Journal of Engineering Mechanics and Machinery (2026)
Clausius Scientific Press, Canada

DOI: 10.23977/jemm.2026.110110
ISSN 2371-9133 Vol. 11 Num. 1

# *Hybrid 3D Coverage Path Planning for Precision Inspection of Large-scale Underwater Structures by AUVs*

**Yushen Duan[1,*], Yao Huang[1], Xiaolong Fan[1], Haoran Wang[1]**

[1]*School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai, China*

[*]*Corresponding Author: 1094818123@qq.com*

***Keywords:*** Autonomous Underwater Vehicle (AUV); 3D Coverage Path Planning; GBNN; Contour Scanning; 3D A* Avoidance

***Abstract:*** To address the challenges of balancing surface conformality and obstacle avoidance during Autonomous Underwater Vehicle (AUV) inspection of deep-water SPAR platforms, this paper proposes a hybrid 3D Coverage Path Planning (CPP) method fusing improved Glasius Bio-inspired Neural Network (GBNN) guidance with local contour repair. Adopting a "global guidance plus local fine-inspection" strategy, an improved static GBNN model efficiently establishes environmental topology to locate targets with low computational cost. Subsequently, a slicing-based contour generation algorithm ensures tight wall-following coverage. To handle unmodeled obstacles like anodes, a "detection-segmentation-repair" mechanism utilizing a 26-neighbor 3D A* algorithm constructs surgical avoidance paths within minimal topological space. Simulation results indicate that in obstacle-free environments, the hybrid method reduces path length by 84% and coverage time by 91% compared to single GBNN. In complex environments, efficiency improves by 8.6% over traditional slicing methods, with a path repetition rate of only 3.74%. The results confirm the algorithm's ability to autonomously select between "horizontal bypass" and "vertical surmounting" strategies based on obstacle geometry, effectively resolving path deadlocks and ensuring complete coverage.

## 1. Introduction

With the rapid advancement of marine engineering, there has been a surging demand for the operation and maintenance (O&M) of critical infrastructures, such as cross-sea bridges, offshore wind farms, and underwater dams. Autonomous Underwater Vehicles (AUVs), benefiting from their tether-free design and wide-area operational capabilities, have emerged as pivotal equipment for precision underwater inspection. However, unlike the two-dimensional scanning of open seabeds, the inspection of underwater columnar structures necessitates AUVs to achieve complete surface coverage within complex and constrained three-dimensional environments. According to a recent 2025 survey by Jayalakshmi et al. [1], balancing global coverage completeness, local collision avoidance safety, and real-time onboard computational efficiency within dynamic and unstructured environments remains a paramount challenge. Existing research on 3D Coverage Path Planning (3D CPP) can be primarily categorized into three streams: geometric slicing and dimensionality

reduction strategies, viewpoint planning and optimization, and bio-inspired grid-based algorithms. To mitigate the complexity inherent in high-dimensional planning, mainstream approaches frequently adopt the "slicing" methodology. The horizontal slicing strategy proposed by Galceran et al.[2] has been widely applied to underwater structure inspection. Similarly, Jung [3], Mansouri [4], and Biundini [5] have respectively validated the effectiveness of layered scanning in the contexts of UAV operations and point cloud planning. However, such methods are often predicated on ideal geometric assumptions. For instance, the 2D unfolding approach attempted by Song and Arshad[6] struggles to accommodate real-world structures characterized by irregular attachments, and traditional slicing methods inherently lack flexible mechanisms for local collision avoidance.

To enhance observation quality, a distinct category of research focuses on viewpoint sampling and optimization. Palomeras[7], Song[8], and Almadhoun[9] achieved high-quality reconstruction through offline planning or surface confidence guidance, while Jing[10], Cao[11], and Karapetyan[12] explored graph search, hierarchical frameworks, and deep learning strategies, respectively. Recently, Han[13] and Su[14] proposed novel optimization frameworks based on the improved Whale Optimization Algorithm (WOA) and WaveOptima, respectively. Although optimization-based methods offer high theoretical performance limits, their prohibitive computational costs for Next-Best-View (NBV) calculation and iterative convergence times often fail to meet the real-time response requirements of single AUVs in unknown environments[15].

In contrast, grid-based bio-inspired algorithms demonstrate significant advantages in computational efficiency and dynamic adaptability. Shen et al.[16][17] realized efficient terrain reconstruction using coverage trees, while Zhou[18] and Cai[19] verified the robustness of dynamic GBNN/BINN algorithms in polar under-ice and IoT scenarios, respectively. However, existing bio-inspired algorithms are predominantly used for global exploration. When confronting columnar structures with large vertical spans, they struggle to guide AUVs to perform precise equidistant scanning closely adhering to the surface.

In summary, existing methods face challenges in simultaneously satisfying the requirements of geometric wall-following accuracy and obstacle avoidance. Inspired by the "global coverage + local fine inspection" phased strategy of Song et al.[20], this paper proposes a hybrid path planning method based on improved GBNN global guidance and local contour scanning. This paper utilizes GBNN for efficient global exploration, switching to a layered contour scanning mode once a vertical target is identified. Addressing complex 3D protrusions within the sliced paths, a 26-neighbor based 3D A algorithm is introduced for local path repair. This method significantly reduces computational load and path redundancy while ensuring 100% full coverage of the structural surface.

## 2. Problem Formulation

This study focuses on a typical deep-water Single Column SPAR Platform. As illustrated in Figure.1, the main structure is a large-diameter vertical cylinder comprising a floating hull, mooring system, and underwater risers. Unlike jacket structures in shallow water (see Figure.2), SPAR platforms feature extreme drafts (up to 100 meters) with irregular surface protrusions such as anodes and J-tubes. To achieve precision 3D inspection, a voxel-based modeling approach is adopted for the underwater environment. Let the 3D workspace be $W \subset \mathbb{R}^3$, including the target structure surface $S_{target}$ and obstacle regions $O_{obs}$. The state attribute of each voxel $v_i$, denoted as $State(v_i)$, is defined as:

$$State(v_i) = \begin{cases} \text{Uncovered}, & \text{Target inspection area} \\ \text{Obstacle}, & \text{Obstacle area} \\ \text{Free}, & \text{Free workspace} \\ \text{Covered}, & \text{Inspected area} \end{cases} \tag{1}$$

For computational efficiency and obstacle avoidance safety, the cylindrical structure is modeled using an Axis-Aligned Bounding Box (AABB). This geometric approximation offers two primary advantages:

**(1) Conservative Avoidance:** The bounding box creates a natural safety buffer at the tangent planes of the cylinder, preventing AUV collisions with the physical entity.

**(2) Computational Simplification:** The regular cubic voxel structure aligns with the 26-neighbor neural network activity propagation, eliminating complex interpolation at curved boundaries and enhancing algorithmic robustness.

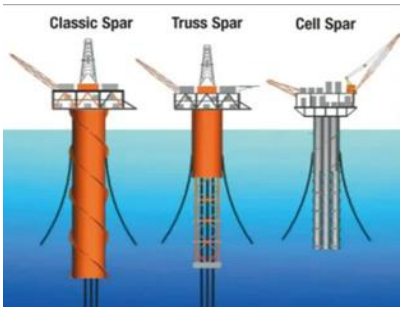



Figure 1. SPAR Platform Models

Figure 2. Comparison of offshore infrastructure inspection scenes: (left) Offshore wind turbine foundation; (right) Jacket-type structural platform. SPAR Platform Models

Based on this discrete model, the 3D trajectory of the AUV is represented as a voxel sequence $T = \{P_1, P_2, \ldots, P_k\}$. The path planning must satisfy the following core requirements:

**(1) Completeness Constraint:** The collective sensor field of view (FOV) must cover all target voxels marked as Uncovered, satisfying: $\Omega_{target} \subseteq \bigcup V(P_i)$.

**(2) Obstacle Constraint:** Path nodes must reside in free space, i.e., $State(P_i) = \text{Free}$, with strictly no penetration of Obstacle regions.

**(3) Cost Optimization:** Minimize the total path length $\sum \| P_{i+1} - P_i \|$ while satisfying the above constraints.

Solving the optimal path in such high-dimensional voxel spaces is inherently NP-hard, making

traditional geometric search methods computationally prohibitive for real-time applications. Consequently, the next chapter introduces the GBNN. By leveraging its "activity propagation" and "local suppression" characteristics, the complex geometric coverage problem is transformed into a gradient descent search of neuronal activity values, achieving efficient global guidance in complex underwater environments.

## 3. Three-dimensional Model Based on Improved GBNN Algorithm

This section details the underwater structure scanning algorithm based on the improved GBNN.

### 3.1. 2D Improved GBNN Algorithm Model

The traditional GBNN algorithm operates as a discrete-time Hopfield neural network. Its corresponding two-dimensional neural network model is illustrated in Figure.3, where each circle represents a single neuron. Each neuron maintains connections with its surrounding neighbors through either linear or diagonal links. Through these connections, neuronal activity values are propagated throughout the network. Furthermore, the dynamic characteristics of the activity values for every neuron within the network adhere to the laws described in Eq. (1).

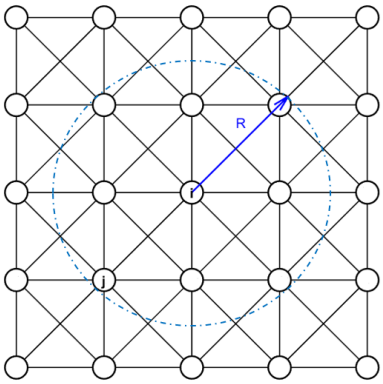



Figure 3. 2D Neural Network Model

However, this dynamic update mechanism presents two significant issues: (1) Dynamic iteration requires multiple rounds of updates, leading to prolonged computational time. (2) It necessitates storing intermediate states at each time step, thereby increasing memory overhead. To overcome these limitations, this paper proposes a simplified static model. By replacing dynamic updates with a one-time calculation, the need to store dynamic intermediate states is eliminated, significantly reducing computational complexity. To enhance computational efficiency and better adapt to discretized scenarios, this subsection introduces a novel neural network model:

$$x_i = f\left(I_i + \sum_{j=1}^{M} \omega_{ij}\right) \tag{2}$$

where $x_i$ denotes the activity value of the $i$-th neuron. The function $f$ is defined as follows:

$$f(z) = \begin{cases} -100, & z \le 0 \\ 0, & 0 < z < 100 \\ 100, & z \ge 100 \end{cases} \tag{3}$$

$I_i$ represents the external input to the $i$-th neuron, defined as:

$$I_i = \begin{cases} +E, & \text{Excitatory input} \\ -E, & \text{Inhibitory input} \\ 0, & \text{Zero input} \end{cases} \tag{4}$$

where $E$ is the external input value, which is a constant. $I_i > 0$ represents an unexplored area, while $I_i < 0$ indicates an impassable area (obstacle).

$\omega_{ij}$ denotes the connection weight coefficient between neuron $i$ and its laterally connected neuron $j$, expressed as:

$$\omega_{ij} = \begin{cases} \mu / |i-j|, & 0 < |i-j| < R \\ 0, & |i-j| \geq R \end{cases} \tag{5}$$

where $|i-j|$ is the Euclidean distance between the $i$-th neuron and the $j$-th neuron, and $\mu$ is a constant. $R$ represents the detection radius of the AUV.

In summary, compared with the traditional GBNN algorithm, the improved GBNN algorithm proposed in this paper simplifies the transfer function. This enhancement increases the neuron update speed and reduces the computational load.

### 3.2. AUV Path Selection Strategy

This study integrates a grid-based underwater environment map with a neural network, where neuronal activity values are dynamically updated in real-time according to the grid map state. Based on these activity values, the AUV selects an appropriate adjacent grid cell as its next path waypoint. However, constrained by onboard energy and mission duration, the AUV should prioritize straight-line navigation during coverage path planning to minimize heading changes, thereby reducing energy consumption. Consequently, the AUV's path selection strategy must simultaneously consider neuronal activity magnitude, heading, and the distance between the current and candidate points. Given the AUV's current position $P_c$, its next position $P_n$ is determined by Eq. (6):

$$P_n = \max(x_k + \lambda\theta_k + \alpha l), \quad k = 1,2,3,\ldots,m \leq 8 \tag{6}$$

where $P_n$ represents the next position obtained by the optimal selection strategy, and $x_k$ denotes the activity value of the $k$-th candidate neuron. $\lambda$ is the weight value for the AUV's heading angle. $\theta_k$ is the heading deviation function, defined as:

$$\theta_k = \left| \arctan\left( \frac{p_{ky} - p_{cy}}{p_{kx} - p_{cx}} \right) \right| \tag{7}$$

where $(p_{cx}, p_{cy})$ and $(p_{kx}, p_{ky})$ represent the coordinates of the current point and the candidate point $P_k$, respectively. $k$ is the number of selectable neurons surrounding the AUV. $\alpha$ is the weight value for the Euclidean distance between the AUV's current position and the candidate position, and $l$ is the Euclidean distance between the current and candidate positions:

$$l = \sqrt{(p_{kx} - p_{cx})^2 + (p_{ky} - p_{cy})^2} \tag{8}$$

### 3.3. Dead Zone Escape Strategy

During the 2D grid-based surface coverage process, the AUV may enter a "dead zone" where all surrounding selectable grid cells are either boundaries, obstacles, or previously covered areas, preventing the standard local selection strategy from advancing and consequently reducing coverage efficiency. To ensure the continuity and efficiency of 2D surface coverage, this paper adopts a dual-movement strategy: the next step is selected based on Eq. (6) within the 4-neighbor connectivity during the normal coverage phase, but upon detecting a dead zone, the system switches to 8-neighbor connectivity and executes an A*-based shortest path search to plan a route directly to the nearest uncovered grid cell, thereby quickly exiting the local enclosed area. The evaluation function for the A* dead zone escape is defined as:

$$t(k) = g(k) + h(k) \tag{9}$$

where $g(k)$ represents the cost from the current node to candidate node $k$, and $h(k)$ represents the heuristic cost from the candidate node to the target (the nearest uncovered node). Based on the method proposed in Reference [19], a rapid escape mechanism is formulated by constructing the cost function using connection weights as $g(k) = -\omega_{ij}$ and the heuristic term using Euclidean distance as $h(k) = \| X_i - X_j \|$. This approach generates a bridging path to "escape the dead zone" on the discrete grid with minimal computational load, where the next node is iteratively selected according to $P_n = \arg\min_k (g(k) + h(k))$ until the AUV reaches the target uncovered area.

### 3.4. 3D Improved GBNN Algorithm Model

To address the complete coverage inspection problem for AUVs in complex 3D underwater environments, this section extends the 2D algorithm model into three-dimensional space. As illustrated in Fig. 4, neurons are arranged in a 3D spatial distribution, forming a fully connected neural network structure. Each neuron maintains connection relationships with up to 26 surrounding neighbors, comprising both linear connections along the coordinate axes and spatial diagonal connections.

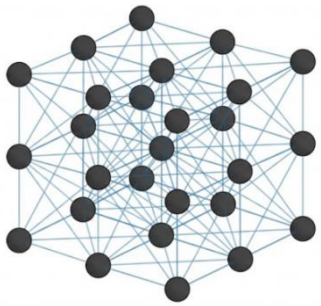


Figure 4. 3D Neural Network Model

The 2D grid map is further extended into a 3D voxel map. Let the underwater workspace be denoted as $W$, which is discretized into a set of equal-sized cubic voxels:

$$W = \{v_p \mid v_p = (x_p, y_p, z_p), p = 1, 2, \ldots, N\} \tag{10}$$

Each voxel corresponds to a single neuron in the neural network. Unlike the 8-neighbor connectivity in the 2D model, the neighbor set of neuron $i$ in the 3D model, denoted as $\mathcal{N}_i^{3D}$, consists of at most 26 neurons. In the 3D neural network, the update rule for the activity output of

each neuron is formulated as:

$$x_i = F\left(\sum_{j\in\mathcal{N}_i^{3D}} W_{ij} + I_i\right) \tag{11}$$

where $I_i$ is the external input: the target to be inspected is set to $E$, and obstacles are set to $-E$. The transfer function $F(\cdot)$ is consistent with $f(\cdot)$ used in Eq. (3), serving to limit the activity range and embody the characteristic of "positive activity propagation and negative activity local suppression".

In the 3D voxel network, the connection weight between neurons is redefined as:

$$W_{ij} = \begin{cases} \dfrac{\mu}{d_{ij}}, & 0 < d_{ij} \le R \\ 0, & d_{ij} > R \end{cases} \tag{12}$$

where $\mu$ is a constant, $R$ is the activity propagation radius (typically chosen to ensure diagonal neighbors are included within the propagation range), and $d_{ij}$ is the 3D Euclidean distance:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \tag{13}$$

Through the aforementioned 3D extension, the neural network activity field creates a global attraction gradient towards uncovered regions within the voxel space and forms local inhibition zones near obstacle voxels. This mechanism provides a unified state representation of "traversable and coverable" areas and offers directional guidance for subsequent 3D coverage planning.

## 4. Hybrid Contour-Layer Coverage Method Integrated with A* Algorithm

Although the 3D GBNN model effectively guides AUV exploration in three-dimensional environments, it often struggles to guarantee uniform, equidistant, and comprehensive inspection when identifying large-scale vertical structures (e.g., bridge piers, subsea pipelines, or reefs). To address this, a hybrid path planning strategy is proposed: utilizing the improved GBNN for surface exploration and switching to a "slice-based contour scanning" mode upon specific target identification. Inspired by the structural inspection framework of Galceran et al.[2], this mode incorporates a heuristic-based path repair mechanism specifically designed for voxel environments to ensure deterministic 3D obstacle avoidance.

### 4.1. Contour Path Generation Based on Slicing Strategy

Adopting the principle of "dimensionality reduction" (see Algorithm 1), the complex 3D coverage problem is decomposed into a sequence of 2D planar planning tasks by discretizing the structure along the depth axis. The specific procedure is as follows:

**(1) Layer Slicing:** Let the height range of the 3D obstacle $O \subset \mathbb{R}^3$ be $[z_{min}, z_{max}]$. The coverage space is partitioned along the Z-axis with an interval $\Delta z$, generating a series of horizontal task planes $\mathcal{L} = \{L_1, L_2, \ldots, L_k\}$.

**(2) Contour Generation:** On each plane $L_i$, a closed-loop path $P_{contour}^{(i)}$ is generated based on the obstacle contour at a constant distance $d_{safe}$ from the surface. This strategy ensures the AUV sensors maintain an optimal observation distance.

**(3) Inter-layer Connection:** To prevent abrupt AUV maneuvers during layer transitions, adjacent paths $P^{(i)}$ and $P^{(i+1)}$ are connected via spiral interpolation, forming a continuous 3D nominal scanning path $\mathcal{T}_{nominal}$.

### 4.2. Path Repair and Obstacle Avoidance Based on A* Algorithm

The "nominal path" $\mathcal{T}_{nominal}$ generated by the geometric method is predicated on an ideal model. In actual voxel grid maps, this path may intersect with unmodeled protrusions or enter dead zones. Unlike the randomized trajectory optimization employed by Galceran et al. to handle path deformation, this paper proposes a deterministic and computationally efficient "Detect-Segment-Repair" mechanism for discrete grid environments (see Algorithm 2). The core of this mechanism involves local path reconstruction using the A* algorithm:

**(1) Collision Detection and Segmentation:** The nominal path is mapped into the 3D voxel map $M$. Nodes overlapping with obstacles (marked as $-2E$) are identified. Consequently, the path is severed by these collision nodes into multiple safe segments $\{S_1, S_2, \ldots\}$.

**(2) Local Replanning (26-Neighborhood):** For any two disconnected safe segments $S_j$ and $S_{j+1}$, a collision-free bridging path connecting $S_j^{end}$ and $S_{j+1}^{start}$ is required. To accommodate the six-degree-of-freedom motion characteristics of the AUV, a 3D A* searcher based on 26-neighbor connectivity is constructed.

**(3) Neighborhood Expansion:** The current node $n(x, y, z)$ expands toward 26 surrounding directions (including vertex diagonals). This ensures the repaired path is not restricted to Manhattan distance, resulting in smoother trajectories.

**Cost Function:** The cost function is defined as:

$$f(n) = g(n) + h(n) \tag{14}$$

where $g(n)$ represents the actual path cost, and $h(n)$ utilizes the 3D Euclidean distance heuristic function:

$$h(n) = \sqrt{(x_{goal} - x_n)^2 + (y_{goal} - y_n)^2 + (z_{goal} - z_n)^2} \tag{15}$$

Path Fusion: The local avoidance path generated by A* is embedded into the break points of the original path, yielding a final feasible trajectory $\mathcal{T}_{final}$ that satisfies coverage requirements while strictly avoiding 3D obstacles.

**Algorithm 1 Hybrid Planar Coverage with Triggered Contour-Layer Scanning**

**Input:** $(P_0, M, \{B_i\}, r_{\text{trig}}, \Delta z, d_{\text{safe}}, d_s, N_{\text{ramp}})$;

**Output:** $(T)$

**1.** $(P \leftarrow P_0; T \leftarrow [P_0]; mode \leftarrow NORMAL; scanned[i] \leftarrow 0)$

**2.** $(\mathcal{C} \leftarrow \varnothing)$

**3.** *while* $(\neg Terminate(P, M, scanned))do$

**4.** *if* $(mode = NORMAL)then$ $(P \leftarrow GBNN\_2D\_Step(P, M); T \leftarrow T \cup [P])$

**5.** *if* $(\exists i : scanned[i] = 0 \wedge NearBBox(P, B_i, r_{\text{trig}}, M))then$

**6.** $(\mathcal{C} \leftarrow BuildAndRepairContourPath(B_i, P, M, \Delta z, d_{\text{safe}}, d_s, N_{\text{ramp}}))$

**7.** *if* $(\mathcal{C} \neq \varnothing)then$ $(mode \leftarrow SCAN; scanned[i] \leftarrow 1)endif$

**8.** *end if*

**9.** *else* $(\triangleright\ SCAN)$ : *if* $(\mathcal{C} \neq \varnothing)$*then* $(P \leftarrow PopFront(\mathcal{C}); T \leftarrow T \cup [P])$

**10.** *else* $(mode \leftarrow NORMAL)$*endif*

**11.** *end while*; *return* $(T)$

**Algorithm 2 Contour-Layer Path Construction and 3D A* Repair**

**Input:** *box* $(B)$, *pose* $(P)$, *map* $(M)$, $(\Delta z, d_{\text{safe}}, d_s, N_{\text{ramp}})$;

**Output:** $26-$*neighbor path* $(C)$

**1.** $for(z = z_{\max})downto(z_{\min})step(\Delta z) : \mathcal{P} \cup = \{OffsetContourRect(B, z, d_{\text{safe}}, d_s)\}$

**2.** $(C \leftarrow\ StitchSmooth(\mathcal{P}, N_{\text{ramp}}))$

**3.** $(C \leftarrow\ RasterizeGrid26(C))$

**4.** $(C \leftarrow\ RotateToNearestStart(C, P))$

**5.** $(C \leftarrow\ ConnectGrid26(P, C[1]) \oplus C)$

**6.** $(\Omega \leftarrow \{v \in C |\ M(v) = -2E\})(collisionnodes)$

**7.** $(\{S_j\}_{j=1}^{n} \leftarrow\ SplitByCollisions(C, \Omega))$; $(C^{\star} \leftarrow S_1)$

**8.** $for\ (j = 1)to\ (n-1)$ : $(s \leftarrow End(S_j)),\ (g \leftarrow Start(S_{j+1}))$

**9.** $(A \leftarrow\ AStar26(s, g, M; obstacle = -2E, cost = 1, h = \|\cdot\|_2))$

**10.** $(C^{\star} \leftarrow C^{\star} \oplus A \oplus S_{j+1})(dropduplicateendpoints)$

**11.** *return* $(RasterizeGrid26(C^{\star}))$

## 5. Simulation and Analysis

This section aims to validate the effectiveness and robustness of the proposed "hybrid path planning algorithm based on GBNN global guidance and local contour repair" in multi-target underwater structure inspection tasks through simulation. The experiments are conducted in two phases: first, the multi-target full coverage capability and search efficiency are verified in an ideal obstacle-free environment; subsequently, unmodeled complex sudden obstacles are introduced on the structural surface to specifically evaluate the algorithm's adaptive obstacle avoidance mechanism and path topology preservation capability.

### 5.1. Simulation Environment and Parameter Settings

The simulation experiments are established on the MATLAB platform. To simulate authentic underwater operational scenarios, the workspace $\mathcal{W}$ is discretized into a $25 \times 25 \times 25$ three-dimensional voxel grid map. As illustrated in **Figure. 5**, the simulation scene contains two deep-water Single Column SPAR platforms to be inspected. According to the environmental model established in Section 2, the main structure of the SPAR platform is approximated as an Axis-Aligned Bounding Box (AABB), represented by yellow voxel regions, while the remaining regions are defined as free water space. To verify the comprehensive performance of the proposed algorithm, comparative experiments are conducted in both an obstacle-free benchmark environment and a complex obstacle environment. By recording key metrics such as coverage rate, total path length, coverage time, and path repetition rate, the algorithm's performance is quantitatively evaluated in terms of "global exploration efficiency" and "local obstacle avoidance optimization".

The initial deployment position of the AUV is designated at the edge node $(1,1,25)$ of the water surface layer (indicated by the green node in **Figure. 5**). The core parameters of the proposed algorithm are configured as follows: $E=100$, $R=2$, $\lambda=0.03$, $\alpha=-0.06$, and $\mu=0.4$. Specifically, the parameters within the GBNN algorithm are set as $\beta=0.5$ and $\alpha=3$; the external excitation input is $E=100$; the lateral connection region radius is $R=\sqrt{3}$; and the parameter for the path selection strategy is set to $c=0.5$.

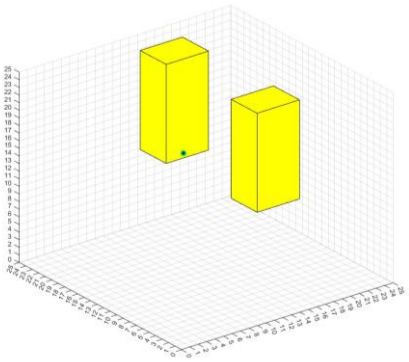


Figure 5. Initialization view of the multi-target SPAR platform simulation environment.

## 5.2. Full Coverage Performance Verification in Benchmark Scenarios

Upon initiation of the simulation, the AUV first executes a 2D coverage exploration task based on the improved GBNN at the water surface layer ($Z=25$). As illustrated in **Figure. 6**, guided by the neuronal activity gradient, the AUV generates a scanning path (indicated by the blue trajectory) that completely traverses the water surface layer. The generated path autonomously circumvents regions occupied by the yellow target structures, thereby effectively preventing collisions. The output of this phase not only accomplishes the surface inspection task but also provides precise entry point coordinates for the subsequent 3D contour-layer scanning.

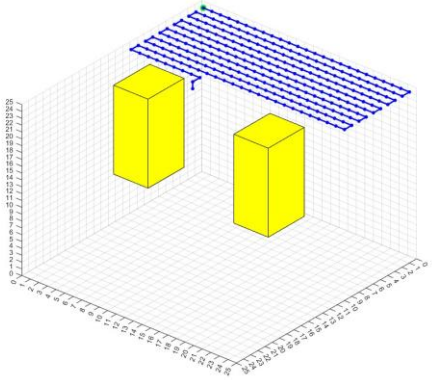


Figure 6. Global search path at the water surface layer and target localization results.

Upon localizing the target position, the AUV transitions to the contour-layer scanning mode, performing sequential spiral coverage of the targets along the depth axis (Z-axis). This process is illustrated in **Figure. 7**. Initially, the AUV enters from the top of the left target and executes a spiral descent to the bottom along a predefined rectangular contour trajectory (**Figure. 7a**); the path adheres closely to the target surface, ensuring comprehensive coverage of the vertical facades. Following the completion of the first target's inspection (**Figure. 7b**), the AUV utilizes the guidance

of the global neural activity map to autonomously plan a collision-free transition path to the subsequent target (**Figure. 7c**) and replicates the aforementioned layered scanning process (**Figure. 7d**).

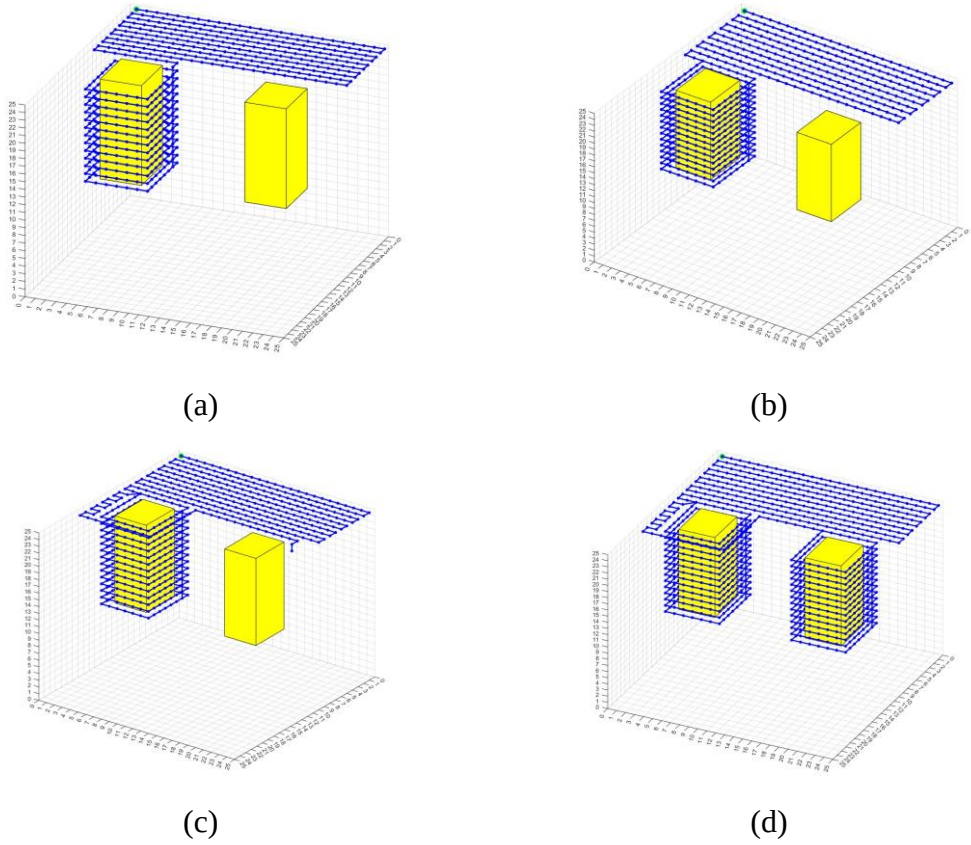
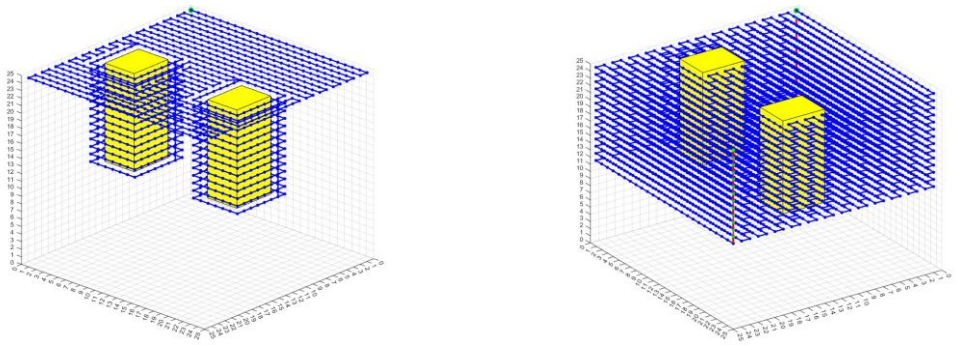


Figure 7. Evolution of the multi-target contour-layer scanning process: (a) Spiral inspection of Target 1; (b) Completion of Target 1 and inter-target transition; (c) Initiation and execution of scanning on Target 2; (d) Completion of the global multi-target coverage task.

As illustrated by the final global trajectory in **Figure 8a**, the discrete blue path points form two complete bounding cages surrounding the targets. The algorithm successfully achieves **100% full coverage** of both the dual SPAR platforms and the water surface region. Furthermore, the transitions between multi-stage tasks (search-scan-transition-scan) are smooth and continuous, verifying the **completeness** of the proposed hybrid strategy in an ideal environment. In contrast, as shown in **Figure 8b**, the traditional GBNN algorithm demonstrates strong global traversal capabilities within the benchmark dual-SPAR scenario. It exhibits a broad trajectory coverage range and good overall connectivity, enabling systematic visitation of large-scale workspaces. This approach is well-suited for task paradigms requiring the "traversal of the entire 3D volumetric region." However, the coverage trajectory generated by GBNN tends to prioritize voxel traversal within the workspace. Consequently, when applied to tasks requiring compact, close-proximity scanning of cylindrical surfaces, it results in **high spatial redundancy**.

In the obstacle-free benchmark scenario of the dual SPAR platforms, as detailed in **Table 1**, although the GBNN algorithm exhibits strong global traversal capabilities, its trajectory suffers from excessive spatial redundancy. In comparison, the hybrid algorithm proposed in this paper demonstrates significant performance improvements while maintaining a **100% coverage rate**.



(a) (b)

Figure 8. Comparison of global coverage trajectories in the benchmark environment: (a) The proposed hybrid path planning algorithm; (b) The traditional GBNN algorithm.

Table 1 Performance comparison in the benchmark environment

| Method | Coverage Rate | Total Path Length | Coverage Time | Path Repetition Rate |
|---|---|---|---|---|
| Proposed Algorithm | 100% | 1274m | 47.12s | 2.6% |
| GBNN Algorithm | 100% | 7936m | 535.08s | 0 |

### 5.3. Verification of Adaptive Obstacle Avoidance in Complex Environments

To further evaluate the robustness of the algorithm in unstructured environments, two obstacles (indicated by the red regions in **Figure 9**) were introduced on the surface of one target. These obstacles simulate isolated anode blocks and complex continuous marine biofouling, respectively, creating physical obstructions along the standard scanning path.

When the AUV detects that the forward path is blocked, the system activates the *3D A local replanning module** based on 26-neighbor connectivity. Simulation results demonstrate that the algorithm possesses the capability to autonomously select the optimal obstacle avoidance strategy according to the geometric dimensional characteristics of the obstacles.

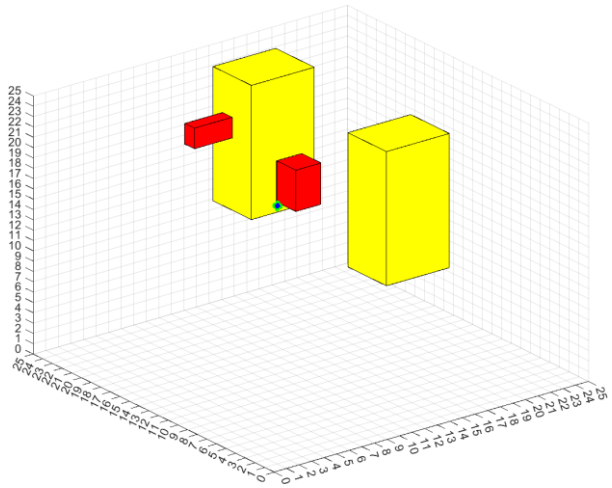


Figure 9. Simulation test scenario containing unstructured protruding obstacles.

Specifically, for obstacles characterized by a wide horizontal extension but limited vertical thickness (as shown in Figure 10), relying solely on horizontal bypassing would introduce excessive path redundancy. Therefore, the algorithm leverages the freedom of 3D space and the advantage of

26-neighbor diagonal search to plan a composite path comprising horizontal and oblique motions.

Conversely, when confronting obstacles with a large vertical span but a small horizontal cross-section (as shown in Figure 11), the algorithm determines that the cost of vertical crossing is excessive. Consequently, it maintains the current depth layer and plans a "U-shaped" lateral bypass trajectory within the horizontal plane, thereby maximizing the preservation of slice scanning continuity.

This adaptive decision-making mechanism, based on the aspect ratio characteristics of obstacles, achieves an intelligent fusion of "horizontal lateral bypass" and "vertical spatial crossing," effectively avoiding path deadlocks on the surfaces of complex structures.

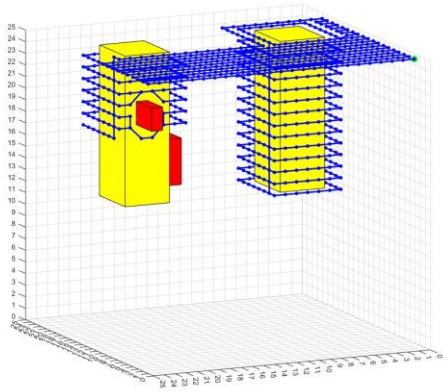


Figure 10. View of the obstacle avoidance trajectory based on the "vertical spatial crossing" strategy.

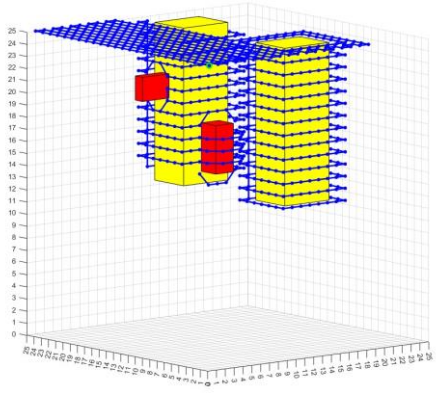


Figure 11. Integrated application of "horizontal lateral bypass" and "vertical spatial crossing" strategies.

**Figure 12** illustrates the final 3D coverage path generated by the proposed algorithm in a complex obstacle environment. Although the tubular path undergoes multiple local deformations, the blue scanning trajectory remains tightly wrapped around all target structures. Notably, after circumventing the obstacles, the AUV is able to rapidly and smoothly realign with the preset contour **scanning track.** This demonstrates the stability of the dual-layer "Global Guidance + Local Repair" architecture, effectively preventing global path divergence caused by local obstacle avoidance maneuvers.

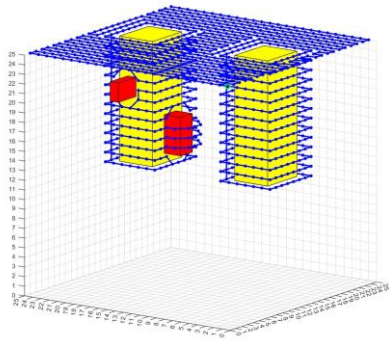


Figure 12. Final 3D coverage trajectory generated by the proposed algorithm in a complex obstacle environment.

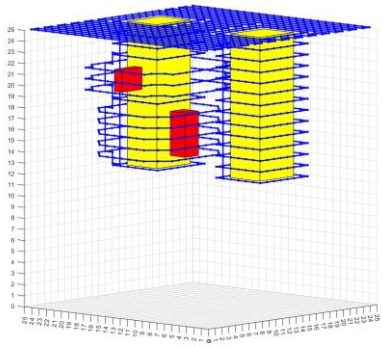


Figure 13. Path deformation and redundancy exhibited by the traditional layered planning method when encountering obstacles.

While both the traditional contour-layer coverage and the proposed method aim to establish a regularized structural sweeping mode, their adaptability differs significantly when confronting sudden unstructured obstacles.

As shown in **Figure 13**, when the traditional layered planning encounters surface protrusions on the cylinder, it often employs passive avoidance mechanisms based on potential fields or random sampling. This "holistic path deformation" strategy causes the scanning trajectory to struggle to return quickly to the nominal scan line after avoidance. This not only disrupts the original equidistant observation geometry but also introduces a significant amount of uncontrollable redundant travel and repetitive coverage, leading to a drastic decline in operational efficiency as obstacle complexity increases.

Table 2 Performance comparison in the complex obstacle

| Method | Coverage Rate | Total Path Length | Coverage Time | Path Repetition Rate |
|---|---|---|---|---|
| Proposed Algorithm | 100% | 1282m | 45.6s | 3.74% |
| Layered Algorithm | 100% | 1376m | 49.9s | 2.64% |

In contrast, the "Detect-Segment-Repair" framework proposed in this paper possesses distinct topological preservation characteristics. Treating obstacles as local disturbances, the algorithm utilizes 3D A* to construct a "surgical" bridging path within the minimal topological space adjacent to the obstacle. This ensures that the AUV can immediately and smoothly "lock" back onto the preset contour track after bypassing the obstacle. This deterministic local repair mechanism maximizes the suppression of disruption to the global scanning mode while ensuring 100% coverage completeness,

achieving a dual optimization of path cost and observation quality.

The results in Table 2 indicate that even under obstacle interference, the proposed algorithm maintains superior operational efficiency. Compared to the traditional algorithm, the proposed method reduces the total path length by approximately 6.8% and saves approximately 8.6% in operation time. This confirms that the hybrid path planning method not only inherits the high efficiency of contour scanning but also overcomes the rigidity and redundancy of the traditional slicing method in complex environments through its active obstacle avoidance mechanism, realizing a dual optimization of path cost and observation quality.

## 6. Conclusion and Future Work

To address the path planning challenges for large-scale underwater structures, this paper proposes a hybrid framework integrating improved GBNN global guidance with local contour scanning. By combining bio-inspired localization and deterministic 3D slicing, the method effectively mitigates path redundancy. Furthermore, a topological-preserving avoidance mechanism based on 26-neighbor 3D A* is introduced to ensure rapid trajectory realignment. Simulation results demonstrate that the proposed strategy guarantees **100% coverage completeness** while reducing path length by **84%** and time by **91%** compared to the standalone GBNN. In complex environments, it improves operational efficiency by **8.6%** over traditional layered strategies. Future work will focus on multi-AUV cooperative coverage and dynamic obstacle avoidance based on receding horizon optimization.

## References


*[1] K. P. Jayalakshmi, V. G. Nair, and D. Sathish, "A Comprehensive Survey on Coverage Path Planning for Mobile Robots in Dynamic Environments," IEEE Access, vol. 13, pp. 60185-60212, 2025.*

*[2] E. Galceran, R. Campos, N. Palomeras, M. Carreras, and P. Ridao, "Coverage path planning with realtime replanning for inspection of 3D underwater structures," in 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 2014, pp. 6586-6591.*

*[3] S. Jung, S. Song, P. Youn, and H. Myung, "Multi-layer Coverage Path Planner for Autonomous Structural Inspection of High-rise Structures," in IEEE/RSJ IROS, 2018.*

*[4] S. S. Mansouri, C. Kanellakis, E. Fresk, D. Kominiak, and G. Nikolakopoulos, "Cooperative coverage path planning for visual inspection," Control Engineering Practice, vol. 74, pp. 118-131, 2018.*

*[5] I. Z. Biundini, M. F. Pinto, A. G. Melo, et al., "A Framework for Coverage Path Planning Optimization Based on Point Cloud for Structural Inspection," Sensors, vol. 21, no. 2, p. 570, 2021.*

*[6] Y. S. Song and M. R. Arshad, "Path planning for an autonomous underwater vehicle in pole inspection," Indian Journal of Geo-Marine Sciences, vol. 47, no. 12, pp. 2477-2484, 2018.*

*[7] N. Palomeras, N. Hurtós, M. Carreras, and P. Ridao, "Autonomous Mapping of Underwater 3-D Structures: From View Planning To Execution," IEEE Robotics and Automation Letters, vol. 3, no. 3, pp. 1965-1972, 2018.*

*[8] S. Song and S. Jo, "Surface-based Exploration for Autonomous 3D Modeling," in IEEE ICRA, 2018.*

*[9] R. Almadhoun, T. Taha, D. Gan, et al., "Coverage Path Planning with Adaptive Viewpoint Sampling to Construct 3D Models of Complex Structures," in IEEE/RSJ IROS, 2018.*

*[10] W. Jing, D. Deng, Z. Xiao, et al., "Coverage Path Planning using Path Primitive Sampling and Primitive Coverage Graph for Visual Inspection," in IEEE/RSJ IROS, 2019.*

*[11] C. Cao, J. Zhang, M. Travers, and H. Choset, "Hierarchical Coverage Path Planning in Complex 3D Environments," in IEEE ICRA, 2020.*

*[12] N. Karapetyan, J. V. Johnson, and I. Rekleitis, "Coverage Path Planning for Mapping of Underwater Structures," in IEEE OCEANS, 2020.*

*[13] G. Han, Y. Hou, W. Lai, et al., "A Multi-Dimensional Optimization Framework for AUV Cooperative Coverage Path Planning in Dynamic Underwater Environments," IEEE Transactions on Vehicular Technology, vol. 74, no. 11, 2025.*

*[14] C. Su, Y. Li, H. Wang, and G. Wan, "3D Coverage Path Planning for Underwater Dam Inspection Based on WaveOptima," Information and Control, vol. 53, 2024.*

*[15] Z. Shang, J. Bradley, and Z. Shen, "A co-optimal coverage path planning method for aerial scanning of complex structures," Expert Systems with Applications, vol. 158, p. 113535, 2020.*

*[16] Z. Shen, "3-D Coverage Path Planning for Underwater Terrain Mapping," Master's thesis, University of Connecticut, 2017.*
*[17] Z. Shen, J. Song, K. Mittal, and S. Gupta, "CT-CPP: Coverage Path Planning for 3D Terrain Reconstruction Using Dynamic Coverage Trees," IEEE Robotics and Automation Letters, vol. 7, no. 1, pp. 135-142, 2022.*
*[18] L. Zhou, Z. Deng, G. Bai, et al., "A real time data-driven dynamic glasius bionic neural network path planning algorithm for polar under-ice feature scanning," Ocean Engineering, vol. 312, p. 119289, 2024.*
*[19] W. Cai, S. Zhang, M. Zhang, and C. Wang, "Improved BINN-Based Underwater Topography Scanning Coverage Path Planning for AUV in Internet of Underwater Things," IEEE Internet of Things Journal, vol. 10, no. 20, pp. 18375-18386, 2023.*
*[20] S. Song, D. Kim, and S. Jo, "Online coverage and inspection planning for 3D modeling," Autonomous Robots, vol. 44, pp. 1431-1450, 2020.*