remote sensing

MDPI

Review

A Comprehensive Review of Path-Planning Algorithms for Planetary Rover Exploration

Qingliang Miao and Guangfei Wei *

Deep Space Exploration Laboratory, Institute of Deep Space Sciences, Hefei 230026, China; qlm2016@mail.ustc.edu.cn
* Correspondence: gfwei0554@gmail.com

Abstract: Path-planning algorithms for planetary rovers are critical for autonomous robotic exploration, enabling the efficient and safe traversal of complex and dynamic extraterrestrial terrains. Unlike terrestrial mobile robots, planetary rovers must navigate highly unpredictable environments influenced by diverse factors such as terrain variability, obstacles, illumination conditions, and temperature fluctuations, necessitating advanced path-planning strategies to ensure mission success. This review comprehensively synthesizes recent advancements in planetary rover path-planning algorithms. First, we categorize these algorithms from a constraint-oriented perspective, distinguishing between internal rover state constraints and external environmental constraints. Next, we examine rule-based path-planning approaches, including graph search-based methods, potential field methods, sampling-based techniques, and dynamic window approaches, analyzing representative algorithms in each category. Subsequently, we explore bio-inspired path-planning methods, such as evolutionary algorithms, fuzzy computing, and machine learning-based approaches, with a particular emphasis on the latest developments and prospects of machine learning techniques in planetary rover navigation. Finally, we synthesize key insights from existing algorithms and discuss future research directions, highlighting their potential applications in planetary exploration missions.

Keywords: planetary rovers; path-planning algorithms; graph search; machine learning; reinforcement learning

check for updates

Academic Editors: Roberto Opromolla and Vincenzo Capuano

Received: 25 March 2025
Revised: 27 May 2025
Accepted: 27 May 2025
Published: 31 May 2025

Citation: Miao, Q.; Wei, G. A Comprehensive Review of Path-Planning Algorithms for Planetary Rover Exploration. *Remote Sens.* **2025**, *17*, 1924. https://doi.org/10.3390/rs17111924

1. Introduction

Humanity has undertaken numerous lunar and deep space exploration endeavors, encompassing lunar orbital investigations [1], such as those conducted by the Chang'E-3 (CE-3) and Chang'E-4 (CE-4) missions [2–6], alongside manned lunar landings exemplified by the Apollo program [7]. Furthermore, Mars exploration initiatives, including landings and rover missions executed by the United States and China, have significantly contributed to our understanding of extraterrestrial environments [8–10]. Looking ahead, several nations are poised to enhance their capabilities in deep space exploration. Notably, the United States' Artemis program is set to resume crewed lunar landings and introduce new exploration technologies [11]. Concurrently, China's forthcoming Chang'E-7 (CE-7) mission aims to perform comprehensive investigations of the lunar south pole, thereby facilitating the advancement of lunar resource utilization and the establishment of research bases [4,12].

The deep space exploration missions that have been implemented or are about to be conducted in the future all require pre-launch path planning. During the actual exploration on the planetary surface, it is also necessary to integrate the surrounding environment to develop intelligent path-planning algorithms to improve mission efficiency and reduce risks. Within the framework of the extraterrestrial exploration programs formulated by various nations, planetary rovers are recognized as critical components for executing diverse exploration missions. These robotic systems are typically equipped with multiple scientific instruments and payloads to conduct comprehensive scientific exploration missions on the planetary surface. From a geological perspective, the planetary surface is characterized by craters, mountains, and plains, and it is covered by a layer of unconsolidated regolith similar to that of the Moon, thereby creating a complex environment for rover exploration. These combined factors present substantial challenges to rover mobility and mission operations, particularly in forthcoming lunar polar exploration missions such as the Artemis program and CE-7 mission. Consequently, implementing effective path-planning strategies is essential to ensure safe and efficient rover traversal. Such strategies enable rovers to avoid hazardous zones while considering operational constraints, including topographical features, energy budgets, and illumination conditions during target-oriented navigation.

Current path-planning methodologies for rovers predominantly implement a hybrid architecture combining "ground-based global planning and local path planning" [13,14]. The ground-based global path planning refers to the process wherein terrestrial control centers, guided by scientific mission objectives, generate comprehensive traversal routes from initial positions to target exploration sites through analysis of planetary surface topographic maps acquired by orbital satellites. These global paths are typically optimized through multi-criteria metrics, including the shortest path length, the least energy consumption, and the optimal lighting conditions, while concurrently incorporating environmental constraints such as terrain characteristics (e.g., slope, surface roughness), communication constraints, and rover-specific operational limitations [15,16]. However, global path planning depends on the planetary surface topography obtained by the orbiting satellite, which has low-resolution accuracy, rendering it incapable of resolving obstacle distributions encountered during actual rover traversal, and the dynamic environment of the planetary surface, such as real-time illumination and temperature. Therefore, it is difficult to effectively control the rover in real time, and one can only plan and guide the movement of the rover as a whole.

Local path planning is essentially reactive, whereas the most common type is fully autonomous perception-based path planning. During the rover's movement on the planetary surface, it dynamically adjusts the initially calculated results of the global path planning in real time based on the actual conditions of the planetary surface to deal with emergencies and unforeseen situations. Equipped with embedded intelligence and autonomous decision-making capabilities, the rover can reconstruct the terrain map through real-time onboard sensory perception of the ambient planetary environment and dynamically plan a local safe and reliable path according to its position and posture, as well as the identified surrounding terrain and obstacles [17,18]. For example, the Curiosity rover, the Opportunity rover, and China's Zhu Rong rover are all equipped with stereo cameras as environmental perception sensors, and they use the sum of absolute differences (SAD) algorithm to construct perception maps [19,20]. In addition to using the fully autonomous perception method for path planning mentioned above, Chen et al. [21] also referred to two other lower-level patrol detection modes, namely "direct drive of the mobile mechanism" and "blind walking mode". These methods have already been successfully applied to the autonomous path navigation of the Zhu Rong rover. However, the local path-planning algorithms also face multiple challenges. For example, they must operate autonomously without human intervention, and they need to plan paths in complex and partially unknown planetary surface environments while having the capability to replan when encountering obstacles or hazardous areas. Furthermore, given the rover's limited computing resources and energy supply, the designed local path-planning algorithms should avoid overly complex designs to ensure both high efficiency and accuracy [19].

Currently, researchers have proposed various types of path-planning algorithms for planetary rovers, such as the A* algorithm, which is suitable for global path planning [22], Rapidly exploring Random Tree (RRT) algorithm [23], genetic algorithms (GAs) [24,25], and algorithms suitable for local path planning like the dynamic window approach (DWA) [26], D* algorithm [9], and artificial potential field (APF) method [27,28], as well as improvements and combinations based on these methods. Additionally, with the rapid development of artificial intelligence, intelligent algorithms based on deep learning and reinforcement learning are gradually being applied to the path planning of planetary rovers [14,29–33]. Recent research has led to a significant increase in literature on planetary rover path planning. Researchers face challenges in selecting the most suitable path-planning algorithm under specific constraints. This review classifies and summarizes recent algorithms, serving as a reference for future researchers.

The organizational framework of this review is illustrated in Figure 1. Section 2 presents an overview of path-planning algorithms from the perspective of constraints, categorizing them into algorithms based on external environmental constraints (e.g., static and time-varying constraints) and those based on self-state constraints (e.g., patroller-specific and energy-related constraints). Section 3 focuses on rule-based path-planning algorithms, which are further classified into graph search-based approaches (such as Dijkstra's algorithm, A*-based algorithms, and D*-based algorithms), potential field methods (including the RRT* algorithm, FMM algorithm, and PRM algorithm), sampling-based techniques, and dynamic window-based methods. Section 4 highlights biologically inspired path-planning algorithms, encompassing evolutionary computation (e.g., genetic algorithms (GAs) and swarm optimization algorithms), fuzzy computation, and machine learning-based approaches (e.g., deep learning and reinforcement learning algorithms). Section 5 discusses the possible research directions and our future work. Finally, Section 6 provides a summary and conclusions of the various path-planning algorithms mentioned in this review.

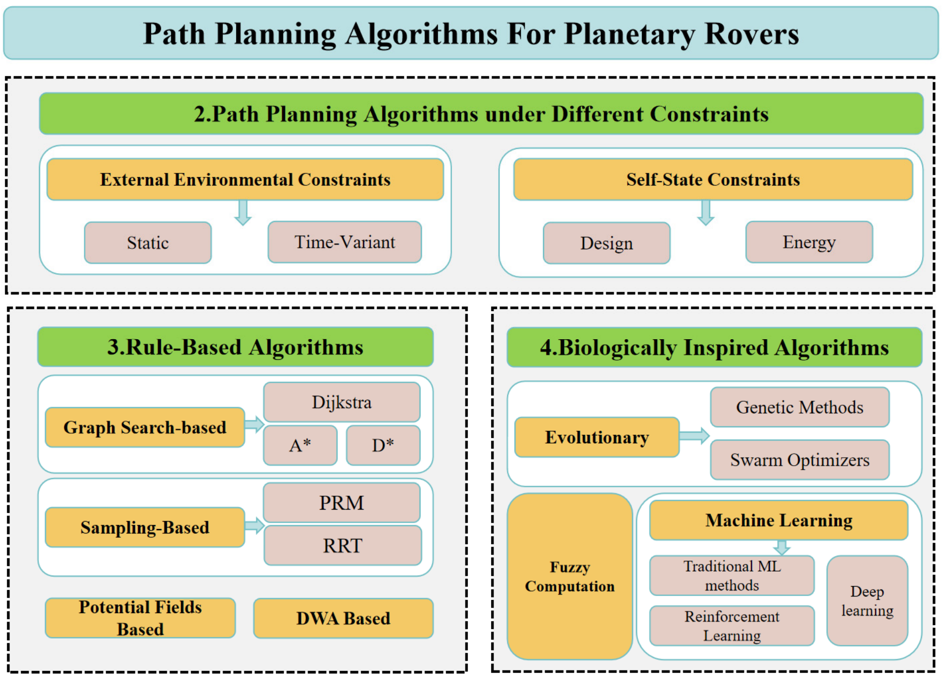

Figure 1. The organizational structure of this review.

2. Path-Planning Algorithms Under Different Constraints

During planetary rover operations on the surface of an extraterrestrial body, its mobility and functionality are constrained by multiple factors, including terrain complexity, lighting conditions, communication limitations, and power and energy availability. Therefore, in this complex and dynamic unknown environment, researching path-planning algorithms under multiple constraints is crucial for the rover to complete scientific mission objectives (such as scientific exploration and sample collection), improve operational efficiency, reduce energy consumption, and avoid potential dangers. Based on whether the constraints are caused by external environmental factors or the rover's factors, path-planning algorithms can be classified into algorithms based on external environmental constraints and those based on the rover's self-state constraints. The following section provides a detailed introduction to these two major categories of path-planning algorithms, and some representative algorithms mentioned below are summarized in Table 1.

Table 1. Representative path-planning algorithms for planetary rovers under different constraints.

	External Environmental Constraints		Self-State Constraints		Concrete Algorithms
Publication	**Static Constraints**	**Time-Varying Constraints**	**Structural Design Factors**	**Resource Constraints**	
Bai et al. [34]	Slope	Lighting conditions	/	/	A*
Biesiadecki et al. [35]	Slope roughness	/	/	/	GESTALT
Chen et al. [36]	Terrain	/	Movement capabilities	/	Not mentioned
Iagnemma et al. [37]	Terrain roughness	/	Rover stability	/	A*
Sutoh et al. [15]	/	Insolation	Locomotion mechanism	/	Grid-based (e.g., Dijkstra's algorithm)
Cunningham et al. [38]	/	Communication availability, illumination	/	Energy constraints	A*
INOUE et al. [39]	Slope	Illumination, communication	/	/	ROBUST-STP3R
Brunner et al. [40]	Rough terrain	/	Robot–terrain interaction	/	Not mentioned
Farritor et al. [41]	/	/	Wheel slip, vehicle stability	Power	Genetic algorithm
Tanaka et al. [32]	Slope	Illumination, temperature	/	Thermal, power status	Reinforcement learning algorithm

2.1. Algorithms Based on External Environmental Constraints

Path-planning algorithms based on external environmental constraints can be further divided into static constraint algorithms and time-varying constraint algorithms [34], depending on whether the constraints change over time. In [34], the author discusses static constraints and dynamic factors separately, using slope and lighting conditions as examples to simulate the impact of different constraints on lunar rover path-planning results with the A* algorithm. The experimental results indicate that the path planned using dynamic lighting constraints can prevent the lunar rover from getting trapped in shadow areas, thereby avoiding mission failure.

Currently, researchers have studied various path-planning algorithms under different static constraint conditions. For example, the Grid-based Estimation of Surface Traversability (GESTALT) algorithm used by the Spirit and Opportunity rovers incorporates static constraint factors such as step suitability, slope suitability, and roughness suitability when planning paths [35,42]. China's Yutu and Yutu-2 lunar rovers, on the other hand, employ a local comprehensive obstacle avoidance path-planning method based on terrain traversal cost assessment, taking into account the characteristics of the lunar surface terrain and the movement capabilities of the rovers [36]. Based on a thorough consideration of modeling methods for local terrain roughness, reference [37] proposed a model-based path-planning algorithm suitable for rough terrain. Static constraints commonly encompass factors such as terrain slope [43], terrain roughness [37], terrain ruggedness index (TRI), and terrain position index (TPI) [34,44], among others. These static constraints are all calculated based on the digital elevation model (DEM). The literature [43] studies eight different methods for calculating terrain slope using DEM images of varying resolutions.

The time-varying constraints that affect the path planning of planetary rovers mainly include sunlight illumination and communication conditions [34]. For example, the lighting conditions on the lunar surface undergo periodic variations due to the Moon's rotational motion around the Earth. For lunar rovers powered by solar energy, lighting conditions are crucial for task execution, and they must move within areas that have sufficient sunlight to obtain an adequate energy supply. Bussey et al. [45] studied the illumination conditions at the lunar poles and attempted to identify areas of permanent sunlight on the Moon. Sutoh et al. [15] investigated the influence of varying sunlight conditions on path planning by conducting simulations under diverse illumination scenarios. Their findings indicate that sunlight conditions during morning and evening periods, as well as in high-latitude regions, significantly affect the power efficiency of lunar rovers. Plonski et al. [46] proposed a data-driven method to create solar maps and then used these maps along with an energy consumption empirical model to plan energy-minimal paths. Kaplan et al. [47] first calculated the solar radiation density at different locations and then employed an improved particle swarm optimization algorithm to find the shortest time path under energy constraints. To achieve risk-aware exploration of the lunar south pole, which has dynamic illumination and permanently shadowed regions (PSRs), Lamarre et al. [48] combined conventional mission-level path planning with stochastic reachability to establish a joint chance-constrained mission-level online path-planning algorithm and validated their approach by simulating a traverse through the lunar PSRs using terrain and solar illumination orbital maps of the Cabeus Crater region.

References [49,50] studied the impact of communication conditions on the path planning of planetary rovers, in which [49] modeled a communication for lunar exploration scenarios, utilizing terrain information to predict the data rate of lunar rovers. In light of the complex signal propagation mechanisms in the lunar surface environment, as well as the constraints of terrain elevation and rover movement, Ref. [50] proposed a hybrid communication navigation system that effectively guides the rover to its target location while maintaining a reliable communication link. Cunningham et al. [38] simultaneously considered energy constraints, illumination, and communication availability, proposing an improved energy-aware spatiotemporal path-planning algorithm that significantly reduces the time required for path planning.

The aforementioned articles treated time-varying illumination or communication conditions as constraints, without considering the impact of time-varying environmental delays on the robustness of the algorithms. Inoue et al. [39] proposed a delay-robust spatiotemporal path-planning algorithm (ROBUST-STP3R) for time-varying environments. This method constructs a spatiotemporal map that combines time-invariant graphs (such as obstacles and steep slopes) with time-varying graphs (such as illumination and communication) and defines a new cost function based on distance and area type. To enhance robustness against scheduling delays, the algorithm models time using a weighted time-varying cost function. Finally, the classic A* algorithm is used to solve the path on the spatiotemporal map.

2.2. Algorithms Based on Self-State Constraints

The constraints on the state of the planetary rover include structural design factors [15,40,51] and resource constraints such as internal temperature and battery power [32]. The rover's motion execution mechanisms can be designed in various ways, such as wheels, tracks, or legs, and different execution mechanisms lead to different interactions between the rover and the terrain. Zhang et al. [52] summarized kinematic and dynamic models with different configurations. In addition, some research has focused on the stability of robots when traversing rugged terrain. For example, Brunner et al. [40] compared various robot–terrain interaction methods and proposed a new iterative contact point estimation method, which was then applied to a spatial global planner for rough terrain to assess the robot's posture and stability. Research efforts [53–55] have investigated various planetary rover designs, including wheeled, hinged, and "push-pull" types, to enhance the rovers' ability to navigate through various complex terrains. Ohki et al. [51] proposed a path-planning method that considers wheel–terrain slip for unknown terrains, ultimately yielding a path that minimizes slip estimation online. Ishigami et al. [56] proposed a path-planning and evaluation strategy that takes into account the dynamic mobility of rovers for (semi) autonomous navigation of planetary rovers in rugged terrain. This strategy designs a dynamic mobility index that simultaneously integrates rover stability, slip, runtime, and energy consumption to select the optimal path, effectively improving the navigation performance and safety of the rover.

In addition to the path-planning algorithms that consider external environments or internal states separately, researchers have also studied methods that take into account both external environmental constraints and internal state constraints. For example, Tanaka et al. [32] proposed a novel global path- and resource-planning method for lunar rovers by comprehensively considering internal resource states such as heat and power of the rover, as well as external environmental factors. The authors addressed the path-planning problem under resource constraints using reinforcement learning on a non-hierarchical grid map, avoiding reliance on hierarchical structures. Sutoh et al. [15] proposed a comprehensive path-planning method suitable for the lunar surface by simultaneously considering the motion mechanisms of the lunar rover and sunlight conditions. The authors investigated the impact of different mobility mechanisms on path planning by modeling and simulating the movement behaviors of lunar rovers equipped with tracked and wheeled systems. They also evaluated the effectiveness of path planning across various lunar latitudes and times, considering the influence of insolation conditions on rover performance.

3. Rule-Based Path-Planning Algorithms

Rule-based path-planning algorithms guide a rover to move towards a target position without collisions under certain constraints by defining a set of clear rules or strategies. Such algorithms can be applied to both global path planning and local path planning. Rule-based path-planning algorithms typically include graph search methods, potential field methods, sampling-based methods, and dynamic window approaches. Since the rules defining these algorithms are set by humans, these algorithms are easy to understand and implement. However, the rules of such algorithms are often local or heuristic, and compared to intelligent algorithms, there is still room for improvement in scalability and adaptability to dynamic environments. The representative methods included in each category of rule-based algorithms and their development timeline are shown in Figure 2.

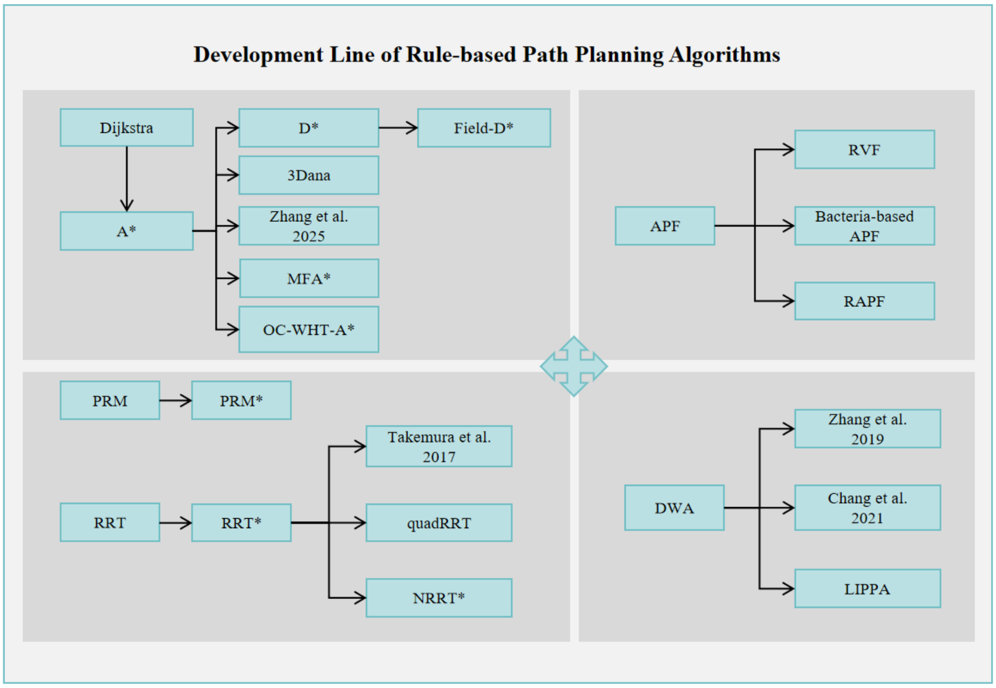

Figure 2. Development line of rule-based path-planning algorithms. For example, Dijkstra [57], A* [22], D* [58], 3Dana [59], Zhang et al. 2025 [60], MFA* [16], OC-WHT-A* [61], Field-D* [34], APF [27], RVF [62], Bacteria-based APF [63,64], RAPF [28], PRM [65], PRM* [66], RRT [67], RRT* [66], Takemura et al. 2017 [17], quadRRT [68], NRRT* [69], Zhang et al. 2019 [70], Chang et al. 2021 [71], LIPPA [72].

3.1. Graph Search-Based Algorithms

Graph search-based path-planning algorithms are a type of algorithm that explores the optimal discrete path by representing the environment as a graph structure. These algorithms divide the spatial environment into multiple discrete nodes, with each node connected to its neighboring nodes. When a rover moves from one node to a neighboring node, there is a certain cost associated with that movement. The goal of the algorithm is to find the optimal path from the starting node to the target node based on specific rules.

Common graph search path-planning algorithms include Dijkstra [57], A* [22], and D* [58], as well as improvements based on these algorithms. The Dijkstra algorithm [57] is the most basic and classic algorithm for solving the single-source shortest path problem. First, the algorithm sets the cost from the starting point to each node in the graph as infinite. Then, it gradually selects the node with the minimum cost to the starting point from the currently unvisited nodes and records the parent node of each visited node, until the entire graph is traversed and the minimum cost from the starting point to the target node is found. Since the Dijkstra algorithm requires traversing the entire graph, its execution speed is very slow when performing precise path searches in large-scale maps. A* [22] is a heuristic pathfinding algorithm based on improvements to Dijkstra's algorithm and greedy strategies. It ensures optimal paths while enhancing computational efficiency, making it suitable for the global path planning of rovers. The global path for the planetary rover using the A* algorithm is shown in Figure 3. D*, also known as Dynamic A*, is an improvement of the A* algorithm. It starts from the target point and gradually searches backward, dynamically updating the path based on changes in obstacles. This makes it suitable for local path planning in dynamically changing environments.

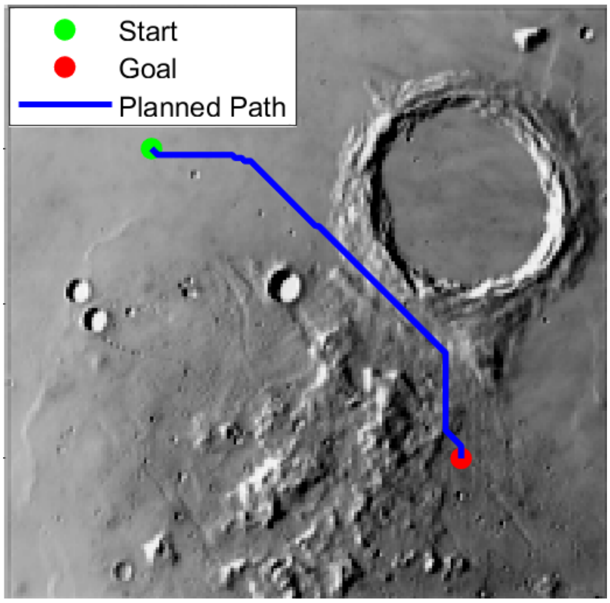

Figure 3. The global path for the planetary rover using the A* algorithm.

Another algorithm widely used in rovers is Field-D*. The GESTALT algorithm and Field D* algorithm were first used on the Spirit and Opportunity Mars rovers to help them navigate through more complex terrain [34]. Reference [59] proposed an algorithm called 3Dana, which utilizes elevation maps for environmental modeling while considering the safety of the generated paths. In graph search-based algorithms, the definition of the heuristic function is key to path planning. Even when using the same algorithm, different evaluation metrics can lead to the generation of different paths. Reference [73] designed a hierarchical path-planning method and developed a smooth curve path search algorithm that comprehensively considers various factors such as terrain, travel distance, and operational costs in navigation unit planning. To ensure the safety and efficiency of planetary exploration rovers, Zhang et al. [60] proposed an improved A* algorithm that takes the complex 3D terrain features, the motion constraints of the rover, and traversability into consideration and reduces planning time by 30.05% and generates smoother paths than the classic A* algorithm. Yu et al. proposed a multi-cost fast A* algorithm, called MFA* [16], where the heuristic cost function takes into account distance, terrain, and lighting conditions. The algorithm can set different weights for different scenarios, thereby generating various paths, while also improving the path search mechanism and reducing the execution time of the algorithm. Finally, the effectiveness of the proposed algorithm was validated in simulation experiments in the area near the landing site of the CE-3 mission. Considering the low computational efficiency associated with long-distance path planning for lunar rovers, Hong et al. [61] introduced a tile pyramid-based distributed path-planning strategy, integrating it with an enhanced A* algorithm to significantly accelerate computational speed. Subsequently, Zhou et al. [74] proposed a method that emphasizes both safety and efficiency in path planning. Their approach employs a distributed computation strategy, which accounts for various environmental factors including terrain slope, roughness, illumination, and rock abundance. They validated the superiority of their proposed algorithm for long-distance tasks through simulation experiments, demonstrating marked improvements in computational efficiency compared to traditional single-machine tasks. Masahiro et al. [75] introduced a machine learning-based terrain classification technique that used the real data from the Mars Science Laboratory (MSL) rover Curiosity to identify potential terrain hazards from images and then used a path search algorithm based on the rapid exploration random graph (RRG) and A* to avoid risks.

The literature review indicates that modern graph search-based path-planning algorithms focus primarily on creating heuristic cost functions and improving computational efficiency. Additionally, ensuring the safety of the resulting paths is a crucial factor that must be considered.

3.2. Potential Field-Based Algorithms

The artificial potential fields method, based on potential field theory, is a common path-planning method in the field of robotics, first proposed in [27]. Its basic idea is to treat the target point as a source of attractive force and the obstacles as sources of repulsive force. The robot moves towards the target under the combined influence of the repulsive and attractive forces that make up the "potential field", as shown in Figure 4. Many APF-based algorithms for mobile robotics and rovers have been proposed in recent years. For example, a potential field algorithm suitable for mobile robots to perform path planning in dynamic environments was proposed in [76]. The proposed approach used a fuzzy logic expert system to provide the mobile robot with the most appropriate heading toward the target. To address the issue of safe path planning for a six-wheeled rover on rough terrain, Raja et al. [77] introduced a gradient force function into the traditional potential field method. In the simulation experiments, the effectiveness of the proposed method was verified by simulating various terrains and obstacle layouts.

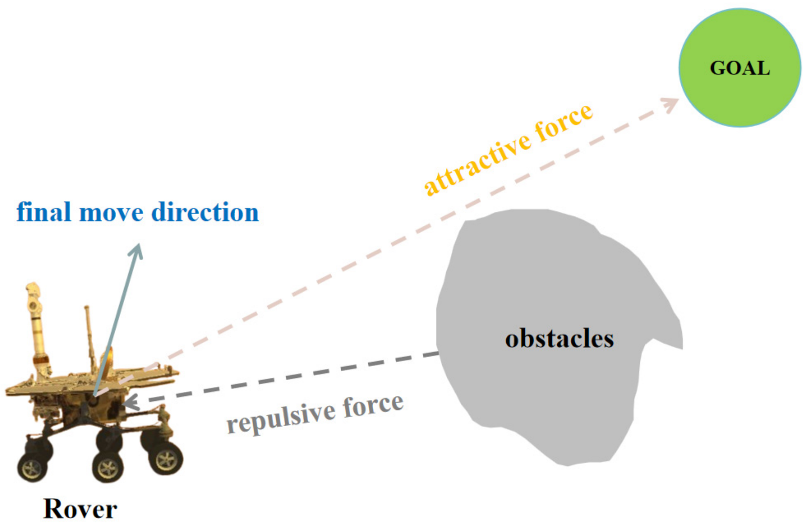

Figure 4. Graphical representations of artificial potential fields used in rovers.

Path-planning algorithms based on potential fields have advantages such as good real-time performance, high computational efficiency, straightforward mathematical analysis, and ease of handling obstacles in dynamic environments. However, this method faces the problem of local minima, where the robot may become stuck in a local position and unable to move. Local minima are often caused by various factors, including dense obstacle environments and the symmetry between different obstacles. Therefore, it is crucial to avoid trapping the rover in local minima when using the artificial potential field method for path planning.

Researchers have proposed various methods to prevent rovers from getting trapped in local minima. One such method, introduced by Nasuha et al. [62], utilizes a rotation vector field (RVF) to create a vortex field around obstacles, allowing mobile robots to escape when a local minimum occurs. This method enables the rover to navigate around obstacles by rotating the field direction, facilitating its "escape." However, when faced with complex terrain, it can cause frequent oscillations in the vector field, making it challenging to create a stable path. Moreover, references [63,64] both employed a bacteria-based APF algorithm, implemented within uncharted and congested environments, such as the navigation of rovers across lunar terrain. Among them, the algorithm in [64] can escape local minima by actively modifying the potential functions and using random walk techniques (RWTs). The RWT method is straightforward to implement, using probabilistic perturbations to escape local minima. However, it lacks directional guidance, and its success rate in escaping local minima is contingent upon the settings of random parameters. Manteaux et al. [28] proposed a robust artificial potential field (RAPF) algorithm for local reliable path planning for rovers. When encountering a local minimum, the system marks it as an artificial obstacle, which affects the total potential function and recalculates the path from the current position. Another improvement in the algorithm is the introduction of the relative position between the rover and the target during the generation of bacteria points. To test the performance and effectiveness of the algorithm in a more realistic environment, a 3×4 m sandy terrain that simulates the lunar surface, including rocks, craters, and slopes, was constructed to verify the effectiveness of the proposed method. The simulation results demonstrate that the proposed algorithm effectively addresses the local minima problem, reducing computation time by 50% compared to the classic APF algorithm. Additionally, it shows potential for application in complex terrains.

3.3. Sampling-Based Algorithms

Sampling-based path-planning algorithms generate feasible paths from a starting point to a target point by random or deterministic sampling in space. Specifically, algorithms of this type include probabilistic roadmaps (PRMs) [65] and Rapidly exploring Random Trees (RRTs) [67]. These algorithms are suitable for high-dimensional complex environments and can effectively handle situations with obstacles and dynamic changes in the environment. However, to obtain or approach a globally optimal solution, the number of samples that the algorithm needs to sample is very large, leading to a slow convergence rate. Additionally, these algorithms are quite sensitive to the initial solution.

The basic idea of the PRM algorithm [65] is to randomly sample in the workspace, construct a graph composed of nodes and edges, and find a feasible path from the starting point to the target point within the constructed graph. The PRM algorithm also has issues such as a large number of samples, low computational efficiency, and difficulty in obtaining optimal solutions. An improved version of the PRM algorithm, called PRM*, was proposed in [66], which is capable of obtaining a globally optimal path.

As shown in Figure 5, the basic idea of the RRT algorithm [67] is to continuously generate a tree from the starting point to the target point through random sampling, gradually constructing a path using the nodes in the tree. However, the paths generated by the RRT algorithm are often winding and may not be optimal. The RRT* algorithm [66] improves upon the RRT algorithm by gradually optimizing the path quality as the number of iterations increases, ultimately yielding the optimal path. To adapt to rough terrain, Reiya et al. [17] proposed a traversability-based RRT$*$ algorithm based on the RRT* algorithm. The authors first constructed an environmental map using point cloud data captured by a light detection and ranging (LiDAR) sensor and then applied the RRT* algorithm to sample the point cloud data, taking into account the roughness of the terrain during the tree expansion process. During the simulation phase, the author used a terrain map captured by LIDAR in a volcanic area at Mt. Mihara in Japan as the original experimental data to validate the applicability of path planning for the rover on different types of real-world rough terrain. Since this algorithm requires complex three-dimensional modeling of the environment for local path planning, it suffers from low computational efficiency, which limits the speed of the lunar rover. To address the computational efficiency issue of the RRT* algorithm, Paniagua et al. proposed a quadRRT algorithm [68]. This algorithm can leverage Nvidia's graphics processing unit to accelerate RRT computations in dynamic large-scale maps.

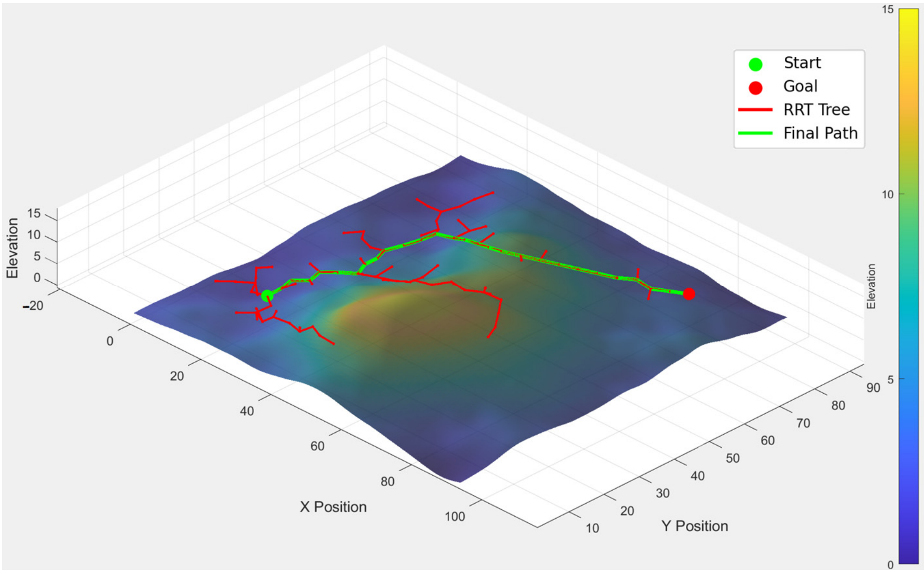

Figure 5. The process of path planning for the planetary rover using the RRT algorithm.

Currently, researchers are not only optimizing single sampling-based algorithms but also combining these algorithms with other types to achieve complementary advantages. For example, Zhang et al. [26] proposed a local path-planning method for lunar rovers that integrates RRT* and DWA for autonomous path planning and dynamic obstacle avoidance in dynamic environments. To address the issues of initial solution sensitivity and slow convergence speed associated with RRT and its variants, a new optimal path-planning algorithm called NRRT* (neural RRT*), which combines convolutional neural networks with RRT, was proposed [69], and the effectiveness of the algorithm was validated through simulation experiments.

3.4. Dynamic Window Approach

The dynamic window approach (DWA) is primarily used for local path planning of planetary rovers while avoiding collisions with obstacles. It samples the possible velocity space of the rover to simulate its motion trajectory, followed by trajectory evaluation and path selection to obtain a relatively optimal path [78]. The DWA algorithm makes decisions based on dynamic information at the current moment, providing high real-time performance and adaptability for path planning in dynamic environments. However, this algorithm may cause the planetary rover to get stuck in a local optimum and does not guarantee a globally optimal path.

To address this issue, Zhang et al. [70] proposed an improved DWA algorithm that uses the results of global path planning as a reference and then designs a new evaluation function to ensure a globally optimal trajectory. In addition, Lu et al. proposed a dynamic window path-planning algorithm based on Q-learning [71]. This algorithm expands the three evaluation functions in the original DWA to five to solve the local optimum problem and then employs a Q-learning-based adaptive tuning method to optimize the DWA parameters. To assist the lunar rover in carrying out complex tasks under weak communication conditions at the lunar south pole and to achieve high-precision autonomous navigation, Wang et al. proposed a fusion path-planning algorithm called LIPPA [72], which combines the A* algorithm and the DWA algorithm. This algorithm integrates the global auxiliary path constructed by the A* algorithm with the DWA algorithm and then introduces significant landmarks at the lunar south pole to construct a new evaluation function. Finally, the effectiveness of the algorithm was validated through an indoor semi-physical simulation experimental scene.

4. Biologically Inspired Path-Planning Algorithms

Biologically inspired path-planning algorithms are inspired by the behaviors and adaptive mechanisms of organisms in nature. By mimicking their decision-making processes, these algorithms provide solutions for efficient path planning for robots or agents in complex environments. Such algorithms mainly include evolutionary algorithms, fuzzy computation, and machine learning algorithms [79]. The representative algorithms included in each category and their development timeline are shown in Figure 6. Biologically inspired path-planning algorithms typically exhibit strong adaptability, allowing them to operate effectively in dynamic and uncertain environments. Therefore, these algorithms can be applied to reliable path planning for rovers on the planetary surface, which is filled with unknowns and obstacles.

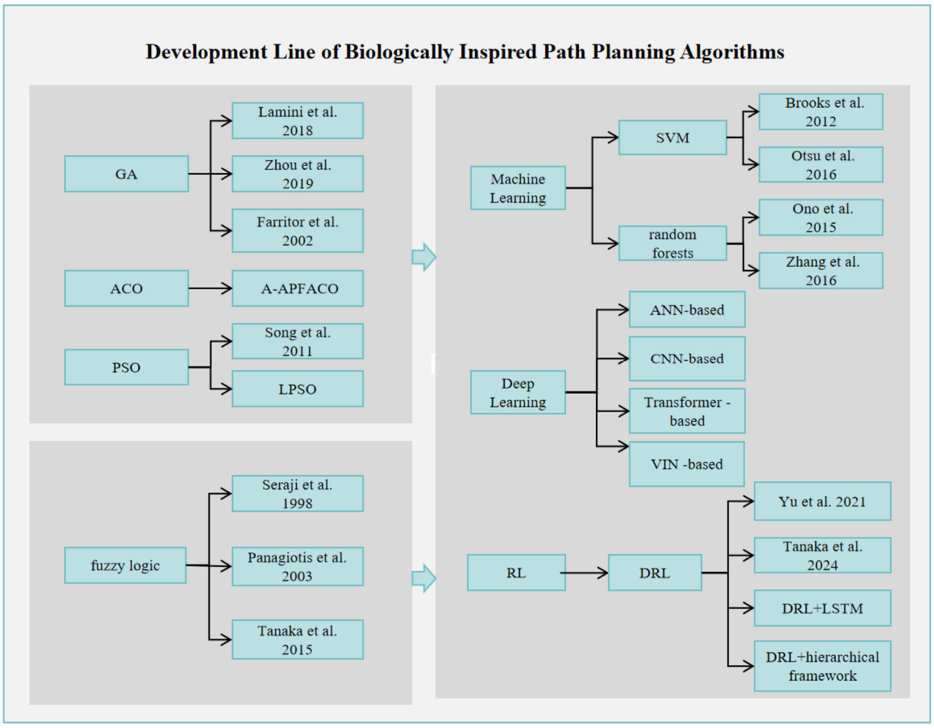

Figure 6. Development line of biologically inspired path-planning algorithms. For example, GA [80], Lamini et al. 2018 [25], Zhou et al. 2019 [81], Farritor et al. 2002 [41], ACO [82], A-APFACO [83], PSO [84], Song et al. 2011 [85], LPSO [86], Seraji et al. 1998 [87], Panagiotis et al. 2003 [88], Tanaka et al. 2015 [89], Brooks et al. 2012 [90], Otsu et al. 2016 [91], Ono et al. 2015 [75], Zhang et al. 2016 [92], ANN-based [29], CNN-based [30,93,94], Transformer-based [93], VIN-based [95,96], Yu et al. 2021 [14], Tanaka et al. 2024 [32], DRL+LSTM [97], DRL+hierarchical framework [98].

4.1. Algorithms Based on Evolutionary Learning

Algorithms based on evolutionary learning mainly include genetic algorithms [80] and swarm optimization algorithms. The genetic algorithm continuously optimizes paths through operations such as "selection, crossover, and mutation", simulating the natural selection and genetic mechanisms of biological organisms, ultimately arriving at an approximately optimal path [99]. In a genetic algorithm, each chromosome can represent a path, and the quality of each path is evaluated using a fitness function. The process then undergoes continuous inheritance and iteration until a termination condition is met. To solve the path-planning problem in a static environment using genetic algorithms, Lamini et al. [25] proposed an improved crossover operator. The proposed crossover operator provides feasible paths with better fitness values, accelerating the convergence speed of the algorithm. For planetary exploration tasks, Farritor et al. [41] developed a genetic algorithm-based autonomous robot path-planning algorithm, which is suitable for areas that are difficult to reach (such as canyons, craters, dry riverbeds, and steep cliffs). This algorithm considers various constraints such as power, actuator saturation, wheel slip, and vehicle stability, and it is validated against an analytical model of the robot and its environment. Zhou et al. [81] proposed a comprehensive genetic algorithm for lunar rover path planning, which improves adaptability in dynamic environments by incorporating a terrain composite cost into the fitness function.

Swarm optimization algorithms solve optimal path problems by simulating the collaborative behaviors of biological swarms in nature, mainly including ant colony optimization (ACO) [82] and particle swarm optimization (PSO) [84]. For three-dimensional grid terrain scenarios, Zhou et al. [100] proposed an improved ant colony algorithm based on slope and slope direction for lunar rover path planning and validated the effectiveness of the proposed algorithm in solving slip prediction path-planning problems using experimental simulation data. In addition, Zhu et al. [83] combined the ant colony algorithm with the artificial potential field method, introducing an induced heuristic factor to dynamically adjust the state transition rules of the ant colony algorithm, which improved the convergence speed of the algorithm and designed a dynamic obstacle avoidance strategy within the algorithm. There has also been extensive research on the application of particle swarm optimization algorithms in planetary rover path planning. For example, Song et al. [85] improved the particle swarm optimization algorithm and applied it to global navigation point planning for lunar rovers to obtain the global path, and simulations were conducted on several different lunar topographic maps to verify planning effectiveness. Katiyar et al. [101] proposed a CG-Space-based real-time dynamic path-planning method. By introducing an improved penalty function into the PSO objective function, they were able to handle obstacles of varying sizes and shapes that move randomly in real time, planning an optimal collision-free path with dynamic obstacles in CG space. To address the issue of slow convergence speed in PSO, Lu et al. [86] proposed a particle swarm optimization algorithm based on generative learning (LPSO). This algorithm first uses a generator to obtain a foreground area with feasible paths and then employs the particle swarm algorithm to conduct a rapid search within that area. It is not only applicable to grid maps but also performs well in real-world environmental maps.

4.2. Algorithms Based on Fuzzy Computation

The fuzzy logic-based path-planning algorithm uses fuzzy control theory to make decisions about the movement of planetary rovers. It generates control commands based on input information such as distance, terrain, and speed, using fuzzy rule reasoning. Such algorithms have the advantages of being simple and easy to implement, but the design of the fuzzy rule base is quite complex and requires human experience and a lot of experimentation to determine. In addition, these algorithms lack dynamic adaptability when facing complex and uncertain planetary surface environments. In 1997, after NASA implemented the Mars Pathfinder mission to land on Mars and deployed the Sojourner rover on the Martian terrain, Seraji et al. [87] proposed an autonomous navigation strategy suitable for Martian rovers based on a fuzzy logic framework and terrain traversability metrics. The navigation strategy consists of three independent behaviors, with their weight factors generated using fuzzy rules, and does not require prior environmental information, allowing the rover to autonomously choose paths that are easy to traverse. Considering scenarios with strong real-time requirements, Panagiotis et al. proposed a fuzzy logic-based system for intelligent motion planning and navigation of mobile robots in dynamic environments [88]. This system is very simple and has a short response time. In addition, references [89,102–104] all use a fuzzy logic framework to describe the environment. Among them, references [89,102] and [103] describe terrain traversability and terrain cost using fuzzy logic algorithms, respectively. The work in [104] is used for the path planning of robots in harsh environments and validates the effectiveness of the proposed algorithm on rugged Martian terrain. More fuzzy logic-based path-planning methods can be found in [105].

4.3. Machine Learning-Based Algorithms

In the extraterrestrial exploration missions that humanity has conducted so far, many rovers can perceive their surrounding environment based on stereo images, detecting rocks, steep slopes, and other geological features. Figure 7 shows the stereo image from the rover "Sojourner" of the 1997 "Pathfinder" mission. At the same time, the Mars Exploration Rover (MER) and the Mars Science Laboratory (MSL) have been equipped with intelligent algorithm processing capabilities, allowing them to process full stereo images and construct traversability maps to select the best driving paths [106]. In addition, the Chang'e 4 mission's Yutu-2 rover is designed with an integrated intelligent architecture that allows it to be intelligently controlled from the ground while also possessing a certain degree of autonomous capability [107].

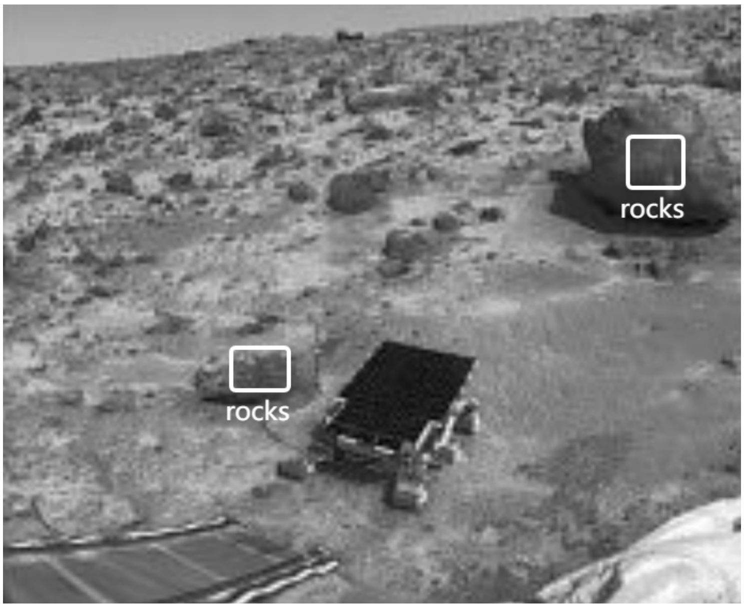

Figure 7. Stereo images from the 1997 Pathfinder mission featuring the Sojourner rover https://www.stereoscopy.com/mars/#pathfinder (accessed on 12 March 2025).

In recent years, machine learning and deep learning technologies have been widely applied across various industries, and the field of learning-based planetary rover path planning has also garnered significant attention from researchers, resulting in many important research outcomes. Conventional path-planning algorithms typically consider multiple constraints when calculating path costs and searching the entire environmental space, leading to an exponential increase in computational complexity as the environmental space expands. In contrast, machine learning-based path-planning algorithms can effectively adapt to the complex and unknown planetary surface environment and respond quickly to unforeseen situations [14,108]. Such algorithms mainly include traditional machine learning algorithms, deep learning-based algorithms, and reinforcement learning-based algorithms, among others.

4.3.1. Traditional Machine Learning Algorithms

Traditional machine learning algorithms are mainly used to assist rovers in path planning, improving planning efficiency. For example, during the autonomous inspection and detection missions performed by the rover, to avoid potential risks that could lead to task failures, supervised learning methods such as support vector machines (SVMs) [90,91] and random forests [75,92] are used to classify and assess the terrain in the planetary surface map. The results of the assessment can be utilized to plan safe paths. Brooks et al. [90] propose a self-supervised learning framework for terrain classification, where a proprioceptive terrain classifier distinguishes terrain types based on features generated from the interaction between the rover and the terrain. Subsequently, the labels produced by this classifier are used to train a vision-based terrain classifier for traversability assessment. Otsu et al. [91] simultaneously employ both vibration-based and vision-based classifiers, using the SVM method for terrain classification. This approach is suitable for model training on sparse datasets and has been validated on Mars-like terrain. Ono et al. [75] trained a random forest classifier based on manually labeled navigation camera (NAVCAM) image data to predict the class label of each pixel in the image, thereby distinguishing various terrain types within the image. This was later used as an input to provide references for the safe path planning of the rover, reducing the risks associated with different terrains. The algorithm execution process is shown in Figure 8. A terrain classification method based on random forests was designed in [92], which can efficiently extract a subset of features related to the terrain and achieve high classification accuracy and speed, making it suitable for real-time terrain classification and traversability assessment.

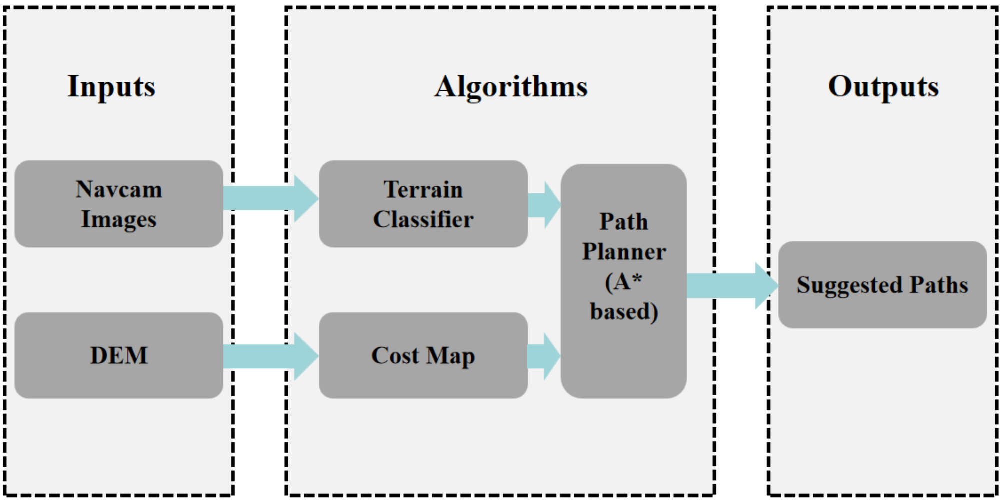

Figure 8. The algorithm [75] execution process of using a random forest classifier to distinguish various terrains.

4.3.2. Deep Learning-Based Algorithms

Algorithms based on deep learning have powerful feature extraction capabilities for complex sensor data, which can be used for environmental modeling beneath the complex planetary surface, enabling early awareness of resources and risks. At the same time, they can conduct vision-based traversability assessments to enhance the autonomous exploration capabilities of planetary rovers. Masahiro et al. introduced machine learning-based analytics for automated rover systems (MAARS) and focused on the implementation of science-driven (DBS) and energy-optimal autonomous driving (EOA) capabilities [109]. These capabilities are built upon research in deep learning, optimal planning, and ground mechanics. The article also discusses topics such as information-theoretic path planning, resource-aware path planning, and onboard strategic path planning.

The characteristics of autonomous planetary rovers are that they can complete required tasks without continuous human guidance, adapt to changing environments, survive in emergencies or failures, and traverse unstructured terrain without human assistance. In recent years, researchers have deeply integrated deep learning algorithms into the specific path planning of autonomous planetary rovers. For example, a simple planetary rover path-planning method based on artificial neural networks (ANNs) was proposed in [29], which mainly consists of three consecutive layers, in which the input layer takes the rover's sensor data as input, and the output layer directly controls the movement of the lunar rover. Zhang et al. [30] proposed a deep learning-based global path-planning algorithm called DB-CNN for rovers, which can perform path planning directly from orbital images of planetary surfaces without relying on environmental mapping. The efficiency and accuracy of path planning are improved through a double-branch structure and non-iteration design, and the effectiveness of the proposed method is validated through simulation experiments. Fan et al. [93] proposed a new dual-branch semantic segmentation network (TerSeg) that combines the strengths of both CNN and vision transformer architectures, and the proposed TerSeg network can achieve high-precision recognition of terrain in deep space environments, enabling autonomous path planning for rovers. Rothrock et al. [94] utilized deep convolutional neural networks to classify and recognize Martian terrain to assess the traversability of rovers. This model was successfully applied to the analysis of the landing site's traversability for the "Perseverance" and the sliding prediction for the MSL mission. Higa et al. [31] proposed a vision-based rover energy prediction algorithm for path selection during the local path-planning process. The algorithm takes RGB images and depth images as input to predict the rover's energy consumption. In addition, the Value Iteration Network (VIN) was used in [95,96] for end-to-end path planning of Mars rovers. The idea of VIN is to embed the value iteration process into a neural network, where [96] proposed a Soft Value Iteration Network (SVIN) based on VIN to optimize the accuracy of path planning.

In addition, deep learning has also been applied to the path-planning research of planetary rovers in certain specific scenarios. For achieving long-distance path planning for lunar rovers, Jia et al. [110] proposed a robust and reliable model framework utilizing a multi-level map model. This framework not only integrates data from different layers, such as slope, relief, roughness, and rock abundance, but also employs a transformer-based model to extract small-scale obstacles. Subsequently, it constructs a multi-level cost map for long-distance path planning. To address the path-planning challenges under the dynamic illumination conditions of the lunar poles, Chen et al. [111] proposed a solar-synchronous spatiotemporal U-Net network to simplify data processing and identify areas with favorable illumination conditions. Afterwards, an improved A* algorithm, 3ST-A*, leveraging preprocessed data, was used for global path planning. The approach proposed in [112] is to handle uncertainty where quantification, utilization, and adaptation are integrated into a single learning and planning framework for rover navigation. The article proposes an end-to-end probabilistic machine learning model using DNNs for traversability prediction, which can help the rover generate more robust paths.

Recent studies have demonstrated the potential of large language models (LLMs) to enhance efficiency in robot path-planning tasks [113–116]. For instance, Meng et al. [113] proposed the LLM-A* algorithm, which integrates the precise pathfinding capabilities of the A* algorithm with the global reasoning abilities of LLMs. Additionally, Xiao et al. [114] introduced LLM-Advisor, a benchmark for cost-efficient path planning across multiple terrains, leveraging LLMs as effective advisors. These research perspectives pave the way for future advancements in planetary rover path planning.

4.3.3. Reinforcement Learning-Based Algorithms

Another representative algorithm is the path-planning algorithm based on reinforcement learning (RL). The RL algorithm adopts a general framework for adaptive decision making, allowing the planetary rover to continuously interact with the environment and gradually learn the optimal path through trial and error. This algorithm demonstrates good generalization capabilities in complex environments. Deep reinforcement learning (DRL) integrates deep neural networks into the RL algorithm, enabling it to successfully handle complex high-dimensional data. Moreover, DRL models that are successfully trained and deployed can directly generate control commands for the planetary rover based on environmental information, eliminating the need for environmental reconstruction and path-replanning steps that are required in traditional autonomous path-planning algorithms. Therefore, reinforcement learning algorithms are particularly suitable for dynamic tasks such as autonomous exploration of the planetary surface in complex environments.

Yu et al. [14] proposed a learning-based end-to-end path-planning algorithm that considers safety constraints, as shown in Figure 9. In this method, the authors first established a realistic lunar surface environment and lunar rover system using Gazebo and then employed the DRL algorithm to train the model for achieving efficient autonomous exploration of the lunar rover. Specifically, they designed the state space and action space for the agent, while using a deep neural network as the policy network, taking depth images, radar point cloud data, and lunar rover state information as inputs and utilizing CNN to extract information from the environment. The article also designed a safety reward function that considers the slipping behavior of the lunar rover to enhance its adaptability in different terrains. Tanaka et al. [32] fully considered various constraints such as lunar terrain, lighting, thermal, and power conditions of the lunar rover, proposing a global path-planning method for the lunar rover based on reinforcement learning, which addresses the shortest path problem under resource-constrained conditions (RCSP). Park et al. [33] combined reinforcement learning with a kinematic model of a lunar rover to address the issue of limited movement when the steering motor of a four-wheeled lunar rover fails. They proposed a fault-tolerant algorithm that ensures the execution of tasks even in the event of a motor failure. Reference [97] integrated the DRL algorithm with long short-term memory networks (LSTM) to achieve obstacle avoidance. Hu et al. [98] proposed a global path-planning method based on a hierarchical framework and DRL algorithm, aimed at improving the path-planning efficiency of long-distance planetary rovers. This method first constructs a binary feature map and then designs a hierarchical planning framework that combines step-by-step planning and block iteration, significantly enhancing computational speed and adaptability. Additionally, a dual-branch residual network, SP-ResNet, is used for action value estimation at each step of the planning execution, and the effectiveness of the proposed method was ultimately validated on real lunar terrain.

In addition, researchers have combined RL algorithms with traditional path-planning algorithms. For example, Daftry et al. [117] proposed a learning-enhanced path-planning framework to improve the navigation efficiency of Mars rovers in complex terrains while ensuring safety. They integrate the strong environmental perception capabilities of traditional machine learning methods with the predictability and safety of classical search methods, guiding the Mars rover's path planning in complex environments. Experimental results show that the proposed method, MLNav, performs excellently in both real Martian terrain and synthetic terrain, particularly in reducing the number of collisions and improving the feasibility of paths. Lu et al. [118] explored a novel cooperative path-planning method for lunar rovers by combining multi-agent reinforcement learning with artificial potential field techniques, enhancing the navigation efficiency and safety of rovers in complex lunar surface environments. The proposed method effectively avoids large obstacles and reduces collisions with small obstacles, while also minimizing waiting times in path planning and improving cooperation efficiency. Compared to a multi-agent A* algorithm that uses improved obstacle avoidance methods, the proposed method can guide the lunar rover's movement more safely, quickly generate paths and action sequences, adapt well in dynamic environments, and achieve higher efficiency.

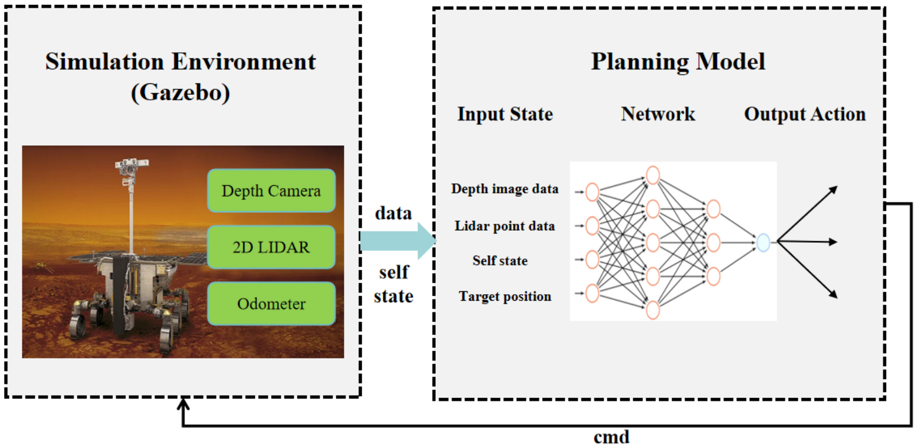

Figure 9. A learning-based end-to-end path-planning framework proposed in [14].

Machine learning-based path-planning algorithms for planetary rovers have the advantages of strong adaptive ability and are good at dealing with complex dynamic environments, which can realize intelligent and efficient path planning for planetary rovers, but their application to the real extraterrestrial surface environment still faces many challenges:

1. The training of machine learning models needs to rely on a large amount of real data. However, the current amount of data on the extraterrestrial surface is extremely limited, which may lead to insufficient model generalization ability. In addition, the uncertainty of planetary surface environmental data (e.g., terrain, obstacles, soil properties, etc.) also increases the difficulty of model training.
2. Machine learning models, due to the "black box" nature of the operation process and the lack of interpretability, may lead to unpredictable and unsafe behaviors of the planetary rover, which may be a greater risk, especially in critical missions.
3. There are challenges in the migration ability and robustness of machine learning-based methods. In future deep space exploration missions, planetary rovers need to plan paths in different terrains, environments, and scientific objectives, and machine learning-based models may have the problem of superior modeling in specific environments but poor performance in real environments due to their strong dependence on training data.
4. Machine learning algorithms have problems such as complex training processes and consumption of computational resources, especially when such algorithms are applied to autonomous path planning for planetary rovers, which is a considerable burden on the rover's computational platform.

Accordingly, future endeavors must prioritize the continued investigation and development of secure and efficient machine learning algorithms. This includes researching path-planning algorithms optimized for environments with constrained resources, addressing concerns related to reliability and robustness, and facilitating the practical implementation of these algorithms in autonomous path planning for planetary rovers.

5. Discussion and Future Works

Path planning for planetary rovers constitutes a critical and complex endeavor within the field of deep space exploration missions. This process is essential for enabling rovers to identify safe and efficient trajectories that adhere to specific constraints while fulfilling significant scientific exploration objectives. Currently, both rule-based path-planning algorithms and biologically inspired intelligent path-planning algorithms face various challenges and limitations.

As future deep space exploration missions progress, planetary rovers are expected to confront increasingly intricate and extreme environmental conditions. Consequently, several prospects and recommendations for future research trajectories in the domain of planetary rover path planning are proposed below.

Development of Advanced Path-Planning Algorithms. There is a pressing need to explore and develop safer and more efficient path-planning algorithms. This is particularly vital for long-distance global path-planning tasks, where challenges arise in designing a safe and effective heuristic function tailored to specific task scenarios. Moreover, the application of distributed computing strategies can significantly enhance computational efficiency in path planning.

Incorporating Comprehensive Constraints. Future research must consider more comprehensive constraints that are reflective of the characteristics inherent to exploration tasks. For instance, the CE-7 mission aims to conduct detailed exploration of lunar regolith, water ice, and volatile components at the lunar south pole. A key challenge for this mission involves optimizing the rover's path to maximize solar energy utilization. Thus, the application of time-varying illumination data on the lunar surface is critically significant for enhancing solar power utilization and facilitating the search for water ice at the lunar south pole.

Integration of Intelligent Algorithms. The future landscape of planetary rover path planning will likely evolve towards deep intelligence, emphasizing the development of an algorithmic system endowed with autonomous cognition and evolutionary capabilities. For example, a dynamic decision-making framework based on deep reinforcement learning (DRL) could transcend the limitations of traditional rule-based systems. By integrating visual, LiDAR, and navigation data to construct a target value network, it would autonomously generate routes that balance safety and scientific value amidst complex scenarios such as impact craters, rugged terrain, and permanently shadowed areas.

Multi-Algorithm Collaborative Evolution. In the future, the tasks associated with path planning for planetary rovers will evolve towards multi-algorithm collaborative evolution. This approach will leverage the strengths of diverse algorithms to address the varied demands encountered in deep space exploration missions. A hierarchical decision framework that integrates traditional algorithms with artificial intelligence models can utilize neural networks for feature extraction, subsequently employing traditional algorithms for effective path planning. To develop a path-planning algorithm characterized by high efficiency and robustness, it is imperative to analyze the constraints imposed by the environment in conjunction with the design of the mission itself. Here, we present a preliminary study aimed at demonstrating the efficacy of lunar rover path planning in support of the upcoming CE-7 polar exploration mission. The Moon's south pole represents a region of significant scientific interest due to the presence of cold-trapped water ice and volatiles, which are vital for understanding the evolution of lunar geology and facilitating in situ resource utilization for future lunar base construction. However, the complex terrain, limited Earth–Moon communication, and variable sunlight conditions pose substantial challenges for solar-powered rover explorations.

To quantitatively assess the efficiency of various path-planning algorithms, we have selected a potential landing site measuring 1300 × 1300 m, located near the Moon's south pole [47,119,120]. Initially, we generated a slope map for this region utilizing a digital elevation model (DEM) derived from the Lunar Orbiter Laser Altimeter aboard NASA's Lunar Reconnaissance Orbiter (http://imbrium.mit.edu/BROWSE/LOLA_GDR/POLAR/SOUTH_POLE/ (accessed on 8 March 2025)), featuring a resolution of 20 m/pixel. Subsequently, we simulated the illumination conditions for this area to determine the time-averaged illumination between 1 November 2026 and 31 December 2026, with a temporal resolution of one hour. Finally, we conducted simulation experiments employing A*, Rapidly exploring Random Tree (RRT*), APF, GA, and Deep Q-Network (DQN) algorithms. The results of these experiments, alongside comparative analyses, are presented below, providing insights into the optimal path-planning strategies for lunar exploration.

The simulation results indicate that the A* path-planning algorithm is capable of generating the optimal path within the simulation scenario, achieving an average execution time of 34 ms and a path length of 1474 m, comprising 59 nodes (refer to Figure 10 and Table 2). In contrast, the performance of the RRT* algorithm is contingent upon its parameter configurations. For example, with a sampling step size and neighborhood radius set to 200 m, the algorithm required 453 ms to produce a path measuring 1623 m and consisting of 11 nodes (see Figure 11a). When the sampling step size was reduced to 100 m, the execution time escalated to 567 ms, while the path length decreased to 1609 m, incorporating 19 nodes (refer to Figure 11b). Further reducing the step size to 60 m resulted in a significant increase in execution time to 1028 ms for a path length of 1593 m, which included 30 nodes (see Figure 11c). In comparison to the A* algorithm, the convergence time for the RRT* algorithm progressively increases as it approaches a globally optimal solution, highlighting its sensitivity to initial parameter settings.

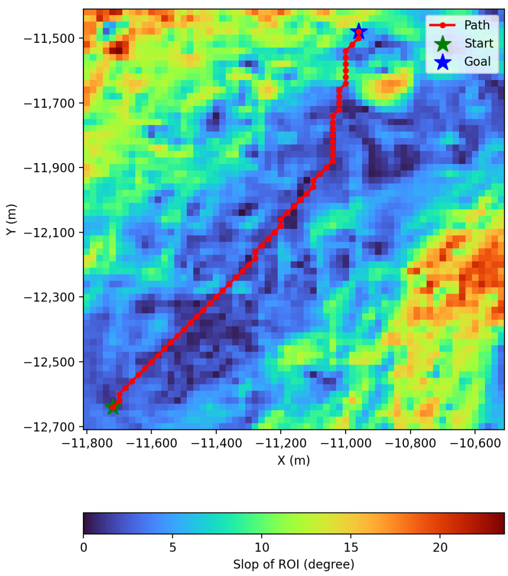

Figure 10. Optimized path planning using the A* algorithm considering the constraints of slope, distance, and illumination. Both the start and goal points are randomly provided as examples. The slope map serves as the background, which is created using a polar stereographic projection centered at 137.2216°W and 89.4586°S.

Table 2. The comparison results of A*, RRT*, APF, GA, and DQN.

Algorithms	Parameter Settings	Computation Time (ms)	Path Length (m)	Generated Node Number
A*	/	34	1474	59
RRT*	Initial parameter set to 200 m	453	1623	11
	Initial parameter set to 100 m	567	1609	19
	Initial parameter set to 60 m	1028	1593	30
APF	Step size set to 20 m	/	/	/
	Step size set to 30 m	184	1454	58
	Step size set to 60 m	88	1426	23
GA	Iterations set to 300	3150	1504	75
DQN	/	1026	1920	97

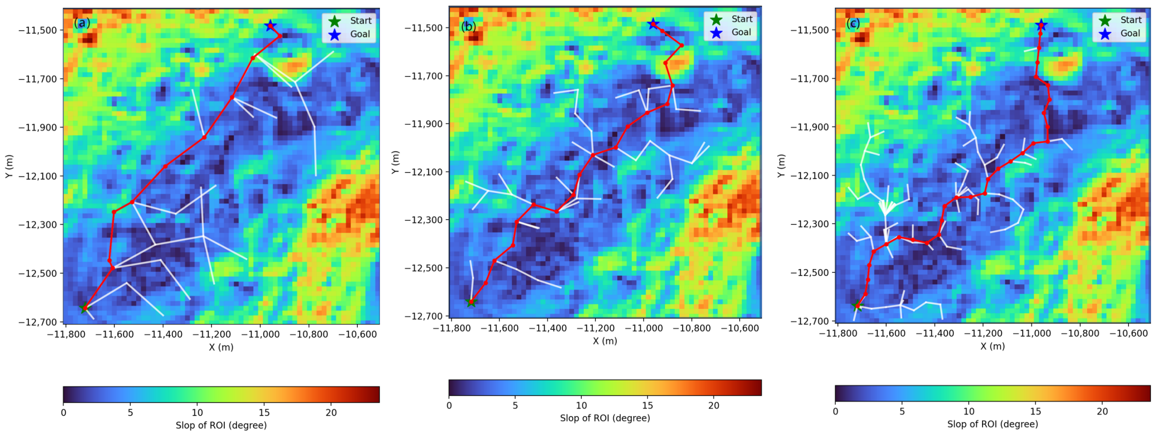

Figure 11. Path planning using the RRT* algorithm considering the constraints of slope, distance, and illumination. Both the start and goal points are randomly provided, which is the same as Figure 10. The red lines represent the final path, while the white lines illustrate the RRT* algorithm's expansion process. The slope map serves as the background. The sampling step sizes and neighborhood radii for (**a**), (**b**), and (**c**) are set at 200 m, 100 m, and 60 m, respectively.

The performance of the APF algorithm is significantly affected by its parameter configurations, which may result in the rover becoming ensnared in local minima (refer to Figure 12). For example, when the step size is set at 20 m, the rover fails to complete the final path planning. Conversely, it successfully executes the task when the step sizes are modified to 30 m and 60 m. Remarkably, the computational time is diminished, the path length is shortened, and the number of nodes is reduced to 60 m in comparison to 30 m. Additionally, the APF method exhibits superior real-time performance relative to RRT*, GA, and DQN.

In the genetic algorithm, the generated path is continuously optimized through operations such as selection, crossover, and mutation. The final path produced by the genetic algorithm is illustrated in Figure 13. As the number of iterations increases, the fitness scores of the paths also improve until convergence is achieved (refer to Figure 14). In comparison to alternative algorithms within our simulation scenario, this algorithm requires the most time to produce feasible paths due to its persistent iterative process (see Table 2).

Figure 15 illustrates the outcome of path planning utilizing the DQN algorithm, while Figure 16 depicts the training process associated with DQN. During the training phase, both the total reward and training loss exhibit a gradual convergence. Subsequently, the trained DQN model is employed for forward inference, facilitating the intelligent generation of paths. However, it is noteworthy that the training process of this algorithm necessitates substantial computational resources and time; for instance, in our experiments, even a simplistic three-layer fully connected network requires approximately two to three hours to complete a single training cycle.

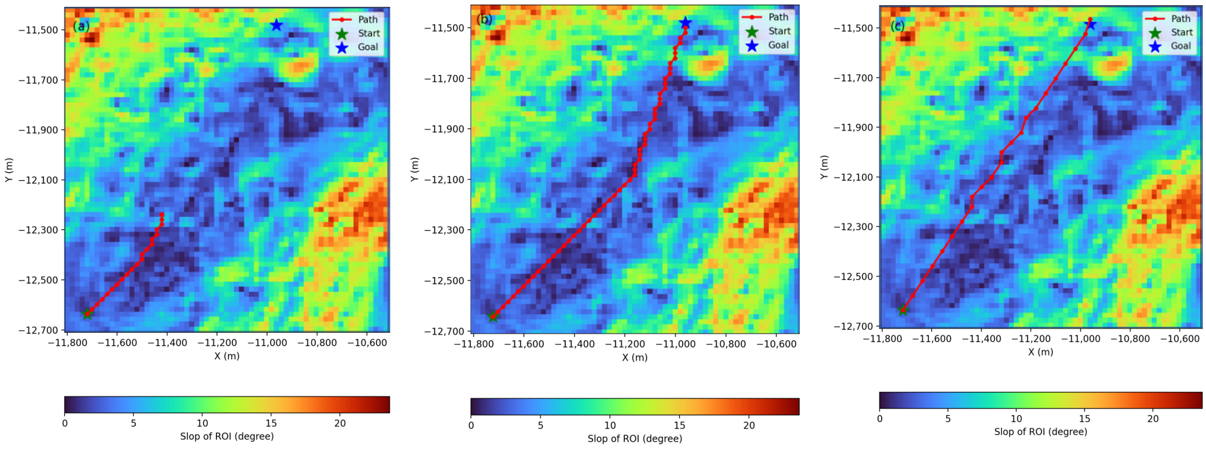

Figure 12. Path planning using the APF algorithm considering the constraints of slope, distance, and illumination. Both the start and goal points are randomly provided, which is the same as Figure 10. The slope map serves as the background. The step sizes for APF in (**a**), (**b**), and (**c**) are 20 m, 30 m, and 60 m, respectively. The path planner in (**a**) is trapped in a local minimum and fails to complete the path, while (**b**,**c**) successfully finish the task.

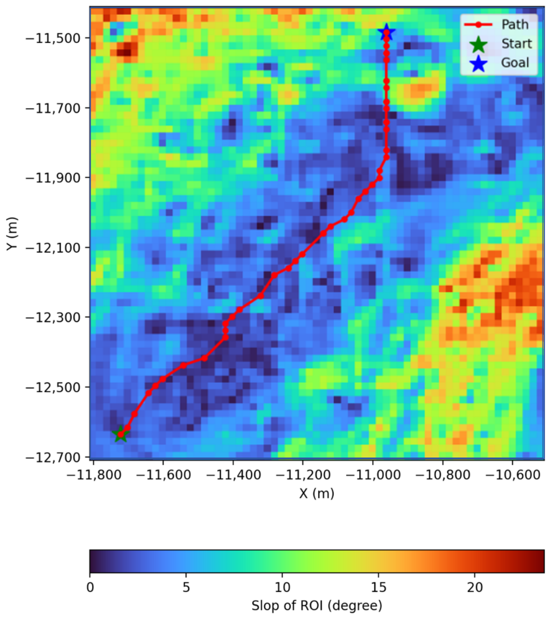

Figure 13. Path planning using the GA algorithm, where the fitness function considered slope, distance, and illumination. Both the start and goal points are randomly provided, which is the same as Figure 10. The slope map serves as the background.

The comparative analysis of A*, RRT*, APF, GA, and DQN within our scenario is presented in Table 2 and Figure 17.

In the future, we will further integrate real-time illumination and Earth–Moon communication, along with the rover's mobility and its interactions with the environment, such as wheel–regolith interface friction and slippage. This integration will facilitate the development of a comprehensive and robust algorithm, employing reinforcement learning techniques to establish an end-to-end path-planning solution.

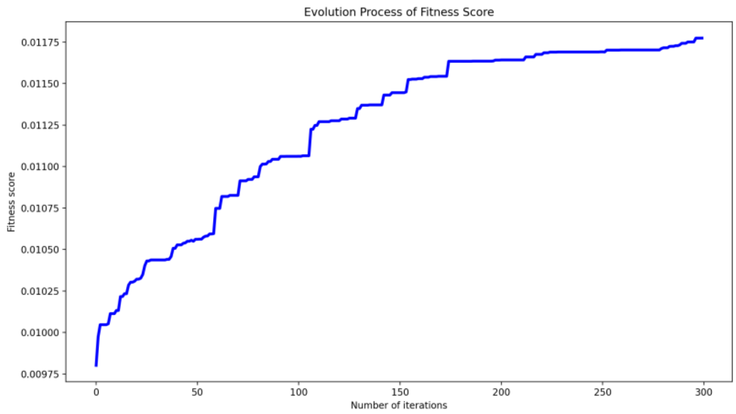

Figure 14. Evolutionary process of fitness score in the GA algorithm. The total number of iterations is set to 300 in our simulations. As the number of iterations increases, the fitness scores of the paths generated by the GA algorithm also increase until convergence.

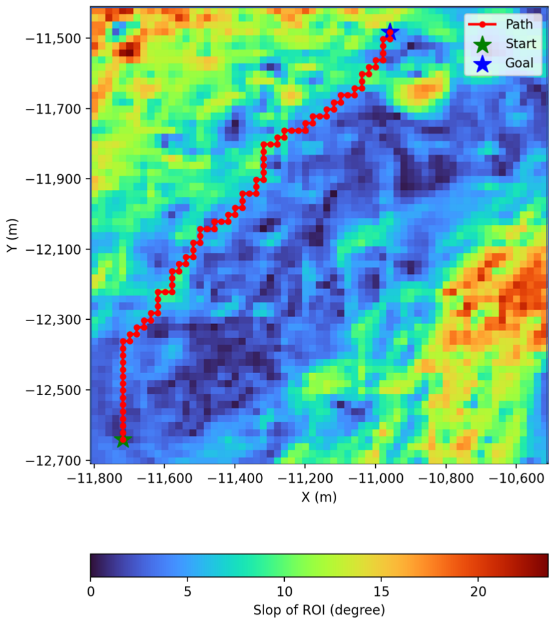

Figure 15. Path planning using the DQN algorithm considering the constraints of slope, distance, and illumination. Both the start and goal points are randomly provided, which is the same as Figure 10. The slope map serves as the background.

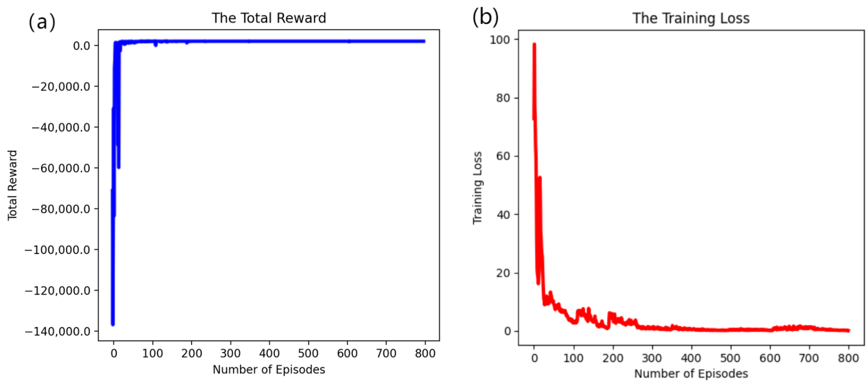

Figure 16. The training process of the DQN algorithm. (**a**) and (**b**) show the changes in total reward and training loss during the training phase as the number of episodes varies, respectively.

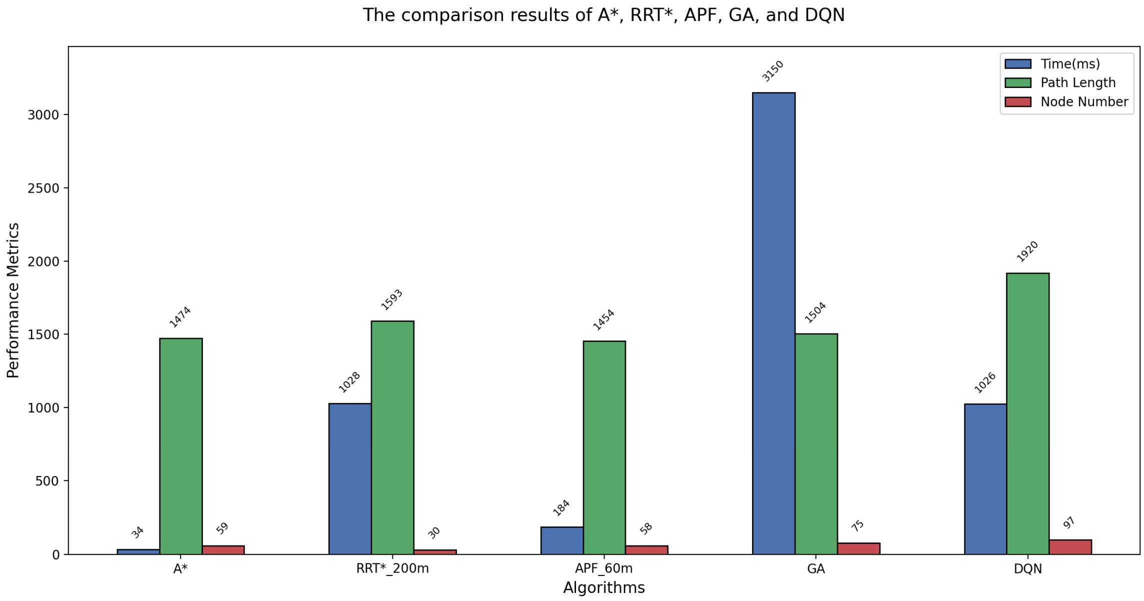

Figure 17. Comparative results of the A*, RRT*, APF, GA, and DQN algorithms. For the RRT* and APF algorithms, we selected parameter settings of 200 m and 60 m, respectively, which yielded relatively optimal paths within their respective categories. These paths are designated as RRT*_200 m and APF_60 m.

6. Summary and Conclusions

This review provides a comprehensive summary of the research progress on planetary rover path-planning algorithms in recent years. It first elaborates on several constraints that affect the path planning of the rovers, including external environmental constraints and the rover's state constraints. It also introduces the research hotspots in planetary rover path planning under different constraints, to assist scientists and engineers in better considering these constraints when designing specific path-planning algorithms. Subsequently, the existing planetary rover path-planning algorithms are categorized into two main types: rule-based path-planning algorithms and biologically inspired path-planning algorithms. The advantages and disadvantages of each type of algorithm are summarized in Table 3, and Table 4 summarizes the performance comparison of various path-planning algorithms based on the classification system proposed in this review.

Rule-based path-planning algorithms include graph search methods, potential field methods, sampling-based methods, and dynamic window methods. These algorithms can guide the movement of planetary rovers through a set of clear rules and are easy to implement. However, each of these methods has certain application limitations. Graph search-based path-planning algorithms have lower computational efficiency because they require traversing the entire environmental space, making them suitable for global path planning of rovers. Therefore, research in this area focuses on improving computational efficiency, for example, by designing more efficient heuristic functions or adopting distributed path computation strategies. Potential field-based path-planning algorithms and dynamic window algorithms both have advantages, such as high computational efficiency, good real-time performance, and the ability to handle obstacles in dynamic environments, but both of them can lead to the lunar rover getting stuck in local minima. Sampling-based path-planning algorithms are suitable for high-dimensional complex environments and can effectively handle obstacles and dynamic changes in the environment. However, these algorithms typically have slower convergence rates and are sensitive to initial solutions, so the research focus in this area is on addressing computational efficiency issues.

Table 3. The advantages and disadvantages of each type of algorithm.

Different Types of Algorithms			Advantages	Disadvantages
Rule-based Path-Planning Algorithms	Graph Search-based Algorithms		Simple	lower computational efficiency
	Potential Field-based Algorithms		1. High computational efficiency. 2. Good real-time performance. 3. Suitable for dynamic environments.	Easy to get stuck in local minima.
	Sampling-based Algorithms		Suitable for high-dimensional complex environments	1. Slower convergence rates. 2. Sensitive to initial solutions.
	Dynamic Window Approach		1. High computational efficiency. 2. Good real-time performance. 3. Suitable for dynamic environments.	Easy to get stuck in local minima.
Biologically Inspired Path-Planning Algorithms	Evolutionary Learning		Strong adaptability to dynamic environments	1. High consumption of computational resources. 2. Slower convergence speeds.
	Fuzzy Computation		Simple	Lack adaptability to complex dynamic environments.
	Machine Learning-based Algorithms	Traditional Machine Learning Algorithms	Good environmental adaptability	1. Require a large amount of data to train the models. 2. Significant computational resources and poor interpretability. 3. May lead to issues with path safety and model generalization
		Deep Learning-based Algorithms	Powerful feature extraction and processing capabilities	
		Reinforcement Learning-based Algorithms	Achieving end-to-end path planning	

Table 4. Performance comparison of planetary rover path-planning algorithms.

Category	Optimality	Completeness	Deterministic	Resource Requirement
Graph Search	optimal	Yes	Yes	Depends on graph size
APF	Sub-optimal	Not ensured	Yes	Low
Sampling-based	Asymptotical	Not ensured	No	Depends on sampling density and environmental complexity
DWA	Sub-optimal	Not ensured	Yes	Medium
Evolutionary Learning	Heuristic	Not ensured	No	Medium
Fuzzy Computation	Heuristic	Not ensured	No	Low
Machine Learning	Heuristic	Not ensured	No	High

With the rapid development of artificial intelligence, biologically inspired intelligent path-planning algorithms for planetary rovers have gradually emerged, attracting widespread attention from researchers. Biologically inspired path-planning algorithms mainly include evolutionary algorithms, fuzzy computing, and machine learning algorithms. Among them, evolutionary learning-based algorithms include genetic algorithms and swarm optimization algorithms. Evolutionary learning-based algorithms have strong adaptability to dynamic environments. However, they require multiple iterations and optimizations for computation, which leads to a high consumption of computational resources and results in slower convergence speeds. Additionally, these algorithms also depend on the quality of the initial population. Fuzzy computing-based algorithms have the advantage of being simple and easy to implement, but the design of fuzzy rule bases is often very complex, and these algorithms lack adaptability to complex dynamic environments. The environmental adaptability of machine learning algorithms is superior, allowing them to handle the complex dynamic environment of the planetary surface. Traditional machine learning algorithms are typically used to assist rovers in path planning, improving the efficiency of the rover's path planning. Deep learning algorithms, due to their powerful feature extraction and processing capabilities, are often used for environmental modeling under complex planetary surface conditions to enhance the rover's autonomous exploration capabilities. Reinforcement learning enables the rover to learn the optimal path gradually through continuous interaction with the environment and trial-and-error processes, achieving end-to-end path planning for the rover. However, applying machine learning algorithms to planetary rover path planning also faces several challenges. For example, the algorithms require a large amount of data to train the models, the model training phase consumes significant computational resources, and machine learning models, especially deep learning models, have poor interpretability, which may lead to issues with path safety and model generalization in complex scenarios.

Author Contributions: Conceptualization, Q.M. and G.W.; investigation, Q.M.; writing—original draft preparation, Q.M.; writing—review and editing, Q.M. and G.W.; funding acquisition, G.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Anhui Natural Science Foundation, grant number 2408085Y021, and the National Natural Science Foundation of China, grant number 42473053.

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

References

1. Vondrak, R.; Keller, J.; Chin, G.; Garvin, J. Lunar Reconnaissance Orbiter (LRO): Observations for lunar exploration and science. *Space Sci. Rev.* **2010**, *150*, 7–22. [CrossRef]
2. Harvey, B. *Soviet and Russian Lunar Exploration*; Springer: Berlin/Heidelberg, Germany, 2007.
3. Li, C.; Wang, C.; Wei, Y.; Lin, Y. China's present and future lunar exploration program. *Science* **2019**, *365*, 238–239. [CrossRef]
4. Weiren, W.; Jizhong, L.; Yuhua, T.; Dengyun, Y.; Guobin, Y.; Zhe, Z. China lunar exploration program. *J. Deep Space Explor.* **2019**, *6*, 405–416.
5. Dengyun, Y.; Xueying, W.; Weiren, W. Review of technology development for Chinese lunar exploration program. *J. Deep Space Explor.* **2016**, *3*, 307–314.
6. Ding, L.; Zhou, R.; Yuan, Y.; Yang, H.; Li, J.; Yu, T.; Liu, C.; Wang, J.; Li, S.; Gao, H. A 2-year locomotive exploration and scientific investigation of the lunar farside by the Yutu-2 rover. *Sci. Robot.* **2022**, *7*, eabj6660. [CrossRef] [PubMed]
7. Johnston, R.S.; Hull, W.E.; Dietlein, L.; Berry, C. Apollo missions. In *Biomedical Results of Apollo*; NASA: Washington, DC, USA, 1975; pp. 9–40.
8. Grotzinger, J.P.; Crisp, J.; Vasavada, A.R.; Anderson, R.C.; Baker, C.J.; Barry, R.; Blake, D.F.; Conrad, P.; Edgett, K.S.; Ferdowski, B. Mars science laboratory mission and science investigation. *Space Sci. Rev.* **2012**, *170*, 5–56. [CrossRef]
9. Farley, K.A.; Williford, K.H.; Stack, K.M.; Bhartia, R.; Chen, A.; de la Torre, M.; Hand, K.; Goreva, Y.; Herd, C.D.; Hueso, R. Mars 2020 mission overview. *Space Sci. Rev.* **2020**, *216*, 142. [CrossRef]
10. Tian, H.; Zhang, T.; Jia, Y.; Peng, S.; Yan, C. Zhurong: Features and mission of China's first Mars rover. *Innovation* **2021**, *2*, 100121. [CrossRef]

11. Smith, M.; Craig, D.; Herrmann, N.; Mahoney, E.; Krezel, J.; McIntyre, N.; Goodliff, K. The Artemis program: An overview of NASA's activities to return humans to the moon. In Proceedings of the 2020 IEEE Aerospace Conference, Big Sky, MT, USA, 7–14 March 2020; pp. 1–10.
12. Wang, C.; Jia, Y.; Xue, C.; Lin, Y.; Liu, J.; Fu, X.; Xu, L.; Huang, Y.; Zhao, Y.; Xu, Y. Scientific objectives and payload configuration of the Chang'E-7 mission. *Natl. Sci. Rev.* **2024**, *11*, nwad329. [CrossRef]
13. Wong, C.; Yang, E.; Yan, X.-T.; Gu, D. Adaptive and intelligent navigation of autonomous planetary rovers—A survey. In Proceedings of the 2017 NASA/ESA Conference on Adaptive Hardware and Systems (AHS), Pasadena, CA, USA, 24–27 July 2017; pp. 237–244.
14. Yu, X.; Wang, P.; Zhang, Z. Learning-based end-to-end path planning for lunar rovers with safety constraints. *Sensors* **2021**, *21*, 796. [CrossRef]
15. Sutoh, M.; Otsuki, M.; Wakabayashi, S.; Hoshino, T.; Hashimoto, T. The right path: Comprehensive path planning for lunar exploration rovers. *IEEE Robot. Autom. Mag.* **2015**, *22*, 22–33. [CrossRef]
16. Yu, X.; Huang, Q.; Wang, P.; Guo, J. Comprehensive global path planning for lunar rovers. In Proceedings of the 2020 3rd International Conference on Unmanned Systems (ICUS), Harbin, China, 27–28 November 2020; pp. 505–510.
17. Takemura, R.; Ishigami, G. Traversability-based RRT* for planetary rover path planning in rough terrain with LIDAR point cloud data. *J. Robot. Mechatron.* **2017**, *29*, 838–846. [CrossRef]
18. Bai, C.; Guo, J.; Guo, L.; Song, J. Deep multi-layer perception based terrain classification for planetary exploration rovers. *Sensors* **2019**, *19*, 3102. [CrossRef]
19. Zhang, H.; Jiang, F.; Liu, C.; Zhang, Z.; Li, Q. Review of autonomous path planning for planetary rovers. *Chin. J. Eng.* **2024**, *46*, 2063–2075.
20. Hoa, D.K.; Dung, L.; Dzung, N.T. Efficient determination of disparity map from stereo images with modified sum of absolute differences (SAD) algorithm. In Proceedings of the 2013 International Conference on Advanced Technologies for Communications (ATC 2013), Ho Chi Minh City, Vietnam, 16–18 October 2013; pp. 657–660.
21. Chen, J.; Xing, Y.; Li, Z.; Mao, X.; Teng, B.; Liu, X.; Jia, Y.; Gu, P. Autonomous environment perception and obstacle avoidance technologies of Zhurong Mars rover. *Sci. Sin. Technol.* **2022**, *52*, 1186–1197. [CrossRef]
22. Hart, P.E.; Nilsson, N.J.; Raphael, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Syst. Sci. Cybern.* **1968**, *4*, 100–107. [CrossRef]
23. Karaman, S.; Walter, M.R.; Perez, A.; Frazzoli, E.; Teller, S. Anytime motion planning using the RRT. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 1478–1483.
24. Han, W.-G.; Baek, S.-M.; Kuc, T.-Y. Genetic algorithm based path planning and dynamic obstacle avoidance of mobile robots. In Proceedings of the 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation, Orlando, FL, USA, 12–15 October 1997; pp. 2747–2751.
25. Lamini, C.; Benhlima, S.; Elbekri, A. Genetic algorithm based approach for autonomous mobile robot path planning. *Procedia Comput. Sci.* **2018**, *127*, 180–189. [CrossRef]
26. Wenyuan, Z.; Jifeng, G.; Chengchao, B. Adaptive Path Planning for Unmanned Planetary Rover with Dynamic Obstacle. In Proceedings of the 2019 IEEE International Conference on Unmanned Systems (ICUS), Beijing, China, 17–19 October 2019; pp. 730–735.
27. Khatib, O. Real-time obstacle avoidance for manipulators and mobile robots. *Int. J. Robot. Res.* **1986**, *5*, 90–98. [CrossRef]
28. Manteaux, T.; Rodríguez-Martínez, D.; Rajan, R.T. RAPF: Efficient path planning for lunar microrovers. *arXiv* **2024**, arXiv:2405.16659.
29. Bassil, Y. Neural network model for path-planning of robotic rover systems. *arXiv* **2012**, arXiv:1204.0183.
30. Zhang, J.; Xia, Y.; Shen, G. A novel learning-based global path planning algorithm for planetary rovers. *Neurocomputing* **2019**, *361*, 69–76. [CrossRef]
31. Higa, S.; Iwashita, Y.; Otsu, K.; Ono, M.; Lamarre, O.; Didier, A.; Hoffmann, M. Vision-based estimation of driving energy for planetary rovers using deep learning and terramechanics. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3876–3883. [CrossRef]
32. Tanaka, T.; Malki, H. A Deep Learning Approach to Lunar Rover Global Path Planning Using Environmental Constraints and the Rover Internal Resource Status. *Sensors* **2024**, *24*, 844. [CrossRef] [PubMed]
33. Park, B.-J.; Chung, H.-J. Deep reinforcement learning-based failure-safe motion planning for a 4-wheeled 2-steering lunar rover. *Aerospace* **2023**, *10*, 219. [CrossRef]
34. Bai, J.H.; Oh, Y.-J. Global path planning of lunar rover under static and dynamic constraints. *Int. J. Aeronaut. Space Sci.* **2020**, *21*, 1105–1113. [CrossRef]
35. Biesiadecki, J.J.; Maimone, M.W. The mars exploration rover surface mobility flight software driving ambition. In Proceedings of the 2006 IEEE Aerospace Conference, Big Sky, MT, USA, 4–11 March 2006; p. 15.
36. Jianxin, C.; Yan, X.; Baoyi, T.; Xiaoyan, M.; Xiang, L.; Yong, J.; Jin, Z.; Lei, W. Guidance, navigation and control technologies of Chang'E-3 lunar rover. *Sci. Sin.* **2014**, *44*, 461.

37. Iagnemma, K.; Genot, F.; Dubowsky, S. Rapid physics-based rough-terrain rover planning with sensor and control uncertainty. In Proceedings of the Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C), Detroit, MI, USA, 10–15 May 1999; pp. 2286–2291.
38. Cunningham, C.; Amato, J.; Jones, H.L.; Whittaker, W.L. Accelerating energy-aware spatiotemporal path planning for the lunar poles. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 4399–4406.
39. Inoue, H.; Adachi, S. Spatio-temporal path planning for lunar polar exploration with robustness against schedule delay. *Trans. Jpn. Soc. Aeronaut. Space Sci.* **2021**, *64*, 304–311. [CrossRef]
40. Brunner, M.; Fiolka, T.; Schulz, D.; Schlick, C.M. Design and comparative evaluation of an iterative contact point estimation method for static stability estimation of mobile actively reconfigurable robots. *Robot. Auton. Syst.* **2015**, *63*, 89–107. [CrossRef]
41. Farritor, S.; Dubowsky, S. Genetic planning method and its application to planetary exploration. *J. Dyn. Sys. Meas. Control* **2002**, *124*, 698–701. [CrossRef]
42. Goldberg, S.B.; Maimone, M.W.; Matthies, L. Stereo vision and rover navigation software for planetary exploration. In Proceedings of the Proceedings, IEEE Aerospace Conference, Big Sky, MT, USA, 9–16 March 2002; p. 5.
43. Weih, R.C., Jr.; Mattson, T.L. Modeling slope in a geographic information system. *J. Ark. Acad. Sci.* **2004**, *58*, 100–108.
44. Mahtab, A.; Narender, B.; Ajai. Satellite derived digital elevation model and terrain parameters—Generation, accuracy assessment and validation. *J. Indian Soc. Remote Sens.* **2003**, *31*, 19–24. [CrossRef]
45. Bussey, D.B.J.; Spudis, P.D.; Robinson, M.S. Illumination conditions at the lunar south pole. *Geophys. Res. Lett.* **1999**, *26*, 1187–1190. [CrossRef]
46. Plonski, P.A.; Tokekar, P.; Isler, V. Energy-efficient path planning for solar-powered mobile robots. *J. Field Robot.* **2013**, *30*, 583–601. [CrossRef]
47. Kaplan, A.; Kingry, N.; Uhing, P.; Dai, R. Time-optimal path planning with power schedules for a solar-powered ground robot. *IEEE Trans. Autom. Sci. Eng.* **2016**, *14*, 1235–1244. [CrossRef]
48. Lamarre, O.; Malhotra, S.; Kelly, J. Safe Mission-Level Path Planning for Exploration of Lunar Shadowed Regions by a Solar-Powered Rover. In Proceedings of the 2024 IEEE Aerospace Conference, Big Sky, MT, USA, 2–9 March 2024; pp. 1–14.
49. Staudinger, E.; Giubilato, R.; Schuster, M.J.; Pöhlmann, R.; Zhang, S.; Dömel, A.; Wedler, A.; Dammann, A. Terrain-aware communication coverage prediction for cooperative networked robots in unstructured environments. *Acta Astronaut.* **2023**, *202*, 799–805. [CrossRef]
50. de Curtò, J.; de Zarzà, I.; Calafate, C.T. UWB and MB-OFDM for Lunar Rover Navigation and Communication. *Mathematics* **2023**, *11*, 3835. [CrossRef]
51. Ohki, T. Online Slip Estimation for Mobile Robot Localization and Reactive Path Planning on Rough and Deformable Terrain. Ph.D. Thesis, Tohoku University, Sendai, Japan, 2013.
52. Zhang, H.; Zhang, Y.; Yang, T. A survey of energy-efficient motion planning for wheeled mobile robots. *Ind. Robot Int. J. Robot. Res. Appl.* **2020**, *47*, 607–621. [CrossRef]
53. Malenkov, M.; Volov, V. Wheel-walking propulsion unit of a planetary rover with active suspension. *Russ. Eng. Res.* **2017**, *37*, 1033–1040. [CrossRef]
54. Moreland, S.; Skonieczny, K.; Wettergreen, D.; Asnani, V.; Creager, C.; Oravec, H. Inching locomotion for planetary rover mobility. In Proceedings of the 2011 Aerospace Conference, Big Sky, MT, USA, 5–12 March 2011; pp. 1–6.
55. Creager, C.; Moreland, S.; Skonieczny, K.; Johnson, K.; Asnani, V.; Gilligan, R. Benefit of "Push-Pull" Locomotion for Planetary Rover Mobility. In *Earth and Space 2012: Engineering, Science, Construction, and Operations in Challenging Environments*; American Society of Civil Engineers: Reston, VA, USA, 2012; pp. 11–20.
56. Ishigami, G.; Nagatani, K.; Yoshida, K. Path planning and evaluation for planetary rovers based on dynamic mobility index. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25–30 September 2011; pp. 601–606.
57. Dijkstra, E.W. A note on two problems in connexion with graphs. In *Edsger Wybe Dijkstra: His Life, Work, and Legacy*; ACM Books: New York, NY, USA, 2022; pp. 287–290.
58. Stentz, A. Optimal and efficient path planning for partially-known environments. In Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA, USA, 8–13 May 1994; pp. 3310–3317.
59. Muñoz, P.; R-Moreno, M.D.; Castaño, B. 3Dana: A path planning algorithm for surface robotics. *Eng. Appl. Artif. Intell.* **2017**, *60*, 175–192. [CrossRef]
60. Zhang, H.; Jiang, F.; Li, Q. An improved path planning and tracking control method for planetary exploration rovers with traversable tolerance. *Biomim. Intell. Robot.* **2025**, *5*, 100219. [CrossRef]
61. Hong, Z.; Tu, B.; Tong, X.; Pan, H.; Zhou, R.; Zhang, Y.; Han, Y.; Wang, J.; Yang, S.; Ma, Z. A fast large-scale path planning method on lunar DEM using distributed tile pyramid strategy. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *16*, 344–355. [CrossRef]

62. Nasuha, A.; Priambodo, A.; Pratama, G. Vortex artificial potential field for mobile robot path planning. *J. Phys. Conf. Ser.* **2022**, *2406*, 012001. [CrossRef]
63. Hossain, M.A.; Ferdous, I. Autonomous robot path planning in dynamic environment using a new optimization technique inspired by bacterial foraging technique. *Robot. Auton. Syst.* **2015**, *64*, 137–141. [CrossRef]
64. Diab, M.; Mohammadkarimi, M.; Rajan, R.T. Artificial Potential Field-Based Path Planning for Cluttered Environments. In Proceedings of the 2023 IEEE Aerospace Conference, Big Sky, MT, USA, 4–11 March 2023; pp. 1–8.
65. Kavraki, L.E.; Svestka, P.; Latombe, J.-C.; Overmars, M.H. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robot. Autom.* **1996**, *12*, 566–580. [CrossRef]
66. Karaman, S.; Frazzoli, E. Sampling-based algorithms for optimal motion planning. *Int. J. Robot. Res.* **2011**, *30*, 846–894. [CrossRef]
67. Lavalle, S.M. *Rapidly-Exploring Random Trees: A New Tool for Path Planning*; Technical Report; Iowa State University: Ames, IA, USA, 1998.
68. Hidalgo-Paniagua, A.; Bandera, J.P.; Ruiz-de-Quintanilla, M.; Bandera, A. Quad-RRT: A real-time GPU-based global path planner in large-scale real environments. *Expert Syst. Appl.* **2018**, *99*, 141–154. [CrossRef]
69. Wang, J.; Chi, W.; Li, C.; Wang, C.; Meng, M.Q.H. Neural RRT*: Learning-based optimal path planning. *IEEE Trans. Autom. Sci. Eng.* **2020**, *17*, 1748–1758. [CrossRef]
70. Zhang, F.; Li, N.; Xue, T.; Zhu, Y.; Yuan, R.; Fu, Y. An improved dynamic window approach integrated global path planning. In Proceedings of the 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO), Dali, China, 6–8 December 2019; pp. 2873–2878.
71. Chang, L.; Shan, L.; Jiang, C.; Dai, Y. Reinforcement based mobile robot path planning with improved dynamic window approach in unknown environment. *Auton. Robot.* **2021**, *45*, 51–76. [CrossRef]
72. Huiting, W.; Meng, Y.; Yuye, L.; Tao, H.; Bo, Z. Autonomous Navigation Path Planning Algorithm for Rovers in Lunar South Pole Surface. *J. Deep Space Explor.* **2023**, *10*, 598–607.
73. Tianyi, Y.; Jiangtao, F.; Lichun, L.; Xiao, C. Study on path planning method of lunar rover. *J. Deep Space Explor.* **2019**, *6*, 384–390.
74. Zhou, R.; Liu, Y.; Hong, Z.; Pan, H.; Zhang, Y.; Han, Y.; Tao, J. A Safe and Efficient Global Path-Planning Method Considering Multiple Environmental Factors of the Moon Using a Distributed Computation Strategy. *Remote Sens.* **2025**, *17*, 924. [CrossRef]
75. Ono, M.; Fuchs, T.J.; Steffy, A.; Maimone, M.; Yen, J. Risk-aware planetary rover operation: Autonomous terrain classification and path planning. In Proceedings of the 2015 IEEE Aerospace Conference, Big Sky, MT, USA, 7–14 March 2015; pp. 1–10.
76. Ge, S.S.; Cui, Y.J. Dynamic motion planning for mobile robots using potential field method. *Auton. Robot.* **2002**, *13*, 207–222. [CrossRef]
77. Raja, R.; Dutta, A.; Venkatesh, K.S. New potential field method for rough terrain path planning using genetic algorithm for a 6-wheel rover. *Robot. Auton. Syst.* **2015**, *72*, 295–306. [CrossRef]
78. Fox, D.; Burgard, W.; Thrun, S. The dynamic window approach to collision avoidance. *IEEE Robot. Autom. Mag.* **1997**, *4*, 23–33. [CrossRef]
79. Sanchez-Ibanez, J.R.; Pérez-del-Pulgar, C.J.; García-Cerezo, A. Path planning for autonomous mobile robots: A review. *Sensors* **2021**, *21*, 7898. [CrossRef] [PubMed]
80. Tang, K.-S.; Man, K.-F.; Kwong, S.; He, Q. Genetic algorithms and their applications. *IEEE Signal Process. Mag.* **1996**, *13*, 22–37. [CrossRef]
81. Lanfeng, Z.; Lina, Y.; Hua, F. Lunar Rover Path Planning Based on Comprehensive Genetic Algorithm Based on Slip Prediction. *J. Phys. Conf. Ser.* **2019**, *1267*, 012097. [CrossRef]
82. Dorigo, M.; Maniezzo, V.; Colorni, A. Ant system: Optimization by a colony of cooperating agents. *IEEE Trans. Syst.* **1996**, *26*, 29–41. [CrossRef]
83. Zhu, S.; Zhu, W.; Zhang, X.; Cao, T. Path planning of lunar robot based on dynamic adaptive ant colony algorithm and obstacle avoidance. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 1729881419898979. [CrossRef]
84. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; pp. 1942–1948.
85. Peng, S.; Jia, Y. Global path planning for lunar rover based on Particle Swarm Optimization algorithm. In Proceedings of the 2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM), Qingdao, China, 17–19 September 2011; pp. 83–88.
86. Wang, L.; Liu, L.; Lu, X. Robot Path Planning Based on Generative Learning Particle Swarm Optimization. *IEEE Access* **2024**, *12*, 130063–130072. [CrossRef]
87. Seraji, H.; Bon, B. *Autonomous Navigation of Planetary Rovers: A Fuzzy Logic Approach*; JPL Internal Document; Jet Propulsion Laboratory: Pasadena, CA, USA, 1998.
88. Zavlangas, P.G.; Tzafestas, S.G. Motion control for mobile robot obstacle avoidance and navigation: A fuzzy logic-based approach. *Syst. Anal. Model. Simul.* **2003**, *43*, 1625–1637. [CrossRef]

89. Tanaka, Y.; Ji, Y.; Yamashita, A.; Asama, H. Fuzzy based traversability analysis for a mobile robot on rough terrain. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 3965–3970.
90. Brooks, C.A.; Iagnemma, K. Self-supervised terrain classification for planetary surface exploration rovers. *J. Field Robot.* **2012**, *29*, 445–468. [CrossRef]
91. Otsu, K.; Ono, M.; Fuchs, T.J.; Baldwin, I.; Kubota, T. Autonomous terrain classification with co-and self-training approach. *IEEE Robot. Autom. Lett.* **2016**, *1*, 814–819. [CrossRef]
92. Zhang, H.; Dai, X.; Sun, F.; Yuan, J. Terrain classification in field environment based on Random Forest for the mobile robot. In Proceedings of the 2016 35th Chinese Control Conference (CCC), Chengdu, China, 27–29 July 2016; pp. 6074–6079.
93. Fan, L.; Yuan, J.; Zha, K. TerSeg: A dual-branch semantic segmentation network for Mars terrain and autonomous path planning. *Expert Syst. Appl.* **2025**, *270*, 126397. [CrossRef]
94. Rothrock, B.; Kennedy, R.; Cunningham, C.; Papon, J.; Heverly, M.; Ono, M. Spoc: Deep learning-based terrain classification for mars rover missions. In *AIAA SPACE 2016*; American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2016; p. 5539.
95. Tamar, A.; Levine, S.; Abbeel, P.; Wu, Y.; Thomas, G. Value Iteration Networks. In Proceedings of the Advances in Neural Information Processing Systems 29 (NIPS 2016), Barcelona, Spain, 5–10 December 2016.
96. Pflueger, M.; Agha, A.; Sukhatme, G.S. Rover-IRL: Inverse Reinforcement Learning with Soft Value Iteration Networks for Planetary Rover Path Planning. *IEEE Robot. Autom. Lett.* **2019**, *4*, 1387–1394. [CrossRef]
97. Hu, T.; Cao, T.; Zheng, B.; Zhang, H.; Ni, M. Large-scale Autonomous Navigation and Path Planning of Lunar Rover via Deep Reinforcement Learning. In Proceedings of the 2021 China Automation Congress (CAC), Beijing, China, 22–24 October 2021; pp. 2050–2055.
98. Hu, R.; Zhang, Y. Fast path planning for long-range planetary roving based on a hierarchical framework and deep reinforcement learning. *Aerospace* **2022**, *9*, 101. [CrossRef]
99. Ram, A.; Boone, G.; Arkin, R.; Pearce, M. Using genetic algorithms to learn reactive control parameters for autonomous robotic navigation. *Adapt. Behav.* **1994**, *2*, 277–305. [CrossRef]
100. Lanfeng, Z.; Lina, Y.; Hua, F. Research on slip prediction path planning based on an ant colony algorithm. *J. East China Norm. Univ.* **2020**, *2020*, 72.
101. Katiyar, S.; Dutta, A. PSO based path planning and dynamic obstacle avoidance in CG space of a 10 DOF Rover. In Proceedings of the 2021 5th International Conference on Advances in Robotics, Kanpur, India, 30 June–3 July 2021; pp. 1–6.
102. Howard, A.; Seraji, H.; Werger, B. Fuzzy terrain-based path planning for planetary rovers. In Proceedings of the 2002 IEEE World Congress on Computational Intelligence. 2002 IEEE International Conference on Fuzzy Systems, Honolulu, HI, USA, 12–17 May 2002; pp. 316–320.
103. Garcia, A.; Barrientos Cruz, A.; Medina, A.; Colmenarejo, P.; Mollinedo, L.; Rossi, C. 3D Path planning using a fuzzy logic navigational map for Planetary Surface Rovers. In Proceedings of the 11 th Symposium on Advanced Space Technologies in Robotics and Automation, Noordwijk, The Netherlands, 12–14 April 2011.
104. Tarokh, M. Hybrid intelligent path planning for articulated rovers in rough terrain. *Fuzzy Sets Syst.* **2008**, *159*, 2927–2937. [CrossRef]
105. Nampoothiri, M.H.; Vinayakumar, B.; Sunny, Y.; Antony, R. Recent developments in terrain identification, classification, parameter estimation for the navigation of autonomous robots. *SN Appl. Sci.* **2021**, *3*, 480. [CrossRef]
106. Peijian, Y.; Linzhi, M.; Jinan, M.; Qiang, W.; Ying, L.; Yu, D.; Shuo, W. Suggestions on artificial intelligence technology application and development in deep space exploration. *J. Deep Space Explor.* **2019**, *6*, 303–316, 383.
107. Estlin, T.A.; Bornstein, B.J.; Gaines, D.M.; Anderson, R.C.; Thompson, D.R.; Burl, M.; Castano, R.; Judd, M. Aegis automated science targeting for the mer opportunity rover. *ACM Trans. Intell. Syst. Technol.* **2012**, *3*, 1–19. [CrossRef]
108. Moreira, I.; Rivas, J.; Cruz, F.; Dazeley, R.; Ayala, A.; Fernandes, B. Deep reinforcement learning with interactive feedback in a human–robot environment. *Appl. Sci.* **2020**, *10*, 5574. [CrossRef]
109. Ono, M.; Rothrock, B.; Otsu, K.; Higa, S.; Iwashita, Y.; Didier, A.; Islam, T.; Laporte, C.; Sun, V.; Stack, K. Maars: Machine learning-based analytics for automated rover systems. In Proceedings of the 2020 IEEE aerospace conference, Big Sky, MT, USA, 7–14 March 2020; pp. 1–17.
110. Yutong, J.; Zhang, S.; Bin, L.; Kaichang, D.; Bin, X.; Jing, N.; Chenxu, Z.; Gang, W. A robust method for large-scale route optimization on lunar surface utilizing a multi-level map model. *Chin. J. Aeronaut.* **2025**, *38*, 103388.
111. Chen, Y.; Wei, G.; Zhang, H.; Lu, J.; Pang, F. A Spatiotemporal U-Net-Based Data Preprocessing Pipeline for Sun-Synchronous Path Planning in Lunar South Polar Exploration. *Remote Sens.* **2025**, *17*, 1589. [CrossRef]
112. Endo, M.; Taniai, T.; Ishigami, G. Deep Probabilistic Traversability with Test-time Adaptation for Uncertainty-aware Planetary Rover Navigation. *arXiv* **2024**, arXiv:2409.00641.
113. Meng, S.; Wang, Y.; Yang, C.-F.; Peng, N.; Chang, K.-W. Llm-a*: Large language model enhanced incremental heuristic search on path planning. *arXiv* **2024**, arXiv:2407.02511.

114. Xiao, L.; Yamasaki, T. LLM-Advisor: An LLM Benchmark for Cost-efficient Path Planning across Multiple Terrains. *arXiv* **2025**, arXiv:2503.01236.
115. Doma, P.; Arab, A.; Xiao, X. LLM-Enhanced Path Planning: Safe and Efficient Autonomous Navigation with Instructional Inputs. *arXiv* **2024**, arXiv:2412.02655.
116. Zhao, Y.; Wu, Q.; Wang, Y.; Tai, Y.-W.; Tang, C.-K. Dynamic Path Navigation for Motion Agents with LLM Reasoning. *arXiv* **2025**, arXiv:2503.07323.
117. Daftry, S.; Abcouwer, N.; Del Sesto, T.; Venkatraman, S.; Song, J.; Igel, L.; Byon, A.; Rosolia, U.; Yue, Y.; Ono, M. Mlnav: Learning to safely navigate on martian terrains. *IEEE Robot. Autom. Lett.* **2022**, *7*, 5461–5468. [CrossRef]
118. Lu, S.; Xu, R.; Li, Z.; Wang, B.; Zhao, Z. Lunar Rover Collaborated Path Planning with Artificial Potential Field-Based Heuristic on Deep Reinforcement Learning. *Aerospace* **2024**, *11*, 253. [CrossRef]
119. Gläser, P.; Scholten, F.; De Rosa, D.; Figuera, R.M.; Oberst, J.; Mazarico, E.; Neumann, G.; Robinson, M. Illumination conditions at the lunar south pole using high resolution Digital Terrain Models from LOLA. *Icarus* **2014**, *243*, 78–90. [CrossRef]
120. De Rosa, D.; Bussey, B.; Cahill, J.T.; Lutz, T.; Crawford, I.A.; Hackwill, T.; van Gasselt, S.; Neukum, G.; Witte, L.; McGovern, A. Characterisation of potential landing sites for the European Space Agency's Lunar Lander project. *Planet. Space Sci.* **2012**, *74*, 224–246. [CrossRef]