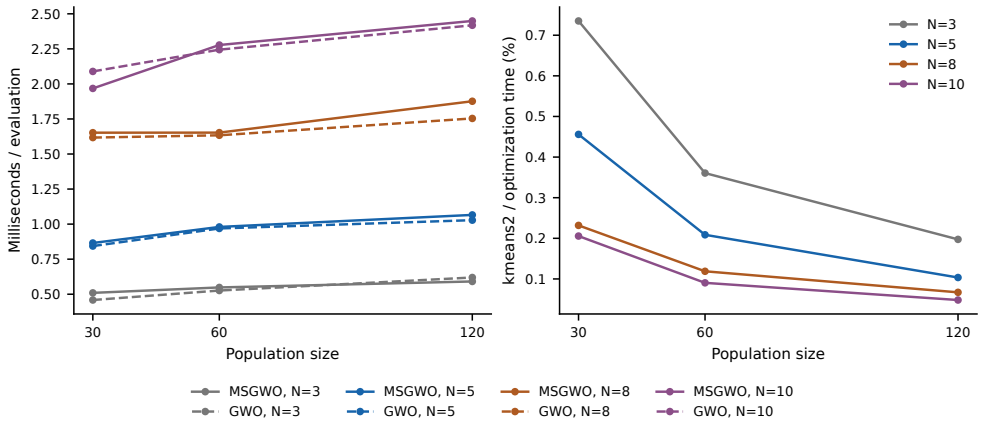


Figure 13. E7 sequential single-process time per evaluation and direct `kmeans2` time fraction over UAV and population sizes. The clustering fraction excludes feature preparation and leader selection. Each cell uses three runs.

4.10. Coupling Controls

The coupling controls show why these missions require joint treatment of the paths. E8 independently plans each of the five paths with MSGWO using 6000 evaluations per aircraft. Although 899/900 single-UAV subproblems pass dense verification, none of the assembled solutions satisfies all joint constraints (0/180). Arrival passes in 55/180 assemblies and

separation in 0/180. Successful individual planning alone therefore does not provide the coordination required by this setting.

Under the present strongly coupled setting and the adaptation conventions in Appendix B, the published two-stage DMGWO method also yields 0/180 feasible E1 solutions. Separation fails in 180 runs, arrival in 130, and threat exclusion in 4, with overlapping categories (Table S15). Its independent first stage and subsequent combination selection leave the joint constraints unresolved in these experiments.

E9 removes only the arrival term while preserving the time-position and separation conventions. It produces 144/180 feasible outcomes for the reduced problem, but only 7/180 pass the full model. Arrival fails in 173 runs; the median longest-to-shortest path ratio is 1.5466, above the permitted speed ratio 4/3. A separation failure can involve either the lower safety bound or the upper dispersion bound. Together, E8, DMGWO, and E9 demonstrate the importance of optimizing common arrival and group separation together with individual path quality.

4.11. Planning and Dense Verification Differences

Every final solution passing the full-model check on the planning grid with margins also passes dense verification. Table 6 counts each physical record once. In E1, three L-SHADE outcomes additionally pass dense verification despite failing the stricter planning check. These reverse differences include the effect of the planning margins. The physical coarse-grid check without margins also has zero coarse-only feasible outcomes. This agreement supports the selected discretization and margins for the tested paths.

Table 6. Full-model verification agreement for physical records only. "Planning only" means feasible with planning grids/margins but infeasible densely; "Dense only" is the reverse. "Physical coarse only" removes planning margins before comparison. E2 excludes its reused MSGWO rows; E6/E6s exclude reused $N = 5$ rows. E7 is the short timing experiment, and E9 uses full-model verification rather than its reduced search flag.

Experiment	Runs	Planning	Dense	Planning Only	Dense Only	Physical Coarse Only
E1	1260	399	402	0	3	0
E2	1080	675	675	0	0	0
E3	900	668	669	0	1	0
E4	90	67	67	0	0	0
E5	360	156	160	0	4	0
E6	630	124	125	0	1	0
E7	72	9	9	0	0	0
E8	180	0	0	0	0	0
E9	180	7	7	0	0	0
E6s	630	127	127	0	0	0

5. Discussion

MSGWO's main strength is its ability to find jointly feasible paths under coupled arrival, separation, and flight limits. Its 73.3% feasibility rate significantly exceeds all six comparators, and its Friedman mean rank is best overall and in five of six scenarios. Lower objective values on mutually feasible runs against GWO, mGWO, and L-SHADE complement this feasibility advantage. The coupling controls explain its practical relevance: individually feasible paths still require coordinated adjustment to satisfy the group constraints. The largest ablation loss follows removal of subpopulations, supporting the role of multiple leader sets, while the nonlinear schedule also makes a significant contribution to feasibility.

MSGWO and I-GWO show no significant difference in pooled mutually feasible objective quality, while L-SHADE obtains lower costs in scenario 5. Random grouping has comparable observed feasibility and better conditional quality; individual and combined memory-update removals have no significant feasibility effect after Holm correction. The sensitivity study favors $\eta = 1$ descriptively. At $N = 3$, feasibility differences from I-GWO and L-SHADE are not significant; at $N = 10$, all methods fail to find a feasible solution.

The present study addresses path planning under static known threats. Future work will examine time-dependent threats and online replanning, together with speed scheduling and closed-loop tracking.

6. Conclusions

This paper addressed the cooperative landing path planning of a UAV swarm in environments with multiple threats. An explicit constrained joint planning model was established that combines dimensionless path length, descent-profile deviation, and threat exposure, with seven constraint families including arrival-time coordination, inter-UAV separation, vertical rate, and turning radius on the paths, which are encoded by spherical vectors and smoothed by B-splines. To solve this model, a multi-strategy improved grey wolf optimizer, MSGWO, was proposed, which combines a multi-population mechanism based on the clustering of cost components, the nonlinear control parameter of mGWO, and the individual memory of PSO. The main conclusions are as follows:

1. In six fixed five-UAV scenarios, MSGWO finds 132/180 densely feasible solutions (73.3%), significantly more than all six comparators after Holm correction, and achieves the best Friedman mean rank overall and in five scenarios. Its objective values are also significantly lower than those of GWO, mGWO, and L-SHADE on pooled mutually feasible runs.
2. Independent per-UAV planning and adapted DMGWO find no feasible joint solution in this setting, demonstrating the need to address arrival and separation together.
3. With dimension-proportional budgets, MSGWO's advantage becomes more pronounced up to eight UAVs, where it finds 9 feasible solutions out of 30 compared with at most 3 for any comparator.
4. Ablations identify the subpopulation structure as the largest contributor and confirm the nonlinear schedule's feasibility benefit.

Future work will extend the joint planning framework to larger swarms, dynamic threats, and closed-loop execution.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/drones10110795/s1. The accompanying Supplementary Materials contain: Table S1, complete E1 results; Table S2, complete E2 results; Table S3, Shapiro–Wilk diagnostics; Table S4, paired objective diagnostics; Table S5, E1 Friedman ranks; Table S6, E1/E2 rank post hoc comparisons; Table S7, grouping sensitivity; Table S8, grouping convergence and diversity snapshots; Table S9, personal-best coefficient sensitivity; Table S10, objective-weight sensitivity; Table S11, fixed-budget scale results; Table S12, dimension-proportional scale results; Table S13, paired scale comparisons; Table S14, controlled timing; Table S15, DMGWO constraint violations. Figure S1 shows all representative paths; Figures S2–S7 show convergence in scenarios 1–6; Figure S8 shows DMGWO combination enumeration; Figure S9 shows fixed-budget scale histories; Figure S10 shows dimension-proportional wall and CPU times.

Author Contributions: Conceptualization, Z.Z. and Y.P.; methodology, Y.W. and Z.Z.; software, Y.W. and Z.Z.; validation, Y.W.; formal analysis, Y.W.; investigation, Y.W. and Z.Z.; writing—original draft preparation, Y.W. and Z.Z.; writing—review and editing, Y.P. and X.W.; visualization, Y.W.; supervision, Y.P. and X.W.; project administration, Y.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 52202441; the Aeronautical Science Foundation of China, grant number 20240013052003; and the Defense Industrial Technology Development Program, grant numbers JCKY2024605C001 and JCKY2025605C011.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data supporting the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

3D	Three-dimensional
GWO	Grey wolf optimizer
mGWO	Modified grey wolf optimizer
MSGWO	Multi-strategy improved grey wolf optimizer
PSO	Particle swarm optimization
SPSO	Spherical vector-based particle swarm optimization
UAV	Unmanned aerial vehicle

Appendix A. Statistical Diagnostic Summary

Shapiro–Wilk diagnostics identify departures from normality in the per-scenario feasible-objective samples: 11 of 28 eligible E1 groups and 25 of 42 eligible E2 groups have raw $p < 0.05$. Tests are undefined for fewer than three observations or constant data. These diagnostics, together with paired-difference skewness, motivate reporting both mean/SD and median/quartiles alongside the paired comparisons in Section 4.3. Table S3 gives every group, and Table S4 reports paired sample sizes, differences, skewness, and effect sizes. Complete scenario results and rank comparisons appear in Tables S1, S2, S5 and S6.

Appendix B. DMGWO Adaptation for Joint Landing Constraints

DMGWO first solves each UAV's single-vehicle problem with the arrival and separation terms disabled and speed set to $v_{\max}$. Its four guided moves (three historical leaders and pbest) are equally averaged. At generation k out of G, its schedule is $a = 2[1 - (k/G)^2]$ for $k/G < 0.5$ and $a = 3(1 - k/G)$ otherwise. Each leader is perturbed by $R_m(\boldsymbol{u} - \boldsymbol{l}) \odot (\boldsymbol{r} - 0.5) \exp(-c_d k/G)$; only strict improvements enter the leader archive. The final candidate paths are combined and evaluated with all seven joint constraints, using Equation (6), without discarding incompatible UAVs or applying joint repair. We use the shared spherical/B-spline representation instead of the source paper's coordinate transformation. The implementation uses stable historical ranking, distinct decision vectors in the archive, and clipping at the encoding bounds.

Let q_c be the number of final paths retained per UAV and $Q_c = q_c^N$. DMGWO counts one single-UAV evaluation as $1/N$ of a complete evaluation; each pack receives

$b = B - Q_c$, giving $Nb/N + Q_c = B$. Initialization, ordinary moves, and all evaluated leader perturbations count. We use $q_c = 3$ for $N = 3, 5, 8$ and $q_c = 2$ for $N = 10$, fixed beforehand because $3^{10} > 30,000$. At $N = 5$, this means 29,757 single-UAV evaluations per pack and 243 complete combinations. Joint evaluation includes quadratic pairwise checks. E8 uses 6000 actual evaluations per aircraft.

References

1. Pan, Z.; Zhang, C.; Xia, Y.; Xiong, H.; Shao, X. An Improved Artificial Potential Field Method for Path Planning and Formation Control of the Multi-UAV Systems. *IEEE Trans. Circuits Syst. II Express Briefs* **2022**, *69*, 1129–1133. [CrossRef]
2. Hao, G.; Lv, Q.; Huang, Z.; Zhao, H.; Chen, W. UAV Path Planning Based on Improved Artificial Potential Field Method. *Aerospace* **2023**, *10*, 562. [CrossRef]
3. Wang, Z.; Gong, H.; Nie, M.; Liu, X. Research on Multi-UAV Cooperative Dynamic Path Planning Algorithm Based on Conflict Search. *Drones* **2024**, *8*, 274. [CrossRef]
4. Guo, J.; Xia, W.; Hu, X.; Ma, H. Feedback RRT* Algorithm for UAV Path Planning in a Hostile Environment. *Comput. Ind. Eng.* **2022**, *174*, 108771. [CrossRef]
5. Wang, Z.; Sun, G.; Zhou, K.; Zhu, L. A Parallel Particle Swarm Optimization and Enhanced Sparrow Search Algorithm for Unmanned Aerial Vehicle Path Planning. *Heliyon* **2023**, *9*, e14784. [CrossRef] [PubMed]
6. Bai, W.; Jia, X.; Lü, T. Application of Improved Sparrow Search Algorithms in Three-Dimensional Path Planning. *Control Eng. China* **2022**, *29*, 1800–1809. (In Chinese) [CrossRef]
7. Bao, Y.Y.; Liu, Y.; Wang, J.S.; Liu, J.X. Genetic Algorithm Based on Grid Maps for Solving Robot Path Planning Problem. *Eng. Lett.* **2023**, *31*, 1635–1648.
8. Li, G.; Liu, C.; Wu, L.; Xiao, W. A Mixing Algorithm of ACO and ABC for Solving Path Planning of Mobile Robot. *Appl. Soft Comput.* **2023**, *148*, 110868. [CrossRef]
9. Meng, Q.; Chen, K.; Qu, Q. PPSwarm: Multi-UAV Path Planning Based on Hybrid PSO in Complex Scenarios. *Drones* **2024**, *8*, 192. [CrossRef]
10. Phung, M.D.; Ha, Q.P. Safety-Enhanced UAV Path Planning with Spherical Vector-Based Particle Swarm Optimization. *Appl. Soft Comput.* **2021**, *107*, 107376. [CrossRef]
11. Liu, Y.; Zhang, H.; Zheng, H.; Li, Q.; Tian, Q. A Spherical Vector-Based Adaptive Evolutionary Particle Swarm Optimization for UAV Path Planning under Threat Conditions. *Sci. Rep.* **2025**, *15*, 2116. [CrossRef]
12. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
13. Yu, F. Research on Spectrum Sensing and Spectrum Allocation Based on Swarm Intelligence Optimization Algorithm. Master's Thesis, Nanjing University of Posts and Telecommunications, Nanjing, China, 2023. (In Chinese)
14. Emary, E.; Zawbaa, H.M.; Grosan, C.; Hassanien, A.E. Feature Subset Selection Approach by Gray-Wolf Optimization. In *Advances in Intelligent Systems and Computing, Proceedings of the 1st International Afro-European Conference for Industrial Advancement (AECIA 2014), Addis Ababa, Ethiopia, 17–19 November 2014;* Springer International Publishing: Cham, Switzerland, 2015; Volume 334, pp. 1–13. [CrossRef]
15. Mirjalili, S. How Effective Is the Grey Wolf Optimizer in Training Multi-Layer Perceptrons. *Appl. Intell.* **2015**, *43*, 150–161. [CrossRef]
16. Xu, D.; Ding, S. Research on Improved GWO-Optimized SVM-Based Short-Term Load Prediction for Cloud Computing. *Comput. Eng. Appl.* **2017**, *53*, 68–73. (In Chinese) [CrossRef]
17. Xu, L.; Zhao, W. UAV Track Planning Based on New Gray Wolf Optimization Algorithm. *Electron. Meas. Technol.* **2022**, *45*, 55–61. (In Chinese) [CrossRef]
18. Sun, Y.; Lv, B.; Yang, H.; Li, X. Multi-UAV Trajectory Planning Based on Improved Multi-Population Grey Wolf Optimizer Algorithm. In Proceedings of the 2024 36th Chinese Control and Decision Conference (CCDC), Xi'an, China, 25–27 May 2024; pp. 6142–6148. [CrossRef]
19. Cao, J.; Zhang, G.; Xu, P. A* Initialized Mutable Gray Wolf Optimizer for UAV Path Planning. *Comput. Eng. Appl.* **2022**, *58*, 275–282. (In Chinese) [CrossRef]
20. Xu, C.; Li, C.; Yu, X.; Huang, Q. Improved Grey Wolf Optimization Algorithm Based on Chaotic Cat Mapping and Gaussian Mutation. *Comput. Eng. Appl.* **2017**, *53*, 1–9. (In Chinese) [CrossRef]
21. Zhang, S.; Wang, L. Multi-UAV Cooperative Path Planning Based on Chaotic Grey Wolf Optimiser. *Aeronaut. J.* **2026**, *130*, 741–761. [CrossRef]
22. Zheng, Z.; Huang, H.; Li, C.; Yu, Y.; Wang, X.; Cai, J.; Huang, X.; Hu, S. A Hybrid Search Behavior-Based Adaptive Grey Wolf Optimizer for Cooperative Path Planning for Multiple UAVs. *Sensors* **2025**, *25*, 7657. [CrossRef] [PubMed]

23. Zhou, Z.; Guo, Y.; Wang, Y.; Lyu, J.; Gong, H.; Ye, X.; Li, Y. Multi-UAV Trajectory Optimization Under Dynamic Threats: An Enhanced GWO Algorithm Integrating a Priori and Real-Time Data. *Int. J. Comput. Intell. Syst.* **2025**, *18*, 140. [CrossRef]
24. Gai, W.; Zheng, Y.; Zhang, J.; Zhang, G. A Novel Leader-Follower-Based Hybrid Particle Swarm-Grey Wolf Optimizer Algorithm for the Constrained UAV Path Planning. *Aircr. Eng. Aerosp. Technol.* **2025**, *97*, 636–647. [CrossRef]
25. Shahid, S.; Zhen, Z.; Javaid, U. Multi-UAV Path Planning Using DMGWO Ensuring 4D Collision Avoidance and Simultaneous Arrival. *Aircr. Eng. Aerosp. Technol.* **2024**, *96*, 1117–1127. [CrossRef]
26. Wu, D.; Hou, Y. Neighborhood Elite Learning Multi-Strategy Cooperative Particle Swarm Optimization for Coordinated Simultaneous Arrival Path Planning of Multiple Amphibious UAVs. *J. King Saud Univ. Comput. Inf. Sci.* **2026**, *38*, 844. [CrossRef]
27. Meng, Q.; Qu, Q.; Chen, K.; Yi, T. Multi-UAV Path Planning Based on Cooperative Co-Evolutionary Algorithms with Adaptive Decision Variable Selection. *Drones* **2024**, *8*, 435. [CrossRef]
28. Guo, J.; Gan, M.; Hu, K. Cooperative Path Planning for Multi-UAVs with Time-Varying Communication and Energy Consumption Constraints. *Drones* **2024**, *8*, 654. [CrossRef]
29. Nguyen, L.V.; Kwok, N.M.; Ha, Q.P. Fermat-Weber location particle swarm optimization for cooperative path planning of unmanned aerial vehicles. *Appl. Soft Comput.* **2024**, *167*, 112269. [CrossRef]
30. Shen, Y.; Liu, X.; Ma, X.; Du, H.; Xin, L. A Bionic Social Learning Strategy Pigeon-Inspired Optimization for Multi-Unmanned Aerial Vehicle Cooperative Path Planning. *Appl. Sci.* **2025**, *15*, 910. [CrossRef]
31. Meng, K.; Wu, B.; Xin, B.; Deng, F.; Chen, C. Multiobjective multi-UAV path planning via evolutionary multitasking optimization with adaptive operator selection and knowledge fusion. *Swarm Evol. Comput.* **2025**, *99*, 102145. [CrossRef]
32. Jin, Z.; Li, H.; Qin, Z.; Wang, Z. Gradient-Free Cooperative Source-Seeking of Quadrotor Under Disturbances and Communication Constraints. *IEEE Trans. Ind. Electron.* **2025**, *72*, 1969–1979. [CrossRef]
33. Xing, X.; Zhou, Z.; Li, Y.; Xiao, B.; Xun, Y. Multi-UAV Adaptive Cooperative Formation Trajectory Planning Based on an Improved MATD3 Algorithm of Deep Reinforcement Learning. *IEEE Trans. Veh. Technol.* **2024**, *73*, 12484–12499. [CrossRef]
34. Liu, H.; Long, X.; Li, Y.; Yan, J.; Li, M.; Chen, C.; Gu, F.; Pu, H.; Luo, J. Adaptive multi-UAV cooperative path planning based on novel rotation artificial potential fields. *Knowl.-Based Syst.* **2025**, *317*, 113429. [CrossRef]
35. Wu, C.; Guo, Z.; Zhang, J.; Mao, K.; Luo, D. Cooperative Path Planning for Multiple UAVs Based on APF B-RRT* Algorithm. *Drones* **2025**, *9*, 177. [CrossRef]
36. Yan, Y.; Liu, S.; Hao, R. RRT*-APF Path Planning and MA-AADRC-SMC Control for Cooperative 3-D Obstacle Avoidance in Multi-UAV Formations. *Drones* **2025**, *9*, 611. [CrossRef]
37. Nadimi-Shahraki, M.H.; Taghian, S.; Mirjalili, S. An improved grey wolf optimizer for solving engineering problems. *Expert Syst. Appl.* **2021**, *166*, 113917. [CrossRef]
38. Tanabe, R.; Fukunaga, A.S. Improving the search performance of SHADE using linear population size reduction. In Proceedings of the 2014 IEEE Congress on Evolutionary Computation (CEC), Beijing, China, 6–11 July 2014; pp. 1658–1665. [CrossRef]
39. Deb, K. An efficient constraint handling method for genetic algorithms. *Comput. Methods Appl. Mech. Eng.* **2000**, *186*, 311–338. [CrossRef]
40. Tong, K.O.; Schoemig, E.G.; Boyle, D.A.; Scharl, J.; Haraldsdottir, A. Descent profile options for continuous descent arrival procedures within 3d path concept. In Proceedings of the 2007 IEEE/AIAA 26th Digital Avionics Systems Conference, Dallas, TX, USA, 21–25 October 2007; pp. 3.A.3-1–3.A.3-11. [CrossRef]
41. Alharbi, E.A.; Abdel-Malek, L.L.; Milne, R.J.; Wali, A.M. Analytical Model for Enhancing the Adoptability of Continuous Descent Approach at Airports. *Appl. Sci.* **2022**, *12*, 1506. [CrossRef]
42. Marston, M.; Valkov, S.; Flock, A. *UAS-NAS NASA 870 Ikhana UAS No Chase COA (NCC) Flights, Flight Test Report*; Technical Report AFRC-E-DAA-TN60378; NASA Armstrong Flight Research Center: Edwards, CA, USA, 2018.
43. Federal Aviation Administration. *Pilot's Handbook of Aeronautical Knowledge*; Chapter 5: Aerodynamics of Flight; U.S. Department of Transportation: Washington, DC, USA, 2023.
44. Mittal, N.; Singh, U.; Sohi, B.S. Modified Grey Wolf Optimizer for Global Engineering Optimization. *Appl. Comput. Intell. Soft Comput.* **2016**, *2016*, 7950348. [CrossRef]
45. Wang, Y.; Jiang, X. Robot Path Planning Using a Hybrid Grey Wolf Optimization Algorithm. *Comput. Eng. Sci.* **2020**, *42*, 1294–1301. (In Chinese) [CrossRef]
46. Liu, Z.; He, L.; Yuan, L.; Zhang, H. Path Planning of Mobile Robot Based on TGWO Algorithm. *J. Xi'an Jiaotong Univ.* **2022**, *56*, 49–60. (In Chinese)
47. Jia, D.; Vagners, J. Parallel Evolutionary Algorithms for UAV Path Planning. In Proceedings of the AIAA 1st Intelligent Systems Technical Conference, Chicago, IL, USA, 20–22 September 2004. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.