**Journal of Computing and Electronic Information Management**
ISSN: 2413-1660 | Vol. 20, No. 3, 2026

# Unstructured Road Obstacle Avoidance Path Planning Algorithm Based on BEV rid Map

**Mingjie Shi [1, 2, 3], Bing Li [1, 2, 3, *] and Wubin Xu [1, 2, 3]**

[1] School of Mechanical and Automotive Engineering, Guangxi University of Science and Technology, Liuzhou 545006, China
[2] Engineering Research Center of Advanced Engineering Equipment, University of Guangxi, Liuzhou 545006, China
[3] Engineering Research Center of Advanced Engineering Equipment Intelligent Manufacturing, Guangxi Zhuang Autonomous Region, Liuzhou 545006, China

*** Corresponding author**

**Abstract:** In view of the narrow, unstructured road surfaces, irregular distribution of obstacles, and the large body and steering lag characteristics of mining trucks, traditional path planning algorithms tend to generate edge-hugging paths or infeasible broken paths, posing collision risks. Based on high-precision BEV grid maps generated by LiDAR-camera multi-sensor fusion, this paper proposes an improved hybrid A* obstacle avoidance path planning algorithm that combines safety potential fields with kinematic constraints. The algorithm constructs an environmental safety potential field through Euclidean distance transformation to quantify collision risks in the grid space; it expands the A* algorithms state space and improves the cost function by introducing steering penalties and maximum steering angle limits to ensure kinematic feasibility of the path; finally, cubic B-spline curves are used to smooth the discrete path, producing a curvature-continuous trajectory. Simulation results show that, compared with the traditional A* algorithm, the proposed algorithm increases the minimum safety distance between the path and obstacles by more than 17 times, effectively eliminates broken paths, and its planning time meets the real-time control requirements for low-speed operation of mining trucks, providing safe and smooth local navigation trajectories for autonomous mining vehicles on unstructured roads.

**Keywords:** BEV grid map; Safety potential field; Hybrid A* algorithm; B-spline curve.

## 1. Introduction

Unstructured environments have become the core challenge scenario for the implementation of autonomous driving and mobile robot path planning technology due to the lack of fixed road markings, random distribution of obstacles, and complex and variable terrain. From open-pit mines, disaster rescue sites to agricultural field operations, these environments pose far higher requirements for the robustness of mobile platform environmental perception, real-time path planning, and obstacle avoidance safety than structured roads. In recent years, the rapid development of deep reinforcement learning, sampling algorithms, and hybrid A * technologies has provided diverse solutions for path planning in unstructured environments. Wu et al.[1]proposed an expert guided deep reinforcement learning method APG-DDQN for unmanned surface vehicles (AUSVs), which achieved local path planning in complex marine environments through modal switching, effectively improving the models adaptability to environmental disturbances; For semi-structured agricultural scenarios, Katikaridis et al.[2]adopted a centralized UAV-UGV collaborative strategy to achieve active path planning, providing support for efficient operation of agricultural robots in uneven fields. In dense 3D point cloud environments, Sazonov et al.[3] proposed the entropy regularized S3PM planning algorithm, which improves the planning efficiency of mobile robots in complex obstacle groups by constraining the sampling space.

Regarding the optimization of sampling algorithms, Loulou et al.[4] introduced a hybrid attention mechanism into the RRT algorithm, learning spatial sampling priors to accelerate path convergence and solve the problem of low search efficiency in unstructured environments of traditional RRT; Zhang et al.[5] improved the two-stage Theta algorithm to address the challenges posed by uncertain obstacles in rescue scenarios, ensuring path safety while enhancing real-time planning. In terms of visual guidance and multimodal fusion, Wang et al. [6] integrated image processing, path planning, and trajectory control to achieve visual autonomous navigation of robots in unstructured environments, providing ideas for the linkage between environmental perception and planning in harsh working conditions. Zhu et al.[7] proposed a dual layer hybrid A algorithm based on phase window, which balances global optimality and local obstacle avoidance ability in unstructured environments; You et al.[8] designed a global local path planning framework with hazard constraints for hexapod robots, ensuring their fault-tolerant motion ability in rugged terrain.

Although significant progress has been made in the field of unstructured environmental path planning, there are still shortcomings for extreme industrial scenarios such as open-pit mines. On the one hand, most algorithms have not fully considered the interference of harsh working conditions such as rain, fog, and dirt on obstacle detection, and the robustness of environmental models is insufficient; On the other hand, the kinematic constraints and operational requirements of heavy-duty platforms such as mining loading machinery make it difficult for existing algorithms to balance path smoothness, fuel economy, and real-time performance. Therefore, this article focuses on the unstructured environment of open-pit mines and proposes an improved and efficient path planning algorithm based on the obstacle perception characteristics under harsh working conditions, in order to achieve safe and efficient autonomous navigation of heavy-duty machinery.

# 2. Research Methods

## 2.1. Overall obstacle avoidance algorithm process

The obstacle avoidance path planning system proposed in this article takes the binary BEV grid map generated by multi-sensor fusion of LiDAR and camera as input, and finally outputs smooth trajectory points that meet the kinematic characteristics of mining trucks, providing execution basis for the underlying vehicle control module. The overall process of the algorithm is divided into three core steps, which progress layer by layer and are coupled with each other to form a complete obstacle avoidance planning loop. The overall algorithm, as shown in the figure, mainly consists of three core components:

(1) Construction of secure potential field: Using Euclidean distance transform to transform the binary grid map into a potential field map containing "risk cost", providing security inspiration for search algorithms.

(2) Constrained path search: Improve the state space of traditional A * algorithm, introduce vehicle heading dimension, and integrate potential field cost and steering penalty in the cost function to generate kinetically feasible discrete paths.

(3) Smooth trajectory optimization: Using cubic B-spline curves to fit and interpolate discrete path points, eliminating path jitter and ensuring curvature continuity.

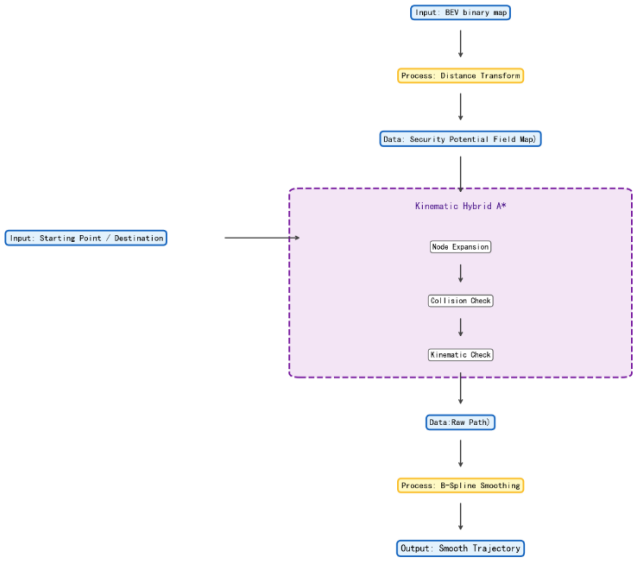



**Fig. 1** Algorithm flowchart

## 2.2. Construction of Safety Potential Field Based on Distance Transformation

Firstly, perform Euclidean distance transformation on the binary BEV grid map for each "drivable" pixel in the map $(row, col)$ , Calculate its nearest obstacle pixel point$Map_{obs_i}$Euclidean distance$d(row, col)$;

$$d(row, col) = \min_{Map_{obs_i} \in Map_{obs}} \left\| (row, col) - Map_{obs_i} \right\|_2 \quad (1)$$

Among them, $Map_{obs}$ is the set of obstacle points. The greater the distance, the farther away the obstacle is from the location, and the higher the safety. In code implementation, utilize the distanceTransform function in the OpenCv library for efficient computation.

## 2.3. Design of Potential Field Cost Function

In order to incorporate distance information into the cost function of path search, we need to construct a potential field cost graph that is inversely proportional to distance. The potential field model defined in this article is as follows:

$$C_{potential}(row, col) = \frac{\alpha}{d(row,col)BEV\_RESOLUTION + \varepsilon} \quad (2)$$

among which

$BEV_RESOLUTION$is the BEV grid resolution (taken as 0.1m/pixel in this article);

$\alpha$ is the Safety Weight coefficient, used to adjust the strength of obstacle avoidance intention;

$\varepsilon$ is used to prevent small constants with a denominator of zero (such as 1.0).

This model constructs a virtual terrain where the value increases sharply in a hyperbolic curve as it approaches obstacles. In order to minimize the total cost, planning algorithms naturally tend to choose areas with lower $C_{potential}$ to pass through, thus establishing an "invisible barrier" between the path and obstacles.

## 2.4. Hybrid A * Algorithm Based on Kinematic Constraints

Traditional A* algorithms typically use 8-neighborhood search on raster maps, and the generated paths often contain a large number of 45 degree or 90 degree turns. This path does not satisfy the non holonomic constraint of the vehicle, which means that the vehicle cannot make sharp turns in place like an omnidirectional mobile robot. Therefore, this article has improved the traditional A* algorithm by adding heading constraints and turning penalties in the node expansion process.

### 2.4.1. Expanding State Space

In order to track the direction of vehicle movement, we expand the state space of the search node from the two-dimensional coordinate $(row, col)$ to the three-dimensional state $(row, col, dir)$ represents the current discrete heading index of the vehicle (0-7, corresponding to 8 directions including east, southeast, and south).

### 2.4.2. Improved Cost Function

The total cost function of the hybrid A * algorithm is defined as

$$F(n) = G(n) + H(n) + C_{extra}(n) \quad (3)$$

$G(n)$ Mobile cost: the cumulative Euclidean distance from the starting point to the current node.

$H(n)$ Heuristic cost: The Euclidean Heuristic distance from the current node to the endpoint.

$C_{extra}(n)$ Additional constraint cost:The core innovation of this article consists of two parts:

(1) Turning cost: $|\Delta\theta|$is the index of the angle difference between the heading of the parent node and the current expansion direction. If the vehicle keeps going straight, the cost is 0; If turning, apply penalty weight$C_{steer}$

(2) Security potential field cost: calculated in Section 2.3, which forces the path to deviate towards the safe zone.

### 2.4.3. Node Expansion and Pruning Strategies

During the search process, the algorithm performs strict kinematic pruning:

Maximum turning angle limit: For mining trucks, set the maximum allowed steering angle difference per step MAX_STEER_DIFF (e.g., 45 degrees). If the angle between

the expansion direction of a child node and the parent nodes heading exceeds this threshold, it is directly pruned and deemed unreachable. This strategy eliminates sharp-angled zigzag paths at the source. The improved algorithm pseudocode is shown in Table 1.

**Table 1.** Pseudocode of Hybrid A* Algorithm Based on Kinematic Constraints

| |
|---|
| Input:MapObs, Start, Goal, CostMap<br>output:Path |
| initialization PriorityQueue, CostSoFar, CameFrom |
| StartState = (Start.x, Start.y, -1) // The initial direction is uncertain |
| Add StartState to PriorityQueue |
| While PriorityQueue is not empty: |
| Current = PriorityQueue.pop() |
| If Dist(Current, Goal) < Threshold: Return ReconstructPath(Current) |
| For each Neighbor in 8_Directions: |
| If Neighbor is Obstacle: Continue |
| //Calculate the steering angle difference |
| DeltaAngle = Abs(Current.dir - Neighbor.dir)<br>SteerCost = DeltaAngle * SteerWeight |
| SafetyCost = CostMap[Neighbor] |
| NewCost = CostSoFar[Current] + Dist(Current, Neighbor) + SteerCost + SafetyCost |
| If NewCost < CostSoFar[Neighbor]: |
| CostSoFar[Neighbor] = NewCost |
| Add Neighbor to PriorityQueue |
| Return Failure |

## 2.5. Path smoothing processing

Although the improved A * algorithm reduces sharp turns through turn penalties, due to the discrete nature of raster maps, the generated path is still essentially a series of line segments composed of polylines. In order to meet the requirement of curvature continuity in vehicle chassis control, this paper uses cubic B-spline curves to smooth fit discrete path points. Using the splprep and splev functions in the scipy.interpolate library, parameterize the discrete waypoints output by A * into functions with respect to normalization parameters:

$$P(u) = \sum_{i=0}^{n} N_{i,k}(u) C_i \quad (4)$$

Where $C_i$ is the control point and $N_{i,k}$ is the basis function. By setting the smoothing factor $s$ and order $k_{degree} = 3$ the algorithm can eliminate jagged noise while maintaining the geometric features of the path (such as avoiding obstacles), and generate executable trajectories that satisfy $C^2$ continuity (curvature continuity).

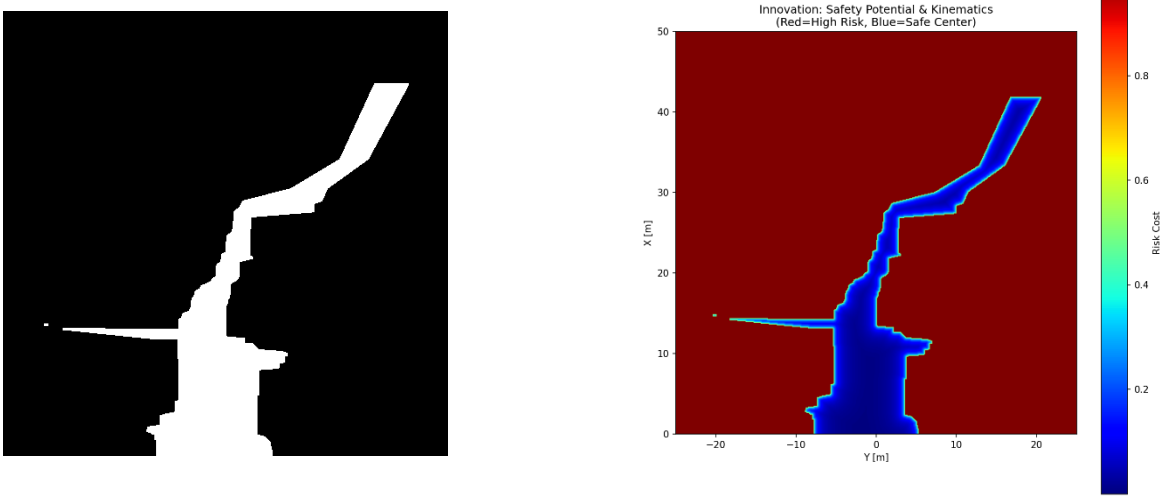
# 3. Analysis of Experimental Results

In order to verify the effectiveness of the obstacle avoidance planning algorithm proposed in this chapter, comparative experiments were conducted in a simulation environment. The experiment selected a typical unstructured road obstacle avoidance scenario, with a starting point of (0,0) and a randomly selected endpoint, and the path containing irregular obstacles invading.

## 3.1. Comparison Algorithm Settings

Baseline: The standard 2D A * algorithm. The cost function only considers Euclidean distance, without potential field and steering constraints. Ours (algorithm in this article): a hybrid A * algorithm that combines safety potential field and kinematic constraints, and includes B-Spline smoothing post-processing.

## 3.2. Visual comparison chart of safety potential field




(a) Original executable area

(b) Security potential field

**Fig. 2** Visual comparison of safety potential field

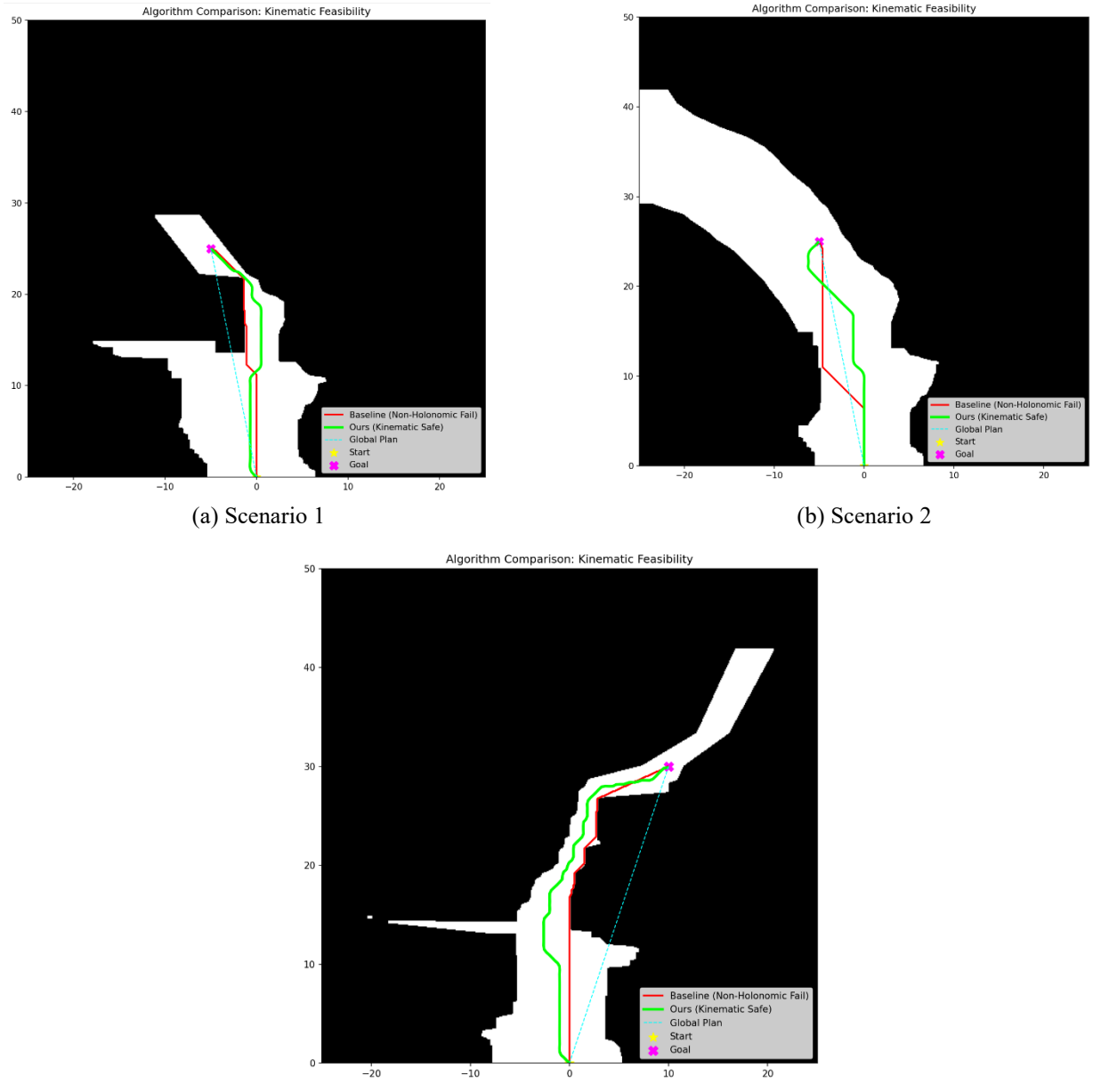
## 3.3. Comparison of Path Forms

Figure 3 shows the comparison of planning results between two algorithms in three different typical scenarios, where subgraphs (a), (b), and (c) correspond to different obstacle distribution conditions. From the comparison results of these three scenarios, it is evident that:

Safety: In all testing scenarios, the baseline path (red line) shows a tendency to ignore environmental risks in pursuit of the geometric shortest distance. Especially in the complex

obstacle intrusion conditions shown in Figures 2 (a) and (c), the baseline path cuts closely to the edge of the obstacle. For mining trucks with a wide body, this zero clearance path is prone to side scraping during actual execution. In contrast, the path generated by the algorithm in this article (green line) always demonstrates robust security. Under the repulsive effect of the safety potential field, no matter how the distribution of obstacles changes, the path in this article can consciously shift towards the "potential field trough" away from obstacles, and sufficient safety buffer space is always reserved in narrow sections and turns.

Smoothness: Due to the limitation of 8-neighborhood search and lack of dynamic constraints, the Baseline path exhibits multiple rigid line turns when bypassing obstacles (as shown in Figure 2 (b)). This frequent large angle correction not only increases the tracking difficulty of the control system, but may also lead to wear and tear of the vehicles actuator. The algorithm in this article benefits from the constraint of steering cost and the post-processing optimization of B-Spline, and the planned trajectory maintains excellent curvature continuity in all three scenarios. The path begins to smoothly change lanes before approaching obstacles, and the entire obstacle avoidance process is smooth and rounded, which is highly consistent with the actual kinematic characteristics of mining trucks.




(a) Scenario 1

(b) Scenario 2

(c) Scenario Three

**Fig. 3** Comparison of Path Planning Results

## 3.4. Safety margin and indicators

To quantitatively evaluate the performance of the algorithm, we calculated the "Minimum Clearance", which is the minimum distance from all points on the path to the nearest obstacle. This indicator directly reflects the security of the path.

From the data in Table 2, it can be seen that although the algorithm in this article slightly increases the total path length (about 1.7m) to avoid high-risk areas, it results in a qualitative change in security. The minimum clearance distance of the baseline is only 0.1m (i.e. close to the edge of the grid), but

this algorithm has increased it to 1.85m, greatly improving the fault tolerance of vehicles driving on unstructured roads. Although the introduction of potential field and steering constraints slightly increases the computation time (from 15ms to 23ms), for low-speed mining truck scenarios, this real-time performance fully meets the control requirements of 10Hz.

**Table 2.** Quantitative Comparison of Obstacle Avoidance Path Planning Performance

| Indicator | Baseline (Traditional A*) | Ours (This articles algorithm) | Increase margin |
|---|---|---|---|
| Total length of path (m) | 32.5 | 34.2 | - |
| Minimum clearance distance (m) | 0.1 | 1.85 | +1750% |
| Maximum steering urgency | High (with a 90° bend) | Low (smooth transition) | Significantly improve |
| Planning time (ms) | 15.2 | 22.8 | - |

# 4. Summary

This paper proposes a local path planning method that combines safety potential field and kinematic constraints based on BEV fusion maps to address the obstacle avoidance problem of unstructured roads.

1). The environmental safety potential field was constructed through Euclidean distance transformation, which solved the collision hazard caused by the traditional A * algorithm path "edge sticking".

2). Improved the state space and cost function of A * search, introduced steering penalties and maximum angle constraints, ensuring the kinematic feasibility of the path.

3). Smooth trajectory processing was achieved using B-Spline.

The experimental results show that compared with traditional algorithms, the proposed method in this paper increases the minimum obstacle safety distance by more than 17 times while ensuring real-time performance, and effectively eliminates path angles, providing a safe and smooth local navigation trajectory for autonomous mining vehicles.

# Acknowledgements

Major Science and Technology Project of Guangxi under Grant No GK AA23062077.

# References

[1] Wu D, Meng Y, Zhang Y, et al. APG-DDQN: Expert-guided deep reinforcement learning for AUSV local path planning with modal switching in unstructured environments. Ocean Engineering. 2026, Vol. 352 (No. P2), p. 124632-124632.

[2] Katikaridis D, Benos L, Kateris D, et al. Proactive Path Planning Using Centralized UAV-UGV Coordination in Semi-Structured Agricultural Environments. Applied Sciences. 2026, Vol. 16 (No. 2), p. 1143-1143.

[3] Sazonov A, Kuchkin O, Cherepanska I, et al. S3PM: Entropy-Regularized Path Planning for Autonomous Mobile Robots in Dense 3D Point Clouds of Unstructured Environments. Sensors. 2026, Vol. 26 (No. 2), p. 731-731.

[4] Loulou A, Unel M. Hybrid attention-guided RRT*: Learning spatial sampling priors for accelerated path planning. Robotics and Autonomous Systems. 2026, Vol. 198, p. 105338-105338.

[5] Zhang J, Zhou M, Liu H, et al. Improved Two-Stage Theta* Algorithm for Path Planning with Uncertain Obstacles in Unstructured Rescuing Environments. Processes. 2026, Vol. 14 (No. 1), p. 167-167.

[6] Wang P, Yu H, Wang S. Vision-Controlled autonomous navigation in unstructured environments: Integrating image processing, path planning, and trajectory control in robotic systems. PloS one. 2026, Vol. 21 (No. 3), p. e0341589.

[7] Zhu T, Xu Z, Zhu R, et al. A Dual-Layer Hybrid-A* Path Planning Algorithm for Unstructured Environments Based on Phase Windows. Sensors. 2025, Vol. 26 (No. 1), p. 43-43.

[8] You B, She H, Li J, et al. Hazard-constrained global-local path planning for fault-tolerant hexapod robots on unstructured terrain. Mechatronics. 2026, Vol. 114, p. 103438-103438.