Journal of *Marine Science and Engineering*

MDPI

Article

Three-Dimensional Path Planning of Deep-Sea Mining Vehicle Based on Improved Particle Swarm Optimization

Changyu Lu [1,2,3], Jianmin Yang [1,2,*], Bernt Johan Leira [3], Qihang Chen [1,2] and Shulin Wang [1,2]

[1] State Key Laboratory of Ocean Engineering, Shanghai Jiao Tong University, 800 Dongchuan Road, Minhang District, Shanghai 200240, China
[2] Yazhou Bay Institute of Deepsea SCI-TECH, Shanghai Jiao Tong University, Sanya 572000, China
[3] Department of Marine Technology, Norwegian University of Science and Technology, 7049 Trondheim, Norway; bernt.leira@ntnu.no
* Correspondence: jmyang@sjtu.edu.cn

Abstract: Three-dimensional path planning is instrumental in path decision making and obstacle avoidance for deep-sea mining vehicles (DSMV). However, conventional particle swarm algorithms have been prone to trapping in local optima and have slow convergence rates when applied to underwater robot path planning. In order to secure a safe and economical three-dimensional path for the DSMV from the mining area to the storage base in connection with innovative mining system, this paper proposes a multi-objective optimization algorithm based on improved particle swarm optimization (IPSO) path planning. Firstly, we construct an unstructured seabed mining area terrain model with hazardous obstacles. Consequently, by considering optimization objectives such as the path length, terrain undulation, comprehensive energy consumption, and crawler slippage rate, we convert the path planning problem into a multi-objective optimization problem, constructing a multi-objective optimization mathematical model. Following that, we propose an IPSO algorithm to tackle the multi-objective non-linear optimization problem, which enables global optimization for DSMV path planning. Finally, we conduct a comprehensive set of experiments using the MATLAB simulation platform and compare the proposed method with existing advanced methods. Experimental results indicate that the path planned by the IPSO exhibits superior performance in terms of path length, terrain undulation, energy consumption, and safety.

Keywords: deep-sea mining vehicles; path planning; improved particle swarm optimization; unstructured seabed terrain

Citation: Lu, C.; Yang, J.; Leira, B.J.; Chen, Q.; Wang, S. Three-Dimensional Path Planning of Deep-Sea Mining Vehicle Based on Improved Particle Swarm Optimization. *J. Mar. Sci. Eng.* **2023**, *11*, 1797. https://doi.org/10.3390/jmse11091797

Academic Editor: Alessandro Ridolfi

Received: 14 August 2023
Revised: 10 September 2023
Accepted: 13 September 2023
Published: 14 September 2023

1. Introduction

The benthic zone of the oceans is replete with mineral resources distinguished by their vast reserves, superior grades, and great extension. These resources prominently include manganese nodules, polymetallic sulfides, and cobalt-rich crusts. Seabed mineral resources have been recognized as a significant strategic objective for global economies [1]. The deep-sea mining system, a critical apparatus for the extraction of these seabed mineral resources, is now widely endorsed by the international community. Currently, the pipe-lifting mining system, as illustrated in Figure 1, has been adopted universally. This system comprises three integral subsystems: the underwater acquisition system, the pipeline delivery system, and the surface support system [2]. Nonetheless, the pipe-lifting mining system continues to face significant challenges. The considerable distance required for pipeline transportation places rigorous demands on system stability. The pipeline configuration further imposes restrictions on operational procedures, thus limiting the extension of the collection area. Additionally, the substantial extraction of sediment raises environmental concerns, as the discharge of tailings detrimentally affects the marine ecosystem and beyond.

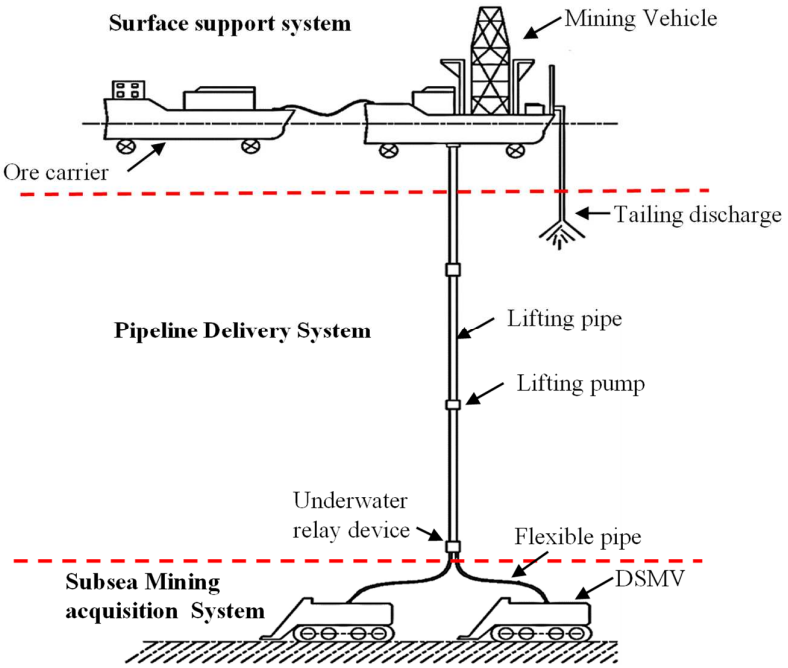

Figure 1. The composition of present deep-sea mining systems.

Facing numerous limitations of the pipe-lifting mining mode, a new pattern of deep-sea mining is being developed (as shown in Figure 2), which consists of a mineral acquisition system, a mineral storage system, a lifting system and a surface support system. The particular operational modality can be delineated as follows: The deep-sea mining vehicle (DSMV) accumulates minerals within the designated extraction zone; upon reaching capacity, the mineral container is conveyed to a storage base, where it is subsequently transferred to a storage hold. Utilizing a steel cable, the storage hold is then elevated to a surface support vessel, culminating in the successful completion of a closed-loop mission cycle. The DSMV represents a critical part of the underwater acquisition system, and it also contains a significant number of complex technical components. These components involve environmental perception technology, soft bottom traveling technology, intelligent path planning technology, and underwater positioning and navigation technology, etc. [3,4]. However, the DSMV still faces numerous challenges in the process of traveling from the mining area to the storage base, such as traversing rugged terrains, lack of transparency, and circumventing unforeseen obstacles, among others. Therefore, the exploration of intelligent path planning methodologies in relation to unstructured terrains emerges as an imperative for amplifying the efficiency of DSMVs concerning navigation, mineral procurement, and the integrated efficacy of deep-sea mining systems.

In the novel type of deep-sea mining system, the purpose of path planning is based on a set of constraints to effectively avoid regions with multiple obstacles in order to obtain an optimal or sub-optimal path with minimal path length and time or energy consumption from the mining area to the storage base. Three-dimensional path planning for DSMVs facilitates navigation, positioning, and trajectory tracking, thereby enhancing the efficiency of the transfer and the safety of mining vehicles. Furthermore, it elevates the operational efficiency of the entire mining system, serving as a pivotal exemplification in relation to advanced planning of these systems [5].

Based on research of traversal path and route selection of DSMV, Mathai [6] introduced a mining area path with high coverage and low repetition rate, while considering the impact of safety and economic effects. Shi [7] regarded the DSMV as a particle and proposed two-dimensional grid path planning based on ant colony optimization (ACO) particle swarm optimization hybrid algorithms, but ignore the kinematic and dynamic constraints of the DSMV. To ensure the maximum acquisition efficiency of the DSMV, Park [8] designed

two novel modes of a DSMV traveling in a mining area: contour-parallel and square-parallel. Dai and Liu [9] proposed two new traveling paths to improve the collection efficiency and ability of obstacle avoidance of the DSMV. Based on a two-dimensional seabed grid terrain, Jiang [10] proposed an improved ACO algorithm to plan the travel path of a DSMV in unstructured mining areas, and the effectiveness of the algorithm was verified by simulation experiments. Chen developed a path-following controller for the DSMV, refining the dynamic model and proposing an improved deep deterministic policy gradient algorithm [11]. Existing literature on DSMV path planning predominantly concentrates on the exploration of traversal paths and navigation methodologies within mining areas, with both aspects being implemented in a two-dimensional Cartesian grid framework. However, such bidimensional grid representations neglect essential seabed elevation data, which may consequently result in an imprecise depiction of the actual seafloor topography. Moreover, significant variations in seafloor terrain, the presence of submarine canyons, and the occurrence of uncharted impediments contribute to the insufficiency of the two-dimensional grid-based pathways in guaranteeing the safety of DSMVs. As a result, there is an urgent need to devise path planning strategies for DSMVs that incorporate a three-dimensional topographical context.

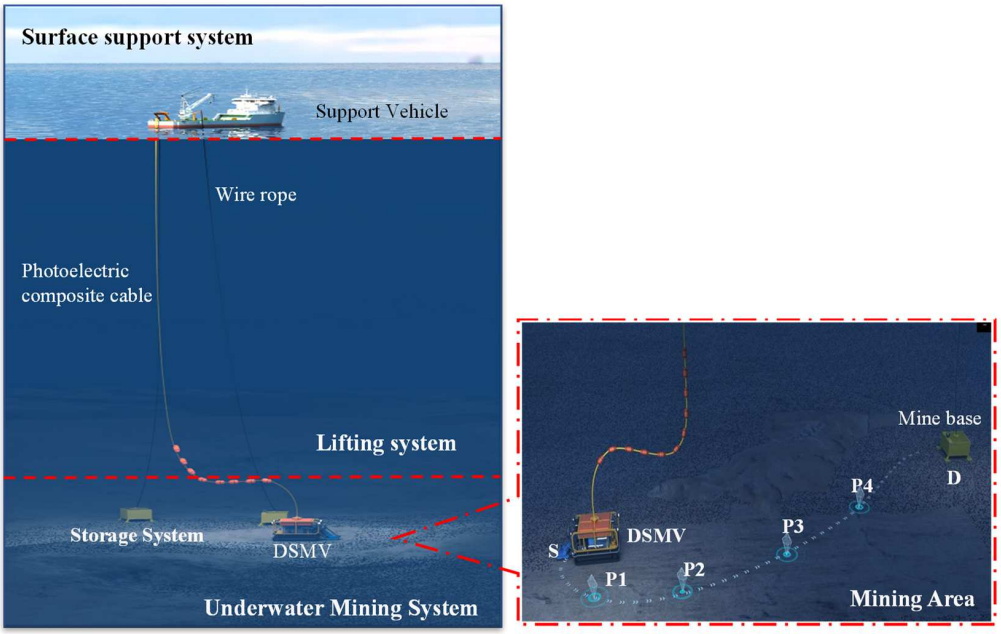

Figure 2. An innovative approach to deep-sea mining systems.

In recent years, there are many kinds of evolutionary algorithms that have been used for solving the path planning problem in a three-dimensional environment for underwater vehicles [12–14]. Leng [15] proposed an optimal and efficient path planner based on shell space decomposition (SSD), which is used to solve the path planning problem of underwater robots in cluttered and uncertain environments. The proven SSD solution can help autonomous underwater vehicles (AUV) navigate in compliance with the best trajectory in a variable ocean environment with obstacles. Yan [16] introduced a multi-objective whale optimization algorithm based on forward-looking sonar, which takes the least energy consumption, the shortest and the safest path as the optimization goals in order to solve the three-dimensional path planning problem of the underwater vehicle. Xiong [17] presented an ACO algorithm for multi-autonomous marine vehicles based on *Voronoi* diagrams to cope with the difficulty of adaptive ocean sampling, and the *Voronoi-based* ACO path planning was verified by simulation experiments and model tests to identify the optimum scheme in relation to adaptive sampling of ocean data. Liu [18] proposed a continuous ACO method based on a probabilistic random walk strategy and adaptive waypoint repair method to optimize the path of each Unmanned Ground Vehicle. Aiming at the problem

of low accuracy of existing terrain matching methods in areas with few characteristic features, Li [19] introduced a seafloor terrain matching navigation method to avoid these areas in order to plan the optimal path of AUVs. Zhu [20] proposed an integrated bionic self-organizing map algorithm for task allocation and path planning of AUV systems for obstacle avoidance in three-dimensional underwater environments. Although the above algorithm can provide an effective reference route for underwater robots, it has several shortcomings: complexity, a tendency to exhibit premature convergence and stagnation behavior.

Considering the limitations of existing algorithms in relation to path planning for underwater vehicles, some researchers have shifted focus to particle swarm optimization (PSO). PSO, a population-based optimization algorithm, was introduced by Eberhart and Kennedy in 1995 [21], drawing inspiration from birds' foraging behavior. Notable for its simplicity, high accuracy, and robustness, PSO has been employed in various combinatorial optimization problems [22], such as multi-objective optimization [23], robot path planning [24], and wind power generation prediction intervals [25]. Despite PSO's potential in addressing complex terrain path selection for robots, its slow convergence and tendency to be trapped in local optima have led researchers to enhance and optimize the algorithm for various applications. Das [26] developed a co-evolutionary PSO method for multi-robot trajectory planning. Liang [27] utilized K-means clustering to construct subpopulations, integrating global historical optima and neighborhood optima for particle swarm velocity and position updates. Tharwat [28] employed Bezier curves for path planning and introduced a novel chaotic PSO algorithm to control Bezier curve points, generating the shortest and smoothest path between initial and goal points. Zheng [29] employs an augmented PSO algorithm for global path optimization and an enhanced Artificial Potential Field (APF) for dynamic obstacle evasion, yielding superior performance in comprehensive path planning and real-time obstacle avoidance as validated by experimental results. While existing literature has indeed enhanced the performance of underwater robotic path planning using PSO, it still exhibits shortcomings when dealing with realistic scenarios, such as with a majority focusing primarily on 2D terrains and with high sensitivity to parameter selection. Furthermore, the shortest path length is mainly considered, while the security and energy consumption aspects are not considered.

The main contributions of this paper can be summarized as follows:

(1) In the context of a novel mining system, a three-dimensional, unstructured seabed terrain has been modeled via simulation, incorporating random hazardous zones and various obstacles. The fundamental goal of this endeavor is to formulate an optimized, collision-free path that traverses from the subaqueous mining location to the mineral storage base.
(2) This paper introduces an improved particle swarm optimization (IPSO) algorithm by means of velocity updating, application of weight factors and learning factors utilization of population iteration. This is to better balance the global search ability and to increase the convergence speed of the algorithm.
(3) Considering the path length, terrain undulation, energy consumption and crawler slippage rate, the path planning problem is transformed into a multi-objective optimization problem, which can make the DSMV move in compliance with the global optimal multi-objective path.
(4) Utilizing the three-dimensional seabed terrain model as the foundation, we propose an enhanced update mechanism for the PSO algorithm, specifically tailored for multi-objective optimization in path planning. The efficacy of this improved PSO variant, henceforth referred to as IPSO, is substantiated through numerical experiments.

The remainder of this paper is structured as follows: Section 2 establishes a three-dimensional seabed environment model. Section 3 elucidates the IPSO algorithm and corresponding verification experiments for each enhancement. Section 4 compares the IPSO with alternative algorithms to demonstrate its feasibility and effectiveness in DSMV path planning. Section 5 presents the simulation and analysis of the DSMV trajectory

planning, comparing the algorithms referenced in this paper to substantiate the rationality of IPSO. Finally, Section 6 offers concluding remarks.

2. Path Planning Strategy

This section presents a methodology for achieving high-precision Cobalt-rich crust seabed environment modeling through numerical simulation, which serves as a crucial component in aiding DSMV with path planning. To this end, a multi-objective mathematical model is developed for the DSMV's path planning process, which takes into account the path length, terrain undulation, minimal track slip, and overall power consumption which are all built into the cost functions. In order to ensure the smoothness of the path, a cubic B-spline curve is introduced to interpolate the trajectory.

2.1. Microtopographic Features of Cobalt-Rich Crusts

The micro-topographic characteristics of cobalt-rich crusts have a great influence on DSMV path planning. Cobalt-rich crust deposits are distributed on the tops and upper slopes of underwater mountains such as seamounts, islands, and mid-ocean ridges with a water depth of 800–3200 m. Crusts mainly occur on table mounts and pointed-topped seamounts. The diameter of the flat roof is usually 5~9 km, and the diameter of the base is 10~20 km. The basement layer is basalt, metamorphic basalt, cellophane clastic rock, pyroclastic rock, etc. Large areas of plate-like crusts and gravel-like crusts are distributed in the crust development area with hard sediment. The tracked DSMV is identified as appropriate for navigation and extraction operations on soft seafloor substrates, owing to its superior traction, minimized ground-specific pressures, and exceptional maneuverability. Figure 3a shows the overall slope change in the subsea flat-topped seamount area, with the S–N cross-section given in Figure 3b, and the E-W cross-section in Figure 3c. Terrain slopes greater than 10° appear on the sloping wings of the seamounts (above the pink line) in Figure 3b,c. A terrain with a slope of less than 5° appears in the top area (below the pink line). The black circles I and II show the slope change at the summit of the seamount, with small topographic relief. To sum up, the flat terrain and small slope changes at the summit of the seamount with cobalt-rich crusts can satisfy the walking criteria for the DSMV, and the harder substrate can provide enough shear force for the track. In order to improve the feasibility of simulated terrain, this paper considers the convex mountain obstacle at the summit of the seamount area.

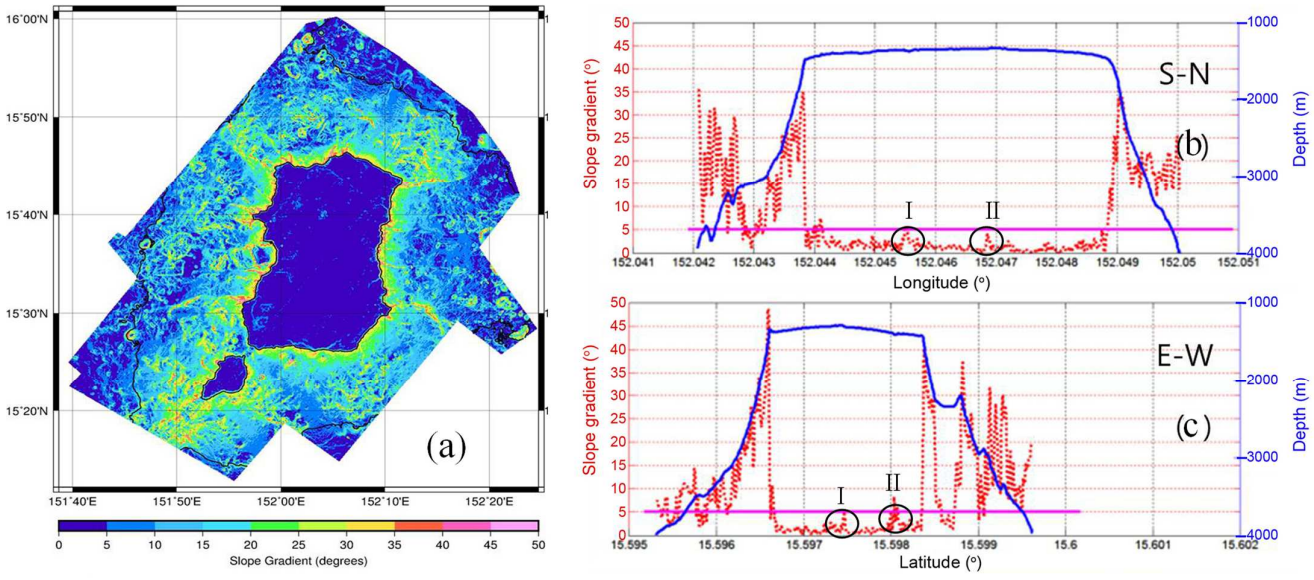

Figure 3. The overall slope change in the seamount. (**a**) the multi-beam-derived slope gradient map of the search area; (**b**) S–N cross-section of the search area; (**c**) E-W cross-section of the search area.

In Figure 3b,c, the blue solid lines show the depth variations (right axis) and the red dotted lines show the slope variations (left axis). The black circles I and II denote the slight undulations at the top of the seamounts. These profiles exemplify the steep-sided flanks and flat summit, as well as axial slope gradient profiles. The areas of high gradient slopes (>5°) are found at the seamount flank, whereas the zones with slope gradients less than 5° (pink lines) are found on the summit [30].

2.2. Submarine Environment Model

Land-based tracked robots and the DSMV require highly precise maps and elevation data to facilitate navigation in challenging mountainous terrains. Similarly, the DSMV necessitate a three-dimensional environmental model of the seafloor to aid in path planning and decision making when traversing complex underwater mining areas. This paper analyses the micro-topographical characteristics of seamounts and proposes a digital simulation model for the mining area. This model effectively captures seafloor terrain variations, slopes, and obstacle information, with a level of precision suitable for DSMV path planning. In this paper, we denote by (*x*,*y*,*z*) the coordinates of a path waypoint in the three dimensional environment. The traveling space for the DSMV can then be expressed as follows:

$$\{(x,y,z)|x_{\min} \le x \le x_{\max}, y_{\min} \le y \le y_{\max}, z_{\min} \le z \le z_{\max}\} \tag{1}$$

where $x_{\min}, x_{\max}, y_{\min}, y_{\max}, z_{\min}, z_{\max}$ define the bounds of x, y, z respectively. We present a three-dimensional environmental model of a cobalt-rich crust seabed mining area based on the Cartesian coordinate system [31].

$$Z_1(x,y) = \sin(y+a) + b \times \sin(x) + c \times \cos(d \times \sqrt{x^2+y^2}) + e \times \cos(y) + f \times \sin(g \times \sqrt{x^2+y^2}) \tag{2}$$

where x, y denote the point coordinates projected on the base plane of the summit of the seamount of the Cobalt-rich crust; Z_1 represent the elevation corresponding to the base plane, where the lowest point of base plane corresponds to $Z = 0$; are constants, which control the datum terrain variation in the map. For the larger mountain obstacles encountered by the DSMV during the traveling process, an exponential function is applied in order to describe a mountain obstacle [32], and a schematic diagram is shown in Figure 4.

$$Z_2(x,y) = \sum_{i=1}^{n} h_i \exp^{\left[-\left(\frac{x-x_i}{x_{si}}\right)^2 - -\left(\frac{y-y_i}{y_{si}}\right)^2\right]} \tag{3}$$

where Z_2 donates the elevation at a specific position with coordinates *(x,y)*; (x_i, y_i) are the center coordinates of the *i*-th mountain; h_i controls the height of the mountain, x_{si}, y_{si} are attenuation parameters of the *i*-th mountain along the x, y axis; n represents the total number of mountain obstacles.

For a DSMV that traverses areas where penetration into the seabed may occur, a high risk of failure of the operation is generally present. In this paper, we consider a mining area with a soft substrate that poses such a threat *k*. Because areas with thin and soft substrates are likely to cause the mining vehicles to sink, making them unable to progress, the corresponding mathematical model of the high-risk area is formulated by means of a geometric cylinder (as shown by the blue transparent cylinder in Figure 4). The simplified mathematical model for a high-risk (hazardous) area is written as:

$$T_k = (x_k^s, y_k^s, z_k^s, r_k) \tag{4}$$

where (x_k^s, y_k^s) is the center (in the horizontal plane) of the *k*-th high-risk area, z_k^s is the height of the cylinder, and r_k is the radius of the cylinder cross-section.

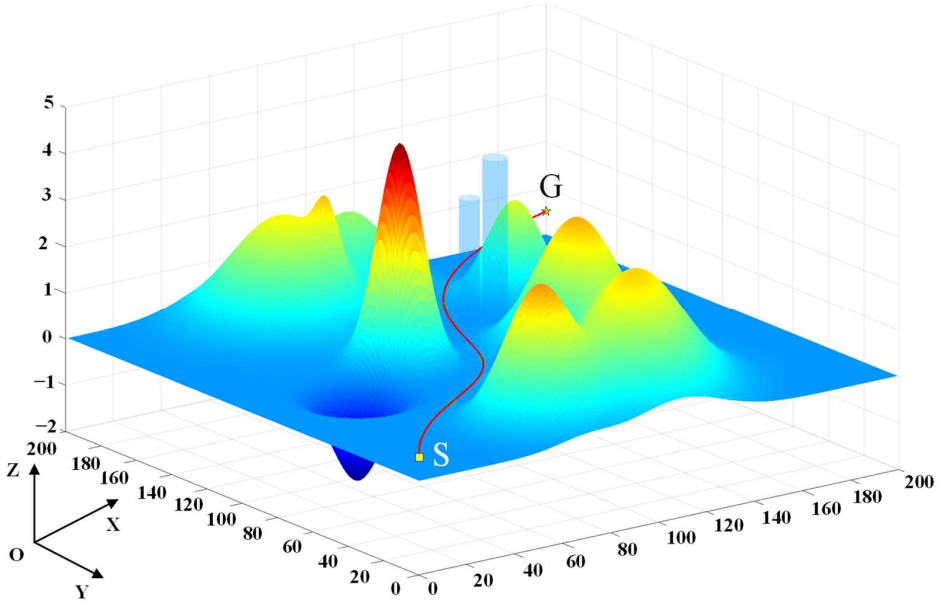

Figure 4. Schematic diagram of simulated seabed terrain-and path followed by the vehicle, the red line represents the planned path, "S" represents the start point, "G" represents the target point.

In the global coordinate system $O-XYZ$, S is the initial starting point of the DSMV and G is the target point. The path of the DSMV is characterized by the spatial coordinate points that are close to each other and connected by smooth curves. Suppose a set of node sequences is $\{S, w_1, w_2, w_3, \ldots, w_{n-1}, G\}$, this sequence consists of N+1 nodes, S represents the initial point of the DSMV, G represents the target point, and w_1-w_{n-1} are the nodes of the path that the DSMV is following. the initial point is referred to as $S = (x_0, y_0, z_0)$, the target point as $G = (x_n, y_n, z_n)$, while $w_i = (x_i, y_i, z_i)(i = 1, 2, \ldots, n-1)$ represents each node during the motion of the vehicle.

2.3. Multi-Objective Model of DSMV Path Planning

Referring to the DSMV, the shortest path length contributes to efficient completion of its tasks and the entire mission; a smaller undulation of the path benefits its traversability; the minimum energy consumption path contributes greatly to providing long-term and energy-demanding mining; the minimum crawler slip path can ensure that the DSMV executes activities with the lowest slip and highest safety. Consequently, four objectives involving path length, undulation of the path, economic cost, and crawler slip of the DSMV should be considered in order to assess the "total quality" of the path.

2.3.1. The Shortest Path Length

Generally, the shorter paths can save more time and energy, which can enable the DSMV to accomplish the task quickly. In this paper, the path of the DSMV consists of several N waypoints $s_i(i = 1, 2, 3, \ldots, N)$, (x_i, y_i, z_i) and $(x_{i+1}, y_{i+1}, z_{i+1})$ represent the three-dimensional coordinates of i-th waypoint and adjacent waypoint, respectively. Then, the distance that is applied to formulate the corresponding objective function (cost function) f_d is defined as:

$$f_d = \sum_{i \in N} \sum_{\substack{i+1 \in N; \\ (i, i+1) \in E}} d_{i,j} = \sum_{i \in N} \sum_{\substack{i+1 \in N; \\ (i, i+1) \in E}} \left(\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2}\right) \quad (5)$$

where N is the set of nodes that form the graph, and E the set of edges of the graph; $d_{i,j}$ is the distance metric between node i and node $i+1$. In our formulation, the Euclidean distance was adopted. $x_i, x_{i+1}, y_i, y_{i+1}, z_i, z_{i+1}$, are the geographical coordinates of node i and node

$i+1$ along the transverse, longitudinal and vertical axes, respectively, the dimension of the path length is *meter*.

2.3.2. The Undulation of the Path

The undulation of a path refers to the variation in elevation over a certain distance of the path, which can be expressed by slope, elevation difference or standard deviation. The greater the undulation of the terrain, the rougher the terrain is, which is not conducive to the travel of the DSMV. Accordingly, it is recommended to select paths with flatter terrains whenever possible. In this paper, the elevation difference is chosen to represent the undulation of the path f_u defined as:

$$f_u = \sum_{i=1}^{N-1} \left| Z_{i-1} - Z_i \right| \tag{6}$$

where $|Z_{i-1} - Z_i|$ represents elevation difference when moving from path point $i-1$ to path point *i.* In order to simplify the calculation, the sum of all of these difference is taken to represent an approximation to the undulation of the path, and the dimension of this undulation is length, usually expressed in meters.

2.3.3. The Energy Consumption of the Path

In general, reduced energy consumption corresponds to decreased economic costs. The DSMV relies on the electrical system of a supporting surface floater, which is connected through a photon–electron composite cable in order to provide the necessary energy. The energy consumption associated with the DSMV arises from multiple components, including the acquisition, control, perception, and travel systems. The present study posits that the acquisition and traveling systems of the DSMV are mutually exclusive in their operation. While the control and perception systems operate continuously throughout the duration of the mission, the power supply to the DSMV system remains constant. Thus, the energy consumption of the control system and perception system is determined by the amount of time of the DSMV is in operation on the seabed, which consist of the mining time t_1 and the travel time t_2. The energy consumption function f_e is defined as:

$$f_e = t_1 P_1 + t_2 P_2 + (t_1 + t_2) P_3 \tag{7}$$

where f_e is energy consumption of the entire system; t_1 is the mining time, t_2 is the travel time. P_1 is the average power of the acquisition system, P_2 is the average power of the travel system, P_3 is the average power of the control and perception system, and the dimension of the energy consumption is kWh.

2.3.4. Path of Crawler Slippage

For the DSMV, the force of the mining vehicle movement is generated by the shearing effect of the crawler on the deep-sea soil, and in the process of the crawler shearing the soil, a slip phenomenon may occur. The deep-sea soil in different areas will provide different shear forces to the crawler, so the DSMV will face different slippage rates. The successful navigation of the DSMV through soft sediment environments is critically dependent on the slippage rate between the vehicle crawler and the seabed soil. Here, we define a threshold slippage rate below which the DSMVs can traverse the mining area smoothly. However, when the slippage rate exceeds an upper threshold, the DSMV cannot get out of this area due to slipping, thereby classifying the region as an obstruction zone. In this paper, the smallest circle k $((x_k, y_k), r_k)$, which covers the area with the slippage rate exceeding the threshold, is used to reflect the obstacle area of DSMV, where k is the number of the obstacle (i.e., Obs_k), and with (x_k, y_k) and r_k denoting the center and the radius of the circle, respectively. The real distance, the least distance, and the lower limit of

the maximum distance between Obs_j and the path are denoted by d_{real_j}, $d_{safe\_\min j}$ and $d_{safe\_\max j}$, respectively. The crawler slips of the path f_s is defined as:

$$f_s = \begin{cases} 0, slip\ rate < 0.5 \\ D, slip\ rate > 0.5 \end{cases} \tag{8}$$

$$D = \begin{cases} 1 & d_{real\_j} < d_{safe\_\min j} \\ 0 & d_{real\_j} \geq d_{safe\_\max j} \\ \frac{d_{real\_j} - d_{safe\_\min j}}{d_{safe\_\max j} - d_{safe\_\min j}} & others \end{cases} \tag{9}$$

When the slippage rete is less than 0.5, it can be considered that the DSMV is not affected by the slippage rate between the track and the seabed soil, so that it can pass through the area safely. When the slippage rete is greater than 0.5, the mining vehicle cannot pass through, and this area is regarded as a high-risk area. At this time, the problem can be equivalent to finding the safe distance for the mining vehicle to pass through this area. Here, D is the distance between the actual path and the high-risk area, The smaller the D value, the safer the path will be, f_s is dimensionless.

2.3.5. The Formulated Path Planning Model for the DSMV

In general, DSMV path planning typically involves a multi-objective optimization, where the different objectives have different optimal solutions. Broadly speaking, there are three ways to deal with multi-objective problems: a direct approach, aggregation or transformation, and Pareto-set approximation [33]. However, the direct approach is difficult, especially in the case when multiple objectives seem conflicting. The limitation of Pareto front includes lack of diversity, scalability, and the inability to account for changes in objectives over time. Therefore, this paper uses aggregation or transformation by combining multiple objectives into a single composite objective function (cost function) so that standard methods for optimization can be used. By using the optimized planned path in terms of such multiple criteria including the path length, path undulation, energy consumption and the crawler slip along the path, the cost function is defined as:

$$F_c = \omega_1 f_d + \omega_2 f_u + \omega_3 f_e + \omega_4 f_s \tag{10}$$

where F_c represent cost function, f_d is objective function of path length, f_u is the objective function of the path undulation, f_e is energy consumption objective function, f_s is the objective function of crawler slip of the path. As these metrics inherently exhibit variations in magnitude and units, an appropriate comparison mandates a standardized framework. To this end, a standardization method was adopted, transforming each variable into a dimensionless form. Specifically, for each metric, the mean was subtracted from individual observations, followed by division by the standard deviation, resulting in a standardized score with a mean of zero and a standard deviation of one: $f_i = \frac{x_i - \mu}{\delta}$. where x_i represents the original value, the μ is the mean value, and the δ represents the standard deviation of variable. Through this process, the differences in scale among the various parameters were effectively neutralized, allowing for their integration into a cohesive and comparable objective function. ω_i is the weighting coefficients, the solution strongly depends on the chosen weighting coefficients. these weights have to be positive, satisfying:

$$\sum_{i=1}^{4} \omega_i = 1, \quad \omega_i \in (0, 1) \tag{11}$$

The optimal solution of the multi-objective path planning problem is dependent on the assigned weight coefficients. In this study, the path planning of the DSMV is conceptualized as a global static process. As a result, the weight coefficients were deliberately chosen to reflect the significance of each objective in the context of deep-sea mining missions.

Specifically, the path length is prioritized with a weight coefficient of $\omega_1 = 0.5$ due to its crucial role in minimizing both time and associated costs. Next in priority are the path undulation and crawler slippage rate, which are fundamental to vehicle stability and safety. We assign them similar weight coefficients of and $\omega_2 = \omega_3 = 0.2$ to balance these objectives. Energy consumption has the least weight ($\omega_4 = 0.1$) in the optimization problem, given that the DSMV receives stable and continuous electric energy from the surface vehicle. This methodological approach to weight setting aims to holistically address the multi-objective optimization problem in question.

2.4. Cubic B-Spline Curve Path Smoothing Model

The conventional path planning algorithm typically involves constructing a series of connected trajectory points to form the path. However, this results in abrupt curvature changes that fail to conform to the dynamic constraints of the DSMV and that does not accurately align with the actual path. In order to reduce the shaft wear of the DSMV actual track and the turbidity plumes on the seabed, the steering path should be smooth. Commonly used methods for path smoothing include spline curve and straight-line segment approximation method for complex curves [34]. It is shown based on both theory and practice that, there are several disadvantages in using straight-line segments to approximate complex curves and using linear interpolation to smooth the path, such as: large speed fluctuation, low smoothing efficiency, and contradiction between smoothing accuracy and speed requirements [35]. Spline curves offer a superior alternative to the straight-line approach when navigating complex curves due to their increased precision of fitting and lower computational requirements, which effectively mitigate potential data scarcity and other associated limitations. Thus, spline curve gradually plays an increasingly important role in path smoothing methods [36].

As one of the most typical functions of spline curves, the B-spline curve function has continuous first and second derivatives, and this represents an advantages relative to the Bezier method. Due to these properties, it can overcome the defects of the Bezier method that do not have such local properties due to the overall representation (i.e., moving one control point will affect the whole curve). The B-spline curve has many advantages in path smoothing: locality, geometrical invariability, symmetry, recursion, continuity, convex hull characteristics, and convexity preservation [37]. It is accordingly able to fit the requirements of continuous change in the DSMV speed. The contradiction between the approximation of the curve and the convexity preservation can be solved by the B-spline curve, thus, the path curve is relatively smooth, and the curvature changes are minor before and after turning. In this section, the smooth path of the cubic B-spline curve is adopted to ensure the stability and continuity of the DSMV traveling on the seabed. A B-spline curve of order k can be defined by a total of $n+1$ control points $p_0, p_1, \ldots, p_n$ and a set of non-decreasing continuously changing vector nodes u_i. The expression is as follows:

$$p(u) = [p_0 p_1 \ldots p_n] \begin{bmatrix} N_{0,k}(u) \\ N_{1,k}(u) \\ \cdots \\ N_{n,k}(u) \end{bmatrix} = \sum_{i=0}^{n} p_i N_{i,k}(u) \tag{12}$$

where $p_i (i = 0, 1, \ldots, n)$ is the control point of B-spline curve, which is used to limit the curve range; u is the vector node, n is the number of control point, k is the order of B-spline curve, and the basis function $N_{i,k}(u)$ is defined as follows:

$$\begin{cases} N_{i,0}(u) = \begin{cases} 1, u_i \le u \le u_{i+1} \\ 0, \quad others \end{cases} \\ N_{i,k}(u) = \frac{u-u_i}{u_{i+k}-u_i} N_{i,k-1}(u) + \frac{u_{i+k+1}-u}{u_{i+k+1}-u_{i+1}} N_{i+1,k-1}(u) \end{cases} \tag{13}$$

where $u_i (i = 0, 1, \ldots, n)$ are vector nodes, and the sequence satisfies the non-decreasing relationship. In this paper, the vector repetition degree of the nodes at both ends is set as

$k+1$, the inner node vector is uniformly distributed, given $k=3$, that is, fitting the path with a cubic quasi uniform B-spline curve equation, the strength of which is to complete the path planning in complex terrains through less design parameters. Figure 5 shows a cubic B-spline curve, the blue circles are the control points and the orange polygon is the control polygon. The blue line is the original path, and the red line is the processed smooth path.

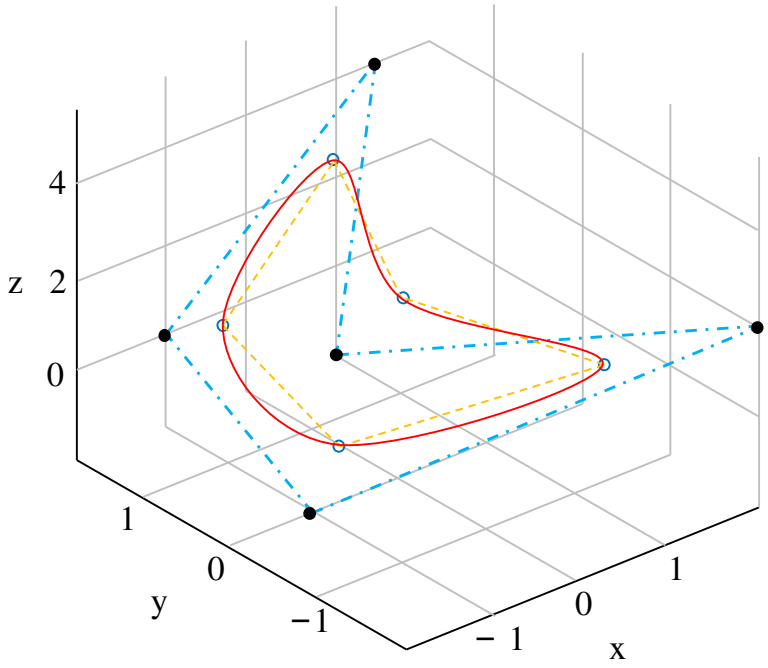

Figure 5. Schematic diagram of cubic B-spline curve, the blue dotted line is the path before optimization, the red line is the path after optimization, and the black points are control points.

3. Improved Particle Swarm Optimization (IPSO)

In this section, first based on a brief description of the PSO algorithm, the basic principle of using PSO to solve the path planning problem is expounded. Analyzing its optimization mechanism and convergence characteristics, the basic PSO is subsequently modified in terms of four different aspects which are velocity update, inertia weight, learning factor, and population iteration in order to avoid falling into a local optimum and premature convergence.

3.1. Particle Swarm Optimization

PSO is a population-based stochastic optimization algorithm, which has been successfully applied in various fields [22,38]. Within the PSO framework, the algorithm commences with a randomized initialization that generates a set of candidate solutions. These solutions are then iteratively improved via updates to their position and velocity, with the aim of finding the global optimal solution. For every individual particle, its velocity denotes the direction of search and is continually revised based on its prior state, as well as the best positions achieved by the particle itself and the overall collective. The subsequent particle position is then determined by its previous state and the instantaneous velocity. The PSO entails that if a particle's current position surpasses its previous position with respect to the prescribed fitness function, the new position will be designated as its personal best. Moreover, if the new position is superior to the positions of all other particles in the swarm, it will be regarded as the global best. For a three-dimensional path planning problem,

suppose that the waypoint number of all the individual particles is D, then the position and velocity vector for the i-th particle can be expression as follows:

$$p_i = (p_{i1}, p_{i2}, \ldots, p_{iD}), i = 1, 2, \ldots, N \tag{14}$$

$$v_i = (v_{i1}, v_{i2}, \ldots, v_{iD}), i = 1, 2, \ldots, N \tag{15}$$

The optimal position searched by the i-th particle is the individual extreme value:

$$p_{best} = (p_{i1}, p_{i2}, \ldots, p_{iD}), i = 1, 2, \ldots, N \tag{16}$$

The optimal position searched by the entire particle swarm is the global optimization:

$$g_{best} = (p_{g1}, p_{g2}, \ldots, p_{gD}), g = 1, 2, \ldots, N \tag{17}$$

After calculating the individual extreme value and the global optimal value, then update the velocity v_{id} and position x_{id} of the particles [39,40]:

$$\begin{cases} v_{id} = \omega_k * v_{i-1,j} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \\ x_{id} = x_{id} + v_{id} \end{cases} \tag{18}$$

where v_{id} is the current velocity of the particle; x_{id} is the current position of the particle; ω_k is inertia weight, for trust in the current speed direction; c_1, c_2 are learning coefficients; r_1, r_2 is a uniform random number in the range of [0, 1] to increase the randomness of the search; p_{id} is the individual best position; p_{gd} is the global best position searched by the entire population. The flight trajectory of the particle determined by Equation (17) which consists of three parts, is illustrated in Figure 6. The first component is the motion inertia of the particle, represented by the green block, which encapsulates the particle's initial velocity information. The second component, indicated by the yellow block, pertains to cognitive information, and comprises the original velocity of the particle. The third component, denoted by the blue block, signifies social cognition and reflects the distance of the best position p_{gd} among the population. This component signifies the sharing of social information among the particles. Finally, the particle update mechanism after integration is represented by the pink block in Figure 6.

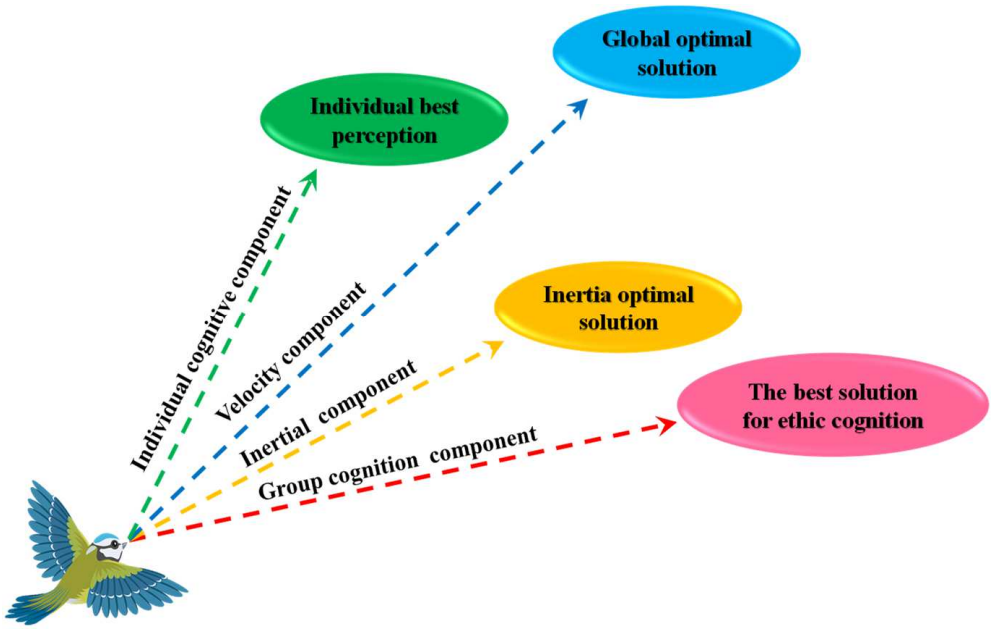

Figure 6. Update method of each generation of particles.

As a heuristic optimization algorithm, the PSO is based on a simple principle, few parameters and easy implementation. Each particle is updated based on the internal speed, memorizing the positives and negatives solutions during the optimization process, and saving the information about pros and cons. However, for high-dimensional complex problems, problems associated with premature and poor convergence performance are usually encountered, so there is no guarantee for convergence to the optimal point. When applied to robot path planning the following problems still exist [41,42]:

(1) The core of PSO is an iterative optimization algorithm. It needs more time to get a feasible solution, so it has limitations when it comes to path planning applications. In addition, the tendency of the standard particle swarm to move to a local optimum and to stagnate represents a basic problem.
(2) Constrained by the characteristics of the modelling environment, PSO is predominantly applied in two-dimensional grids or 2.5D environments, with scant experience in relation to DSMV trajectory planning for three-dimensional settings.

3.2. The Improved Particle Swarm Optimization Algorithm

In view of the weakness of the PSO algorithm mentioned above in solving the path planning problem, this paper proposes several enhancements to the PSO. Specifically, the PSO is optimized and the adaptive weight method is leveraged to optimize the search and convergence capabilities of the PSO. Furthermore, to enhance the early local search ability and the late global convergence ability of the algorithm, asynchronously changing learning factors are employed and the velocity update scheme is modified. Finally, to improve the optimization efficiency and increase the population diversity, the crucial natural selection operation applied in genetic algorithms is integrated.

3.2.1. The Improved Particle Velocity Update Method

The introduction of the average best position S into the velocity update scheme enables the particles to gather information from other particles, resulting in enhanced global search capability and group cooperation in the PSO algorithm.

$$v_{id} = \omega * v_{i-1,j} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) + c_3 r_3 (s - x_{id}) \tag{19}$$

$$c_3 = 1 + \frac{t}{t_{\max}} \tag{20}$$

$$s(t) = \frac{1}{M} \sum_{i=1}^{M} p_{id}(t) \tag{21}$$

where c_3 represents the newly introduced acceleration coefficient, while r_3 denotes a random number within the range of (0, 1). while s signifies the average best position. M represents the total number of particles, t is the current number of iterations; $t_{\max}$ is the maximum number of iterations.

3.2.2. Adaptive Weight Function

As an important parameter in the PSO algorithm, the inertia weight ω is a pivotal parameter that dictates the propensity of a particle to retain its previous velocity, thus influencing its exploratory and exploitative behaviors in the search space. A larger ω amplifies the particle's global search capabilities, whereas a smaller ω enhances its focus on localized, fine-grained optimization. When combined with high velocity, an elevated inertia weight introduces stochasticity into the particle's movement, enriching its global exploration potential and increasing the probability of escaping local optima. This interplay between inertia weight and velocity is instrumental in shaping the algorithm's overall performance in complex optimization tasks. Thus, an appropriate value of the inertia weight ω is the key to avoid that the algorithm falls into a local optimum and to improve the search efficiency. This paper adopts a non-linear dynamic inertia weight ω strategy:

When the fitness values of each particle tend to be consistent or tend to be locally optimal, the inertia weight is increased, otherwise, the inertia weight is decreased. For the particle whose objective function value is better than the average, its inertia weight factor is small, so as to protect the particle. On the contrary, for the particle whose objective function value is worse than the average, the corresponding inertia weight factor is larger, which makes the particle move closer to a better search area. The improved non-linear dynamic inertia weight coefficient ω is expressed as follows:

$$\omega = \begin{cases} \omega_{\min} - \frac{(\omega_{\max} - \omega_{\min}) * (f - f_{\min})}{(f_{avg} - f_{\min})}, (f \leq f_{avg}) \\ \omega_{\max}, (f > f_{avg}) \end{cases} \tag{22}$$

where $\omega_{\max}, \omega_{\min}$ is the maximum and minimum value of ω respectively; f donates the current fitness value of particles; f_{avg} is the average fitness value; $f_{\min}$ is the minimum fitness value.

3.2.3. Asynchronously Varying Learning Factor

The individual learning factor c_1 and the social learning factor c_2 play a crucial role in guiding the trajectory of each particle in the search space. Specifically, c_1 influences how much a particle relies on its own best-known position, whereas c_2 determines how much the particle is influenced by the best-known position in its neighborhood or among all particles in the swarm. Different settings for c_1 and c_2 can significantly affect the optimization results. A higher value of c_1 would make the particle more self-reliant and focused on its individual experiences, potentially making it harder for the algorithm to escape local optima if the particle's own best-known position is not globally optimal. On the other hand, a higher c_2 value emphasizes collective intelligence, allowing particles to converge more quickly to a promising area of the search space but also risking premature convergence to suboptimal solutions if not properly balanced. c_1 and c_2 change differently over time during the optimization process which is referred to as asynchronous change. In this way, the particles have a strong self-learning ability and a strong global search ability during the initial stage of the optimization. In the later stage of the optimization, particles have greater social learning ability and less self-learning ability, which is conducive to convergence to the global optimal solution. The improved learning factors are expressed as follows:

$$c_1 = c_{1,ini} + \frac{c_{1,fin} - c_{1,ini}}{t_{\max}} * t \tag{23}$$

$$c_2 = c_{2,ini} + \frac{c_{2,fin} - c_{2,ini}}{t_{\max}} * t \tag{24}$$

where $c_{1,ini}$ and $c_{2,ini}$ donate the initial values of c_1, c_2; $c_{1,fin}$ and $c_{2,fin}$ denote the iteration end values of c_1, c_2; t is the current number of iterations; $t_{\max}$ is the maximum number of iterations, in this paper, $t_{\max}$ is defined as the anticipated or expected number of iterations.

3.2.4. Population Iteration Method Based on Natural Selection

In order to avoid particles getting trapped in local optima and further keep the particle population diverse, this paper introduces the principle of survival of the fittest in Darwin's theory of natural selection, integrating the natural selection mechanism of the genetic algorithm into the particle swarm optimization algorithm. The core idea is to sort the newly generated particle swarm according to the new fitness value in each iteration process according to the sorting selection method. Replace the particles in the worst p% (p is the percentage constant, that is, the probability of elimination) positions with the positions of the particles in the best top p% in the population, while retaining the historical optimal value memorized by each individual, thereby increasing the proportion of the particles closest to the optimal particle in the particle population. This selection can ensure that the particles have good optimization performance in each iteration process and can speed up

the convergence speed of the algorithm. The improved particle swarm update process is as follows:

$$\begin{cases} [\text{sort}(f), \text{sort } x] = \text{sort}(f) \\ e = \text{round}(M-1)/2 \\ x(\text{sort}((M-e+1):M)) = x(\text{sort}(1:e) \\ v(\text{sort}((M-e+1):M)) = v(\text{sort}(1:e) \end{cases} \tag{25}$$

where f donates particle fitness value; M is the total number of particles; *sort* indicates preferential sorting; *round* represents the rounding function.

To sum up, this paper proposes an IPSO algorithm, with the update formulas being described as follows:

$$\begin{cases} v_{id} = \omega * v_{i-1,j} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) + c_3 r_3 (s - x_{id}) \\ x_{id} = x_{id} + v_{id} \end{cases} \tag{26}$$

The Sphere is a non-linear unimodal function, and the Rastrigin is a non-linear multi-modal function. In order to simplify the expression, we make the following abbreviations: IV-PSO: improved velocity update method; IW-PSO: Improved weight; ILF-PSO: improved learning factor; IPSO: improved population iteration, it is also the final improved algorithm of this paper, and it is the same as the IPSO in the Section 5. After conducting an empirical analysis using the group method, the resulting algorithmic parameter configurations are presented in Table 1.

Table 1. Parameter for the improved PSOs.

PSO Kind	Parameter
PSO	M = 100, $t_{\max}$ = 100, $c_1 = c_2 = 2$, $w = 0.5$
IV-PSO	M = 100, $t_{\max}$ = 100, $c_1 = c_2 = 2$, $w = 0.5$, c_3 and S are set by (20–21)
IW-PSO	M = 100, $t_{\max}$ = 100, $c_1 = c_2 = 2$,$w_{\max} = 0.9$, $w_{\min} = 0.4$
ILF-PSO	M = 100, $t_{\max}$ =100, $c_{1\_ini} = 2$,$c_{1\_fin} = 0.5$, $c_{2\_ini} = 2$,$c_{2\_fin} = 0.5$, $w_{\max} = 0.9$, $w_{\min} = 0.4$
IPSO	M = 100, $t_{\max}$ = 100, $c_{1\_ini} = 2$,$c_{1\_fin} = 0.5$, $c_{2\_ini} = 2$,$c_{2\_fin} = 0.5$, $w_{\max} = 0.9$, $w_{\min} = 0.4$, p = 0.05

To verify the effectiveness of the above improvements, four classical benchmark functions from Congress on Evolutionary Computation (CEC)'2020 multi-modal multi-objective optimization (MMO) [43,44] are used to evaluate its performance, the simulation results are shown in Table 2.

Table 2. Calculation results and contrast based on benchmark functions.

Function	Statistics	PSO	IV-PSO	IW-PSO	ILF-PSO	IPSO
Kowalik	Best value	0.0009171	0.000564	0.000756	0.000823	**0.0003021**
	Mean value	0.003873	0.004325	0.005631	0.004325	**0.002758**
	St. Dev.	0.008256	0.00756	0.007258	0.006652	**0.006231**
Six-Hump Camel	Best value	−1.03163	−1.03163	−1.03163	−1.03163	**−1.03163**
	Mean value	−1.03163	−1.03163	−1.03163	−1.03163	**−1.03163**
	St. Dev.	1.23×10^{-10}	6.56×10^{-13}	4.32×10^{-11}	3.32×10^{-9}	$\mathbf{2.13 \times 10^{-14}}$
Branin	Best value	0.45376	0.42358	0.41852	0.402568	**0.397861**
	Mean value	0.456854	0.43675	0.43345	0.423482	**0.397863**
	St. Dev.	6.32×10^{-9}	1.32×10^{-13}	2.65×10^{-13}	3.32×10^{-14}	$\mathbf{1.05 \times 10^{-16}}$
Hartman	Best value	−3.28633	−3.28633	−3.28633	−3.28633	**−3.32046**
	Mean value	−3.323963	−3.138	−3.1224	−3.3224	**−3.321**
	St. Dev.	0.7728	0.5828	0.6448	0.5328	**0.45463**

Learning factors c_{1_ini} = 2, c_{2_ini} = 2, c_{1_fin} = 0.5, c_{2_fin} = 0.5: These were initially set to 2 and gradually decreased to 0.5 as the iterations progressed. The rationale behind this setting is to promote broad exploration of the solution space in the early iterations and then focus on exploitation around the best-found solutions in the later stages. This approach is supported by prior empirical studies that demonstrate the effectiveness of varying c_1 and c_2 in this manner for balancing exploration and exploitation. Additional learning factor c_3: This was introduced to further fine-tune the algorithm's performance. It is calculated as $c_3 = 1 + \frac{t}{t_{\max}}$, where t is the current iteration number and $t_{\max}$ is the maximum number of iterations. The purpose of c_3 is to adaptively change the acceleration of particles based on the progress of iterations. Inertia weights $w_{\max}$ and $w_{\min}$: We adopted an adaptive inertia weight strategy with $w_{\max}$ = 0.9 and $w_{\min}$ = 0.4. This adaptive method allows particles to have greater momentum in early iterations for global searching and reduced momentum in later iterations for more localized searching.

As shown in Table 2, the IPSO algorithm has good performance (best value, mean value and standard deviation) for the Kowalik, Six-Hump Camel, Branin and Hartman functions. The achievement of min best values and mean value can be considered as reliable indicators of the IPSO's enhanced optimization performance, and the min standard deviation can also be interpreted as evidence of the IPSO's robustness and algorithmic stability. Therefore, it can be concluded that the IPSO algorithm has excellent performance after being improved step by step.

The framework for the numerical simulation shown in Figure 7. The path planning process of DSMV based on IPSO is as show in Algorithm 1:

Algorithm 1 IPSO for path planning

//Environment construction
Establish the simulation environment model according to Equations (1)–(3);
Set the start point and target point for DSMV.
//IPSO initialization
Initialize the particle swarm parameters: $t_{\max}$,$w_{\max}$,$w_{\min}$,c_1,c_2;
  Initialize particles and velocities;
  Set a relatively large value for the fitness value;
  Calculate the fitness value and select p_{best} and g_{best}.
//main loop
  for t = 1: $t_{\max}$
    for j = 1:M
    update IPSO parameters by Equations (19)–(24);
    generate new particle position and velocity by Equation (25);
    calculate particle fitness f_{it};
      select p_{best} and g_{best};
      Remember the value and the iteration for g_{best};
          if $f_{it} < p_{best}$, **then** $p_{best} = f_{it}$
          else $p_{best} = p_{best}$
          if $p_{best} < g_{best}$, **then** $g_{best} = p_{best}$
          else $g_{best} = g_{best}$
      end
      $t = t + 1$
  End
//Path smoothing
    define control points;
    Use cubic B-spline function to fit the control point and generation of a smooth path;
//Output
   Output the planned path.

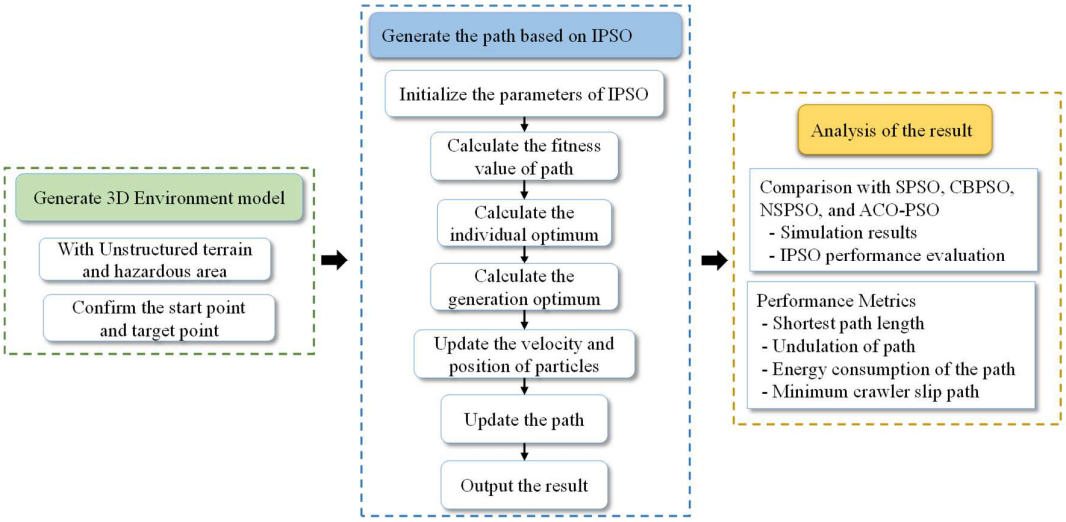

Figure 7. Framework for the numerical simulation.

4. IPSO for Function Optimization

To provide a lucid illustration, we introduce a number of benchmark functions to evaluate the efficacy of the IPSO. In addition, to establish a comparative analysis, we incorporate three advanced PSO methodologies, namely CBPSO (chaos-based initialization particle swarm optimization) [45], NSPSO (non-dominated sorting particle swarm optimization) [46], ACO-PSO (ant colony and particle swarm hybrid algorithm) [7] and SPSO (standard particle swarm optimization). Twelve well-known benchmark functions are selected [47]: Sphere, Tablet, Rosenbrock, BentCigar, Rastrigin, Griewank, Ackley, Schaffer, Kowalik, Six-Hump Camel, Branin, and Hartman. The details are shown in Table 3, where $f_i(x), i = 1, 2, 3, 4$ are non-linear unimodal functions, $f_i(x), i = 5, 6, 7, 8$ are non-linear multimodal functions and $f_i(x), i = 9, 10, 11, 12$ are dimension-fixed functions used to test the ability of the algorithm to solve the simple problem. These functions are all optimization problems for which there is a global minimum value.

In every scenario, this study performs a total of 50 independent trials for each benchmark function, with algorithmic parameters detailed in Table 4. The metrics employed to evaluate algorithmic performance include the optimal value, the average value, and the standard deviation, each of which is summarized in Table 5. Specifically, the optimal value serves as an indicator of the algorithm's search capability, while the average value provides a measure of its convergence accuracy. The standard deviation, meanwhile, offers insights into the algorithm's stability across multiple runs.

Table 3. Benchmark functions.

Figure	Formulation	Range	Optimum/ Minimum	Search Ability
Sphere	$f_1(x) = \sum_{i=1}^{D} x_i^2$	$[-100, 100]$	[0, 0, 0,..., 0]/0	Local
Tablet	$f_2(x) = (1000 * x_1)^2 + \sum_{i=2}^{D} x_i^2$	$[-100, 100]$	[0, 0, 0,..., 0]/0	Local
Rosenbrock	$f_3(x) = \sum_{i=1}^{D-1} (100(x_i^2 - x_{i+1}))^2 + (x_i - 1)^2$	$[-100, 100]$	[1, 1, 1,..., 1]/0	Local
BentCigar	$f_4(x) = x_1^2 + 10^6 \sum_{i=2}^{D} x_i^2$	$[-100, 100]$	[1, 1, 1,..., 1]/0	Local
Rastrigin	$f_5(x) = \sum_{i=1}^{D} (x_i^2 - 10\cos(2\pi x_i) + 10)$	$[-5.12, 5.12]$	[0, 0, 0,..., 0]/0	Global
Griewank	$f_6(x) = \sum_{i=1}^{D} x_i^2/4000 - \prod_{i=1}^{D} \cos(x_i/\sqrt{i}) + 1$	$[-600, 600]$	[0, 0, 0,..., 0]/0	Global
Ackley	$f_7(x) = -20\exp(-0.2\sqrt{\sum_{i=1}^{D} x_i^2/n})$	$[-32, 32]$	[0, 0, 0,..., 0]/0	Global

Table 3. *Cont.*

Figure	Formulation	Range	Optimum/ Minimum	Search Ability
Schaffer	$f_8(x) = 78.33 + \sum_{i=1}^{D} \frac{x_i^4 - 16x_i^2 + 5x_i}{D}$	$[-500, 500]$	[420.96, 420.96, 420. 96,..., 420.96]/0	Global
Kowalik	$f_9(x) = \sum_{i=1}^{11} [a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}]^2$	$[-5, 5]$	0.00030	-
Six-Hump Camel	$f_{10}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + x_2^4$	$[-5, 5]$	-1.0316	-
Branin	$f_{11}(x) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos x_1 + 10$	$[-5, 5]$	0.398	-
Hartman	$f_{12}(x) = -\sum_{i=1}^{4} c_i \exp(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2)$	$[0, 1]$	-3.32	-

Table 4. Parameter for improved PSOs.

PSO Kind	Parameter
IPSO	M = 30, $t_{\max}$ = 300, c_{1_ini} = 2,c_{1_fin} = 0.5, c_{2_ini} = 2,c_{2_fin} = 0.5, $w_{\max}$ = 0.9, $w_{\min}$ = 0.4, p = 0.05
SPSO	M = 30, $t_{\max}$ = 300, $c_1 = c_2 = 2$, $w = 0.5$
CBPSO	M = 30, $t_{\max}$ = 300, $c_1 = c_2 = 2$, $w_{\max} = 0.9$, $w_{\min} = 0.4$, $\mu = 4$
NSPSO	M = 30, $t_{\max}$ = 300, $c_1 = c_2 = 2$, $w_{\max} = 0.9$, $w_{\min} = 0.4$
ACO-PSO	M = 30, $t_{\max}$ = 300, $c_1 = c_2 = 2$, $w = 0.5$, $\alpha = 1$, $\beta = 5$, $\rho = 0.5$

Table 5. Experiment results of benchmark functions.

Functions	Statistics	IPSO	SPSO	CBPSO	NSPSO	ACO-PSO
Sphere	Best value	$\mathbf{3.03 \times 10^{-37}}$	1.03×10^{-21}	1.40×10^{-28}	1.30×10^{-13}	1.22×10^{-13}
	Mean value	$\mathbf{0.37 \times 10^{-35}}$	5.55×10^{-20}	4.89×10^{-25}	5.51×10^{-3}	8.25×10^{-2}
	St. Var	$\mathbf{1.61 \times 10^{-70}}$	3.26×10^{-38}	3.46×10^{-27}	7.16×10^{-1}	1.04×10^{-1}
Tablet	Best value	$\mathbf{1.25 \times 10^{-27}}$	1.09×10^{-7}	3.21×10^{-27}	1.20×10^{-13}	1.00×10^{-12}
	Mean value	2.73×10^{-3}	1.43×10^{5}	$\mathbf{9.78 \times 10^{-24}}$	4.49×10^{-2}	1.84×10^{-3}
	St. Var	1.43×10^{-2}	2.63×10^{6}	$\mathbf{6.92 \times 10^{-26}}$	2.5×10^{-2}	3.2×10^{-3}
Rosenbrock	Best value	$\mathbf{2.17 \times 10^{-15}}$	2.73×10^{-1}	1.21×10^{-8}	9.91×10^{-5}	6.54×10^{-6}
	Mean value	$\mathbf{8.91 \times 10^{-9}}$	1.38×10^{3}	3.51×10^{-6}	9.45×10^{-3}	5.32×10^{-5}
	St. Var	$\mathbf{7.94 \times 10^{-16}}$	4.95×10^{7}	2.48×10^{-10}	1.32×10^{-5}	2.35×10^{-5}
BentCigar	Best value	2.19×10^{-9}	1.45×10^{-18}	$\mathbf{1.56 \times 10^{-22}}$	1.25×10^{-13}	1.31×10^{-14}
	Mean value	1.25×10^{-8}	4.59×10^{-10}	$\mathbf{4.20 \times 10^{-19}}$	6.03×10^{-2}	-8.28×10^{-2}
	St. Var	3.47×10^{-16}	6.16×10^{-7}	$\mathbf{2.97 \times 10^{-21}}$	1.83×10^{-3}	1.47×10^{-1}
Rastrigin	Best value	$\mathbf{8.61 \times 10^{-13}}$	5.38×10^{-5}	3.61×10^{-7}	1.54×10^{-10}	1.00×10^{-8}
	Mean value	$\mathbf{1.98 \times 10^{-12}}$	2.17×10^{-4}	4.11×10^{-5}	-3.24×10^{-2}	-1.39×10^{-1}
	St. Var	$\mathbf{7.16 \times 10^{-24}}$	5.46×10^{-2}	2.21×10^{-5}	2.26×10^{-3}	3.64×10^{-1}
Griewank	Best value	$\mathbf{1.13 \times 10^{-14}}$	1.34×10^{-12}	5.42×10^{-8}	3.03×10^{-4}	3.12×10^{-4}
	Mean value	$\mathbf{3.45 \times 10^{-2}}$	6.85×10^{1}	3.21×10^{-5}	3.21×10^{-4}	4.42×10^{-3}
	St. Var	$\mathbf{1.92 \times 10^{-3}}$	2.48×10^{1}	5.22×10^{-4}	1.16×10^{-2}	3.73×10^{-3}
Ackley	Best value	8.11×10^{-8}	$\mathbf{3.77 \times 10^{-14}}$	4.44×10^{-6}	1.67×10^{-13}	1.00×10^{-12}
	Mean value	$\mathbf{2.80 \times 10^{-7}}$	-1.97×10^{-1}	1.19×10^{-4}	-1.37×10^{-1}	3.34×10^{-3}
	St. Var	$\mathbf{5.04 \times 10^{-14}}$	3.61×10^{-1}	8.13×10^{-12}	1.78×10^{-1}	2.92×10^{-2}
Schaffer	Best value	$\mathbf{5.32 \times 10^{-9}}$	2.96×10^{-5}	3.98×10^{-2}	2.30×10^{-3}	1.00×10^{-6}
	Mean value	$\mathbf{2.16 \times 10^{-7}}$	1.14×10^{-4}	3.97×10^{-2}	2.30×10^{-3}	1.55×10^{-1}
	St. Var	$\mathbf{4.51 \times 10^{-14}}$	1.39×10^{-3}	6.09×10^{-1}	2.90×10^{-11}	1.13×10^{-1}
Kowalik	Best value	$\mathbf{3.02 \times 10^{-4}}$	9.171×10^{-4}	2.12×10^{-4}	8.31×10^{-4}	6.32×10^{-4}
	Mean value	$\mathbf{2.75 \times 10^{-3}}$	3.87×10^{-3}	1.21×10^{-2}	3.78×10^{-3}	4.12×10^{-3}
	St. Var	$\mathbf{6.23 \times 10^{-3}}$	8.26×10^{-3}	6.02×10^{-2}	7.54×10^{-1}	5.31×10^{-1}
Six-Hump Camel	Best value	$\mathbf{-1.0316}$	-1.0316	-1.03149	-1.03159	-1.03152
	Mean value	$\mathbf{-1.0316}$	-1.0313	-1.03162	-1.03172	-1.03163
	St. Var	$\mathbf{2.13 \times 10^{-14}}$	1.23×10^{-10}	3.56×10^{-8}	5.02×10^{-9}	6.20×10^{-11}
Branin	Best value	**0.3978**	0.4537	0.4378	0.4030	0.3997
	Mean value	**0.3978**	0.4568	0.4062	0.4130	0.4183
	St. Var	$\mathbf{1.05 \times 10^{-16}}$	6.32×10^{-9}	8.21×10^{-10}	6.55×10^{-9}	8.11×10^{-11}
Hartman	Best value	$\mathbf{-3.3204}$	-3.2863	-3.2504	-3.3631	-3.2932
	Mean value	$\mathbf{-3.321}$	-3.3239	-3.3352	-3.3936	-3.3632
	St. Var	**0.4546**	0.7728	0.8654	1.1228	0.9652

As demonstrated in Table 6, the IPSO algorithm exhibits superior performance for six out of twelve benchmark functions, with respect to both the best and mean values. This illustrates that the IPSO algorithm surpasses the SPSO, CBPSO, NSPSO and ACO-PSO algorithms in terms of local and global search capabilities. Additionally, the IPSO algorithm showcases exceptional performance in six out of eight benchmark functions with regard to the standard deviation, which indicates that the IPSO has better robustness and stability. Consequently, it can be inferred that the IPSO algorithm has stronger features as compared to the other five algorithms when it comes to convergence, diversity, and distribution aspects.

Table 6. Comparison of significance level results between IPSO and others algorithm.

Pair of Algorithms	*p*-Value
IPSO vs. SPSO	0.012
IPSO vs. CBPSO	0.023
IPSO vs. NSPSO	0.012
IPSO vs. ACO-PSO	0.024

To further evaluate the performance of these optimization algorithms, parametric statistical tests are performed using the Wilcoxon rank sum test in this paper. From a statistical point of view, the test is robust since it does not assume normal distributions [48]. Firstly, eight benchmark functions ($f_i(x), i = 1, 2, \ldots, 8$) are used to test the five algorithms (IPSO, SPSO, CBPSO, NSPSO, and ACO-PSO), each benchmark function corresponds to five results obtained by each of the five algorithms. These results are analyzed using Wilcoxon rank sum test to compare the Best Value of the different algorithms for the same benchmark function. Under normal conditions, the significance level is set to $\alpha = 0.5$. When $p < 0.05$, there is a significant difference between them, but when $p > 0.05$, there no significant difference is observed. Table 6 shows p-value computed for all the pairwise comparisons concerning IPSO (the p-values have been computed by using SPSS). As the table states, all the $p < 0.05$, that IPSO shows a significant improvement in SPSO, CBPSO, NSPSO, and ACO-PSO.

In summary, IPSO performs well in optimizing complex high-dimensional functions, which not only has strong local search ability and global search ability, but also has strong robustness. Therefore, this paper adopts the IPSO algorithm to solve the three-dimensional path planning of DSMV on the unstructured seabed topography.

5. Simulation Evaluation

In this section, to evaluate the path planning performance of IPSO, we carry out several comparisons between different important indicators, such as solution optimality, convergence speed, and computation time for a given algorithm. The simulation results for the different PSOs (i.e., SPSO, CBPSO, NSPSO, and ACO-PSO) are summarized. All the comparison results clearly illustrate the effectiveness and advantages of the proposed improvements. The present simulations are performed on an Intel(R) Core (TM) i7–12700H 2.30 GHz, on-board RAM 24.0GB, Nvidia GeForce RTX 3060 Laptop GPU, MATLAB-2022b under win11 platform Compile and simulate.

5.1. Simulation Parameters

When path planning in the simulation environment, in order to simplify the calculation, DSMV is regarded as a particle, ignoring the force of the crawler and the submarine water current on the DSMV. The size of the three-dimensional simulation environment is set to [200 200 10]m, the underwater sensing equipment is installed on the DSMV main frame and the vertical distance from the seabed datum is 0.5~1 m. The initial location point of the DSMV is set to [1 1 0.5], the goal point is [200 200 0.5], and with 3 random control points, where the control points are considered as particles. The free space is regarded as the feasible region for each particle and as the solution space. According to

Equations (1)–(3), different specifications and hazardous areas are set in the traffic area, and their coordinates in the $x-y$ plane are shown in Table 7. To ensure that the results are reasonable, the maximum number of iterations of the algorithm is set to $t_{\max}$ = 100, and the population size is set to M = 50. Other parameter settings of the IPSO, SPSO, CBPSO, NSPSO, and ACO-PSO are shown in Table 8. Five algorithms are applied in order to calculate the best route of the DSMV from the initial point to the goal point. A comparison between is made in terms of path length, path undulation, minimum crawler slip path, energy consumption, convergence time and the difference between the optimal values for the five algorithms. When the slippage rete between the crawler and the sediment is greater than 0.5, the mining vehicle cannot pass through, and this area is regarded as a hazardous area. When calculating the energy consumption, the DSMV related parameters refer to those of the "Pioneer No. 1" of Shanghai Jiao Tong University. The speed of the DSMV is set to a constant 1m/s, the energy consumption of the acquisition system P_1 is equal to 90KW, and the energy consumption of the traveling system P_2 is equal to 50KW, the energy consumption of the control and perception system P_3 is equal to 5KW, the mining time accounts for 60% of the total time, and the traveling time accounts for 40%. Finally, the differences and applicability of each algorithm are analyzed and discussed.

Table 7. Center points of the obstacles in the x–y plane.

	Obstacle Point					Hazard Point
Point No.	p_1	p_2	p_3	p_4	p_5	p_a
(x, y)	(28.32,70.58)	(29.22,109.53)	(51.05,129.46)	(89.84,131.10)	(148.62,149.23)	(148.62,149.23)
Point No.	p_6	p_7	p_8	p_9	p_{10}	p_b
(x, y)	(68.10,89.45)	(108.56,72.03)	(189.85,111.53)	(171.55,69.85)	(160.13,79.56)	(160.13,79.56)

Table 8. Parameters of the algorithms.

PSO Kind	Parameter
IPSO	M = 50, $t_{\max}$ = 100, c_{1_ini} = 2,c_{1_fin}= 0.5, c_{2_ini} = 2,c_{2_fin} = 0.5, $w_{\max}$ = 0.9, $w_{\min}$ = 0.4, p = 0.05
SPSO	M = 50, $t_{\max}$ = 100, $c_1 = c_2$ = 2, w = 0.5
CBPSO	M = 50, $t_{\max}$ = 100, $c_1 = c_2$ = 2, $w_{\max}$ = 0.9, $w_{\min}$ = 0.4, $\mu = 4$
NSPSO	M = 50, $t_{\max}$ = 100, $c_1 = c_2$ = 2, $w_{\max}$ = 0.9, $w_{\min}$ = 0.4
ACO-PSO	M = 50, $t_{\max}$ = 100, $c_1 = c_2$ = 2, w = 0.5, $\alpha = 1$, $\beta = 5$, $\rho = 0.5$

5.2. Analysis of Simulation Results

Five optimization algorithms are considered, all of which are intended to provide three-dimensional path planning for the DSMV. Their objectives are accordingly to avoid hazardous regions and areas with a high crawler slippage rate, to identify the shortest route, to support minimum terrain relief, and to facilitate minimum energy consumption. Figure 8 depicts the path planning simulation results for five algorithms, where the blue circular cylinders represent the hazardous area due to exceeding the threshold for the DSMV crawler's slip rate. Figure 9 illustrates the iterative progression of the algorithm.

Figure 8 provides a comprehensive comparative analysis of paths generated by multiple optimization algorithms—IPSO, SPSO, CBPSO, NSPSO, and ACO-PSO—within a simulated three-dimensional environment. The figure elucidates the spatial distribution of the generated trajectories across the x, y, z coordinate axes. Among the evaluated algorithms, the IPSO-generated path (Figure 8a) is characterized by its remarkable geometric smoothness and angular stability in the heading direction, eliminating the necessity for abrupt maneuvering. In contrast, the paths generated by SPSO and NSPSO algorithms (Figure 8b,d) are predominantly situated along the environmental periphery and are marked by pronounced angular turns, further complicated by substantial fluctuations in heading angles at midpoints. A noticeable large-angle turn is also evident in the NSPSO-generated path near the second control point, as depicted in Figure 8e. A multi-metric

evaluation further reveals discernible differences in path quality, including criteria such as the cumulative distance traversed, the frequency of hazardous regions encountered, and topographical fluctuations. The trajectories optimized by IPSO, CBPSO, and ACO-PSO (Figure 8a,c,e) are noticeably smoother and less topographically undulating, indicating their enhanced suitability for deep-sea mining vehicle (DSMV) navigation. Specifically, the path delineated in Figure 8a outperforms its counterparts in Figure 8c,e, underlining the superior optimization capabilities of the IPSO algorithm. Conversely, the paths generated by SPSO and NSPSO (Figure 8b,d) manifest longer lengths and greater topographical variance, highlighting their relatively inferior capacity for efficient path optimization. These observations implicitly question the efficacy of SPSO and NSPSO algorithms in generating optimal navigational paths compared to their algorithmic counterparts.

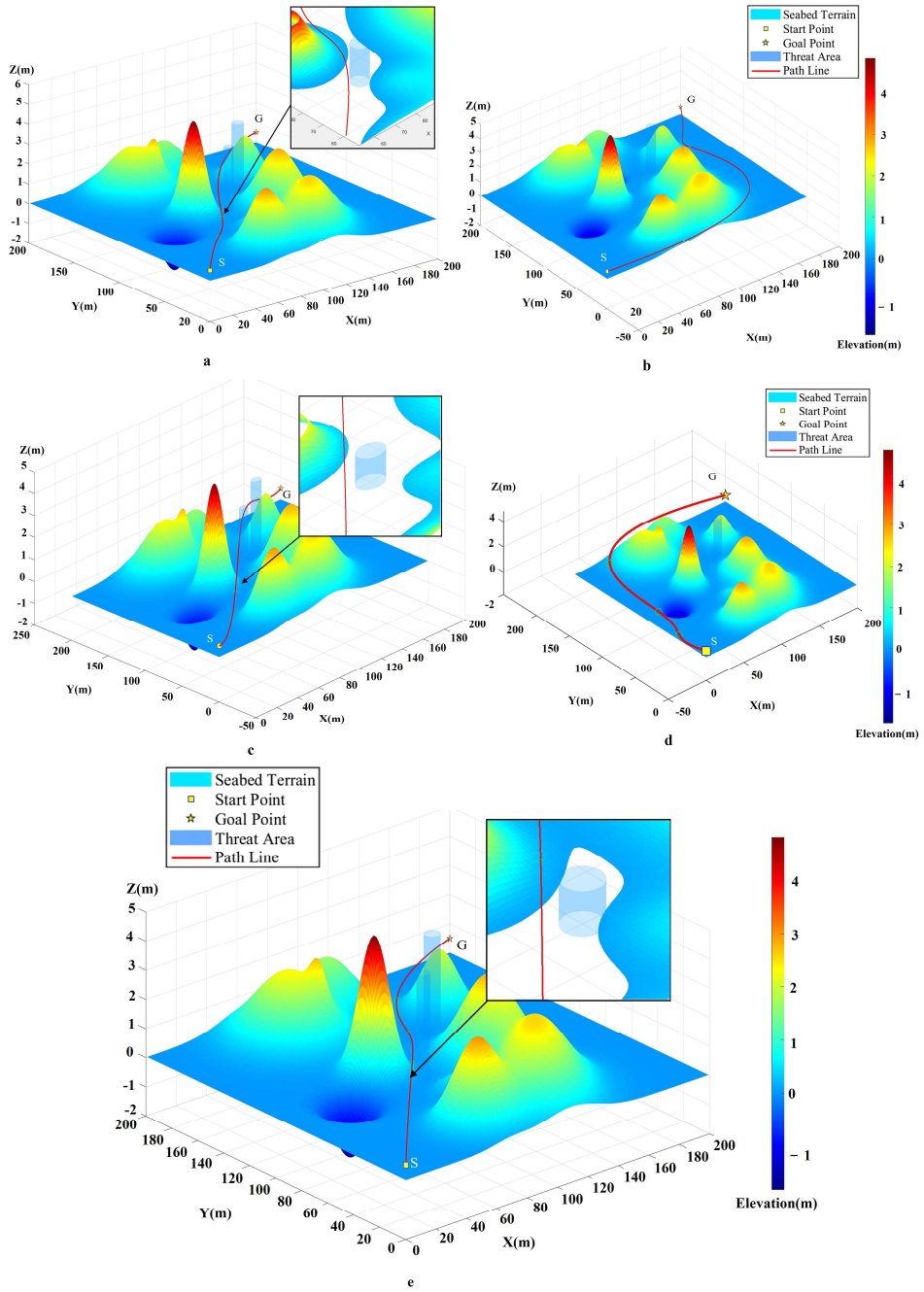

Figure 8. The paths resulting from the five different algorithms applied to the simulated environment; (**a**) IPSO; (**b**) SPSO; (**c**) CBPSO; (**d**) NSPSO; (**e**) ACO-PSO.

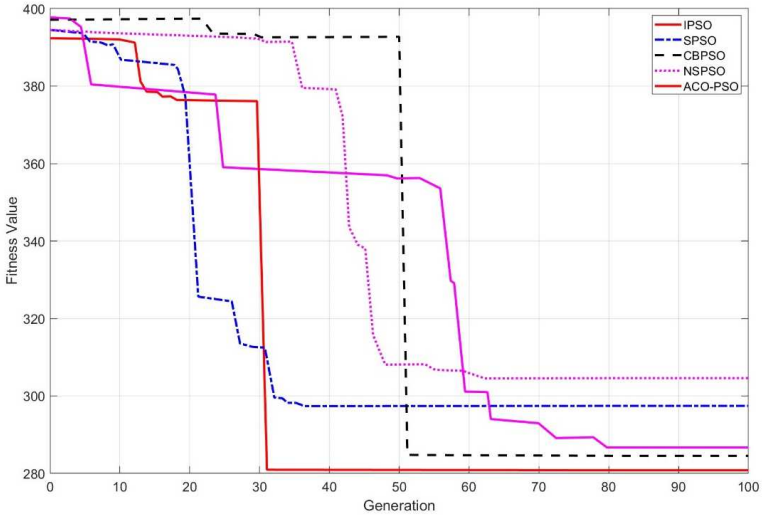

Figure 9. Iterative graph of the five algorithms (IPSO, SPSO, CBPSO, NSPSO, and ACO-PSO) when applied to the path optimization.

This multi-faceted analysis serves not only to validate the computational robustness of the IPSO algorithm but also to lay the foundation for future research aiming to address more complex navigational scenarios involving dynamic constraints and uncertainties.

Figure 9 shows the iterative convergence diagrams of the five algorithms. The IPSO algorithm demonstrates rapid overall convergence, reaching the optimal solution within the 31st generation and maintaining stability at this optimum, which indicates strong robustness. Combining the algorithm's performance with the path plot in Figure 8a reveals its notable optimization capabilities. Although the SPSO algorithm exhibits a relatively high convergence rate in the initial stages, its overall convergence speed is slower than that of the IPSO. Both the CBPSO and NSPSO algorithms begin to converge around the 50th generation, exhibiting a slower overall convergence rate. However, they ultimately achieve and maintain the optimal solution, demonstrating satisfactory robustness. The ACO-PSO algorithm exhibits the slowest convergence speed, requiring approximately 80 generations to approach the optimal solution, and its stability is comparatively inferior.

The results of the path analysis conducted by application of the IPSO, SPSO, CBPSO, NSPSO, and ACO-PSO algorithms for the simulated environment are illustrated in Figure 10.

The sub-Figures (a), (b), and (c) of Figure 10 provide the optimal value, the difference from the theoretical value, and the percentwise difference with respect to the theoretical value for each algorithm. It is found that the IPSO algorithm produces the shortest path, with differences of 14.692 m, 1.892 m, 21.682 m, and 4.122 m shorter than those of the SPSO, CBPSO, NSPSO, and ACO-PSO algorithms, respectively. Moreover, the path generated by the IPSO algorithm is the closest to the theoretical optimal value (The theoretical optimal path is 281.4285 m). The convergence time of the algorithms to the optimal value is demonstrated in Figure 10d. The results reveal that the IPSO algorithm outperforms the other algorithms in terms of convergence rate, with a convergence time that is 14.7s shorter than that of the SPSO algorithm, 5.72 s shorter than that of the CBPSO algorithm, 17.29 s shorter than that of the NSPSO algorithm, and 16.34 s shorter than that of the ACO-PSO algorithm. In Figure 10e, the path undulation is depicted, and the results reveal that the paths generated by all five algorithms exhibit minimal undulation. Specifically, the overall path undulation of IPSO, SPSO, CBPSO, and NSPSO is less than 1m, which satisfies the requirements for DSMV traffic. In Figure 10f, the overall energy consumption is presented, and the results indicate that IPSO generates the path with the lowest energy consumption for the DSMV, with reductions of 18.25%, 7.99%, 20.79%, and 19.88% compared to SPSO, CBPSO, NSPSO, and ACO-PSO, respectively. These findings suggest that utilizing the IPSO

algorithm for DSMV path planning can lead to significant energy savings and promote environmental sustainability.

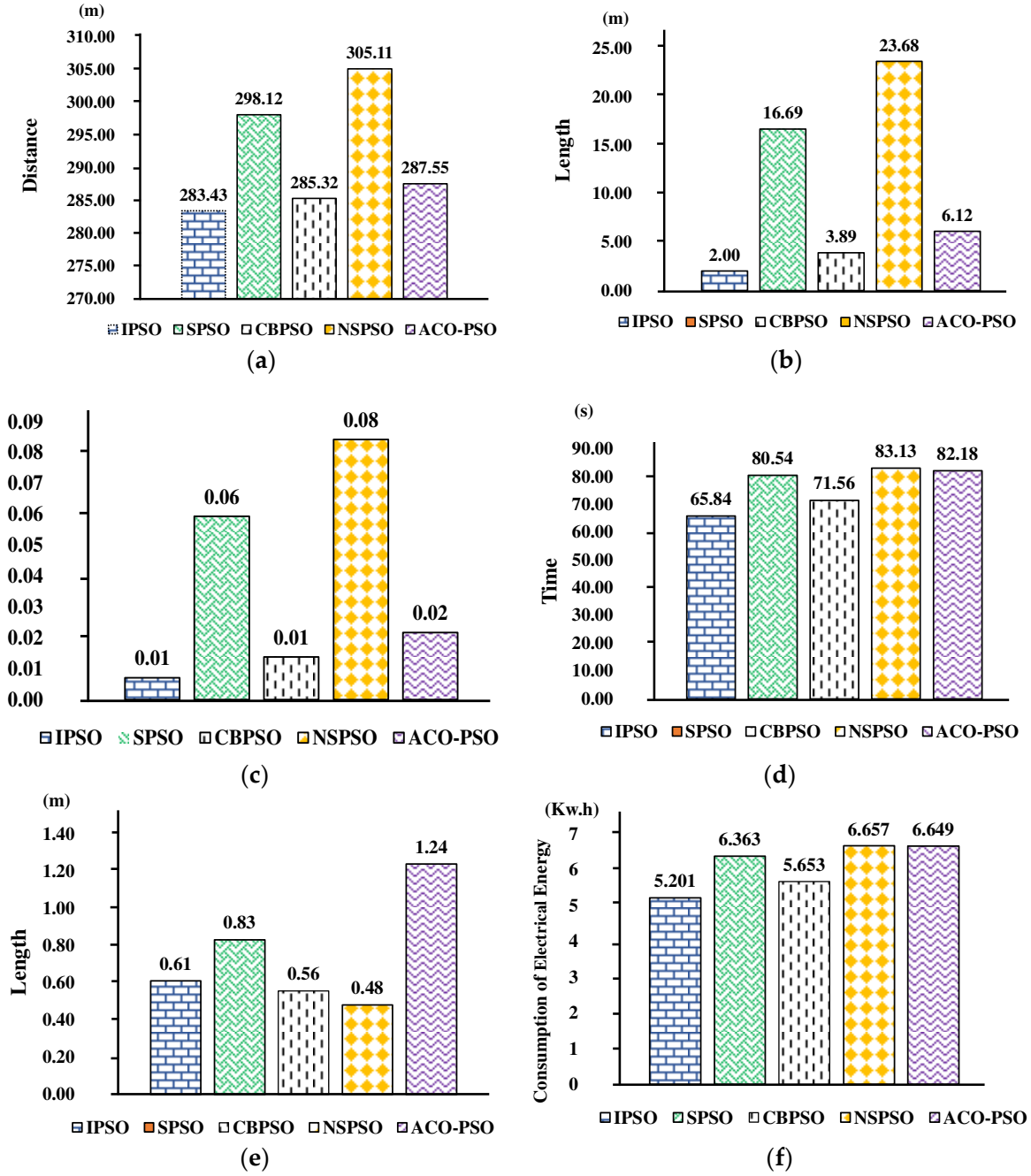

Figure 10. Comparison of path planning results for the five algorithms. (**a**) Best value of path length; (**b**) difference from optimal theoretical value; (**c**) percentage difference from optimal value; (**d**) commutation time; (**e**) fluctuation; (**f**) energy consumption.

Table 9 illustrates the best value, the mean value, and the standard deviation for the paths obtained by the different algorithms after 10 iterations. The results indicate that the IPSO algorithm outperforms the other algorithms in terms of both path optimization ability and stability, as evidenced by its performance with respect to the best value, the mean value, and the standard deviation.

Table 9. Calculation results and contrast for the five algorithms.

Algorithm	Best Value	Mean Value	St. Dev.
IPSO	283.428	284.232	0.026
SPSO	298.12	302.679	3.448
CBPSO	285.32	287.154	0.636
NSPSO	305.11	306.488	5.451
ACO-PSO	287.55	290.251	1.181

To sum up, when compared with other algorithms, they are outperformed by IPSO in terms of path planning speed, path accuracy, and convergence speed, making it the preferred algorithm for path planning of the DSMV. The potential of the IPSO algorithm in order to enhance the efficiency and safety of mining operations is highlighted, and this approach can significantly contribute to the advancement of mining automation and optimization.

6. Conclusions

The purposes of path planning are to minimize the energy consumption and execution time, to avoid hazardous regions and to obtain an optimal or suboptimal path from the mining area to the storage base within the context of the novel mining system. In this study, an improved particle swarm optimization algorithm is proposed for achieving three-dimensional path planning for DSMVs. In relation to the accuracy and effectiveness of the IPSO, compared with other algorithms, it is found that the IPSO has stronger overall optimization and search ability. The experimental results indicate that the IPSO algorithm is more robust, and that it possesses better convergence efficiency and higher precision with respect to the solution. Furthermore, the IPSO algorithm is a practical method to obtain the best optimized path and has certain benefits including stability properties for solving the challenging task of three-dimensional path planning of DSMVs.

The next stage of this work is to improve the practicability of the current algorithm in realistic and complex deep-sea mining environments and consider the introduction of the dynamic and kinematic equations of the mining vehicles to make the planned path more applicable to actual usage scenarios. The mining area is composed of random obstacles, irregular terrain as well as soft and sparse sediments. Therefore, a natural extension of the above work is to develop an effective path planner that can integrate current forecast information and be able to accommodate planning a complex and variable seabed mining missions.

Author Contributions: C.L., conceptualization, writing—reviewing and editing, methodology, investigation, software, and writing—original and draft. J.Y., supervision, funding, and acquisition. B.J.L., visualization and writing—reviewing and editing. Q.C., writing—original and draft, software, S.W., formal analysis and data curation. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Hainan Provincial Joint Project of Sanya Yazhou Bay Science and Technology City grant number 2021JJLH0001 and Science and Technology Committee Shanghai Municipality grant number 19DZ1207300.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: The research was supported by the Hainan Provincial Joint Project of Sanya Yazhou Bay Science and Technology City (2021JJLH0001), the Science and Technology Committee Shanghai Municipality(19DZ1207300), the China Scholarship Council and the Major Projects of Strategic Emerging Industries in Shanghai. The authors are grateful for the financial support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kang, Y.; Liu, S.; Zou, W.; Zhao, H.; Hu, X. Design and analysis of an innovative deep-sea lifting motor pump. *Appl. Ocean Res.* **2018**, *82*, 22–31. [CrossRef]
2. Wu, Q.; Yang, J.; Lu, H.; Lu, W.; Liu, L. Effects of heave motion on the dynamic performance of vertical transport system for deep sea mining. *Appl. Ocean Res.* **2020**, *101*, 102188. [CrossRef]
3. Leng, D.; Shao, S.; Xie, Y.; Wang, H.; Liu, G. A brief review of recent progress on deep sea mining vehicle. *Ocean Eng.* **2021**, *228*, 108565. [CrossRef]
4. Yang, J.; Liu, L.; Lyu, H.; Lin, Z. Deep-Sea Mining Equipment in China: Current Status and Prospect. *Chin. J. Eng. Sci.* **2020**, *22*, 1–9. [CrossRef]
5. Patle, B.K.; Pandey, A.; Parhi, D.R.K.; Jagadeesh, A.J.D.T. A review: On path planning strategies for navigation of mobile robot. *Def. Technol.* **2019**, *15*, 582–606. [CrossRef]
6. Mathai, T.; Rajarama, K.N.; Kumar, S.; Chandran, M.S.; Kumar, M.R.A. Geotechnical aspects of clayey sediments off Badagara on the Kerala Coast, India. *Mar. Georesources Geotechnol.* **2012**, *30*, 180–193. [CrossRef]
7. Shi, C.; Bu, Y.; Li, Z. Path Planning for Deep Sea Mining Robot Based on ACO-PSO Hybrid Algorithm. In Proceedings of the 2008 International Conference on Intelligent Computation Technology and Automation (ICICTA), Changsha, China, 20–22 October 2008; pp. 125–129.
8. Park, S.J.; Yeu, T.K.; Yoon, S.M.; Hong, S.; Sung, K.Y. A study of sweeping coverage path planning method for deep-sea manganese nodule mining robot. In Proceedings of the OCEANS'11 MTS/IEEE KONA, Waikoloa, Hawaii, USA, 19–22 September 2011; pp. 1–5.
9. Dai, Y.; Liu, S.-J. Theoretical design and dynamic simulation of new mining paths of tracked miner on deep seafloor. *J. Cent. South Univ.* **2013**, *20*, 918–923. [CrossRef]
10. Jiang, Y.; Jiang, Z. Simulation of Route Planning for Deep Sea Mining Vehicle Based on Improved Ant Colony Algorithm. In Proceedings of the 2017 International Conference on Computer Technology, Electronics and Communication (ICCTEC), Dalian, China, 19–21 December 2017; pp. 319–323.
11. Chen, Q.; Yang, J.; Mao, J.; Liang, Z.; Lu, C.; Sun, P. A path following controller for deep-sea mining vehicles considering slip control and random resistance based on improved deep deterministic policy gradient. *Ocean Eng.* **2023**, *278*, 114069. [CrossRef]
12. Hao, K.; Zhao, J.; Li, Z.; Liu, Y.; Zhao, L. Dynamic path planning of a three-dimensional underwater AUV based on an adaptive genetic algorithm. *Ocean Eng.* **2022**, *263*, 112421. [CrossRef]
13. Yan, M.; Zhu, D.; Yang, S.X. A Novel 3-D bio-inspired neural network model for the path planning of an auv in underwater environments. *Intell. Autom. Soft Comput.* **2013**, *19*, 555–566. [CrossRef]
14. Zhu, D.; Liu, Y.; Sun, B. Task Assignment and path planning of a multi-AUV system based on a glasius bio-inspired self-organising map algorithm. *J. Navig.* **2018**, *71*, 482–496. [CrossRef]
15. Zeng, Z.; Lammas, A.; Sammut, K.; He, F.; Tang, Y. Shell space decomposition based path planning for AUVs operating in a variable environment. *Ocean Eng.* **2014**, *91*, 181–195. [CrossRef]
16. Yan, Z.; Zhang, J.; Zeng, J.; Tang, J. Three-dimensional path planning for autonomous underwater vehicles based on a whale optimization algorithm. *Ocean Eng.* **2022**, *250*, 111070. [CrossRef]
17. Xiong, C.; Chen, D.; Lu, D.; Zeng, Z.; Lian, L. Path planning of multiple autonomous marine vehicles for adaptive sampling using Voronoi-based ant colony optimization. *Robot. Auton. Syst.* **2019**, *115*, 90–103. [CrossRef]
18. Liu, J.; Anavatti, S.; Garratt, M.; Abbass, H.A. Modified continuous Ant Colony Optimisation for multiple Unmanned Ground Vehicle path planning. *Expert Syst. Appl.* **2022**, *196*, 116605. [CrossRef]
19. Li, Y.; Ma, T.; Chen, P.; Jiang, Y.; Wang, R.; Zhang, Q. Autonomous underwater vehicle optimal path planning method for seabed terrain matching navigation. *Ocean Eng.* **2017**, *133*, 107–115. [CrossRef]
20. Zhu, D.; Cao, X.; Sun, B.; Luo, C. Biologically inspired self-organizing map applied to task assignment and path planning of an Auv system. *IEEE Trans. Cogn. Dev. Syst.* **2017**, *10*, 304–313. [CrossRef]
21. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; pp. 1942–1948.
22. Wang, D.; Tan, D.; Liu, L. Particle swarm optimization algorithm: An overview. *Soft Comput.* **2018**, *22*, 387–408. [CrossRef]
23. Xue, B.; Zhang, M.; Browne, W.N. Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE Trans. Cybern.* **2012**, *43*, 1656–1671. [CrossRef]
24. Zhang, Y.; Gong, D.-W.; Zhang, J.-H. Robot path planning in uncertain environment using multi-objective particle swarm optimization. *Neurocomputing* **2013**, *103*, 172–185. [CrossRef]
25. Quan, H.; Srinivasan, D.; Khosravi, A. Particle swarm optimization for construction of neural network-based prediction intervals. *Neurocomputing* **2014**, *127*, 172–180. [CrossRef]
26. Das, P.K.; Behera, H.S.; Panigrahi, B.K. A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning. *Swarm Evol. Comput.* **2016**, *28*, 14–28. [CrossRef]
27. Tan, L. A clustering K-means algorithm based on improved PSO algorithm. In Proceedings of the 2015 Fifth International Conference on Communication Systems and Network Technologies, Gwalior, India, 4–6 April 2015; pp. 940–944.
28. Tharwat, A.; Elhoseny, M.; Hassanien, A.E.; Gabel, T.; Kumar, A. Intelligent Bézier curve-based path planning model using Chaotic Particle Swarm Optimization algorithm. *Clust. Comput.* **2019**, *22*, 4745–4766. [CrossRef]

29. Wang, Z.; Li, G.; Ren, J. Dynamic path planning for unmanned surface vehicle in complex offshore areas based on hybrid algorithm. *Comput. Commun.* **2020**, *166*, 49–56. [CrossRef]
30. Joo, J.; Kim, J.; Ko, Y.; Kim, S.-S.; Son, J.; Pak, S.J.; Ham, D.-J.; Son, S.K. Characterizing Geomorphological Properties of Western Pacific Seamounts for Cobalt-rich Ferromanganese Crust Resource Assessment. *Econ. Environ. Geol.* **2016**, *49*, 121–134. [CrossRef]
31. Nikolos, I.K.; Valavanis, K.P.; Tsourveloudis, N.C.; Kostaras, A.N. Evolutionary algorithm based offline/online path planner for UAV navigation. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **2003**, *33*, 898–912. [CrossRef]
32. Huang, S.; Tian, J.; Qiao, L.; Wang, Q.; Su, Y. Unmanned aerial vehicle path planning based on improved genetic algorithm. *J. Comput. Appl.* **2021**, *41*, 390.
33. Yang, X. Chapter 14—Multi-Objective Optimization. In *Nature-Inspired Optimization Algorithms*; Elsevier: Oxford, UK, 2014; pp. 197–211. ISBN 978-0-12-416743-8.
34. Chen, M.; Sun, F.H.; Jian, X.G.; Zhang, Z.M. Development of diamond-coated drills for high-speed machining SiC particles reinforced aluminum-matrix composite. *Key Eng. Mater.* **2003**, *259*, 853–857. [CrossRef]
35. Kim, J.-D.; Choi, I.-H. Micro surface phenomenon of ductile cutting in the ultrasonic vibration cutting of optical plastics. *J. Mater. Process. Technol.* **1997**, *68*, 89–98. [CrossRef]
36. Moriwaki, T.; Shamoto, E. Ultraprecision diamond turning of stainless steel by applying ultrasonic vibration. *CIRP Ann.* **1991**, *40*, 559–562. [CrossRef]
37. Lian, J.; Yu, W.; Xiao, K.; Liu, W. Cubic spline interpolation-based robot path planning using a chaotic adaptive particle swarm optimization algorithm. *Math. Probl. Eng.* **2020**, *2020*, 1–20. [CrossRef]
38. Zhang, Y.; Wang, S.; Ji, G. A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications. *Math. Probl. Eng.* **2015**, *2015*, 1–38. [CrossRef]
39. Huang, C.; Fei, J. UAV Path planning based on particle swarm optimization with global best path competition. *Int. J. Pattern Recognit.* **2018**, *32*, 1859008. [CrossRef]
40. Liu, Y.; Zhang, X.; Guan, X.; Delahaye, D. Adaptive sensitivity decision based path planning algorithm for unmanned aerial vehicle with improved particle swarm optimization. *Aerosp. Sci. Technol.* **2016**, *58*, 92–102. [CrossRef]
41. Zhao, Y.; Zu, W.; Zeng, H. A modified particle swarm optimization via particle visual modeling analysis. *Comput. Math. Appl.* **2009**, *57*, 2022–2029. [CrossRef]
42. Fernandez-Martinez, J.L.; Garcia-Gonzalo, E. Stochastic Stability Analysis of the Linear Continuous and Discrete PSO Models. *IEEE Trans. Evol. Comput.* **2010**, *15*, 405–423. [CrossRef]
43. Liang, J.J.; Qu, B.Y.; Gong, D.W.; Yue, C.T. Problem definitions and evaluation criteria for the CEC 2019 special session on multimodal multiobjective optimization. *Comput. Intell. Lab. Zhengzhou Univ.* 2019. [CrossRef]
44. Huang, C.; Zhao, Y.; Zhang, M.; Yang, H. APSO: An A*-PSO hybrid algorithm for mobile robot path planning. *IEEE Access* **2023**, *11*, 43238–43256. [CrossRef]
45. Shao, S.; Peng, Y.; He, C.; Du, Y. Efficient path planning for UAV formation via comprehensively improved particle swarm optimization. *ISA Trans.* **2020**, *97*, 415–430. [CrossRef]
46. Durillo, J.J.; García-Nieto, J.; Nebro, A.J.; Coello, C.A.C.; Luna, F.; Alba, E. Multi-objective particle swarm optimizers: An experimental comparison. In Proceedings of the Evolutionary Multi-Criterion Optimization: 5th International Conference, EMO 2009, Nantes, France, 7–10 April 2009; pp. 495–509.
47. Ebraheem, M.; Jyothsna, T.R. Comparative Performance Evaluation of Teaching Learning Based Optimization against Genetic Algorithm on Benchmark Functions. In Proceedings of the 2015 IEEE Power, Communication and Information Technology Conference (PCITC-2015), Bhubaneswar, India, 15–17 October 2015; pp. 327–331.
48. Derrac, J.; García, S.; Molina, D.; Herrera, F. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evol. Comput.* **2011**, *1*, 3–18. [CrossRef]