Research Square

Preprints are preliminary reports that have not undergone peer review. They should not be considered conclusive, used to inform clinical practice, or referenced by the media as validated information.

An Autonomous Spraying Robot for Sheep Houses: Hierarchical Navigation, Speed-Dependent Flow Control, and Multi-Stage Evaluation

Shuaizhi Wang
Henan Agricultural University
Xiaojuan Liu
Zhengzhou Normal University
Zequan Huang
Henan Agricultural University
Tong Han
Henan Agricultural University
Zhiyu Li
Henan Agricultural University
Zezhong Feng
Henan Agricultural University
Hanqing Gu
Henan Agricultural University
Zhengyang Wu
Henan Agricultural University
Pan Ding
dingpan@henau.edu.cn

Henan Agricultural University

Article

Keywords: sheep-house robot, autonomous navigation, simultaneous localization and mapping, dynamic obstacle avoidance, coverage planning, spray-command control

Posted Date: October 19th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10889918/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

An Autonomous Spraying Robot for Sheep Houses: Hierarchical Navigation, Speed-Dependent Flow Control, and Multi-Stage Evaluation

Shuaizhi Wang[a,b,1], Xiaojuan Liu[c,1], Zequan Huang[a,b,1], Tong Han[a,b], Zhiyu Li[a,b], Zezhong Feng[a,b], Hanqing Gu[a,b], Zhengyang Wu[a,b], Pan Ding[a,b,]*

a Henan Agricultural University, Zhengzhou 450002, China

b Zhengzhou Key Laboratory of Intelligent Technology and System of Agricultural Equipment, Zhengzhou 450002, China

c Zhengzhou Normal University, Zhengzhou 450044, China

1 These authors contributed equally.
* Corresponding author: Pan Ding. Email: dingpan@henau.edu.cn

Abstract

Autonomous spraying in sheep houses is constrained by narrow aisles, repetitive pen geometry, dust-related scan loss, uneven floors, track slip, and temporary blockage by animals. This study developed a tracked mobile spraying platform and a ROS-based three-level navigation framework that coordinates coverage-task generation, B-spline-smoothed A* global routing, DWA local avoidance, and speed-dependent pump commands. In 15 repeated simulations per configuration, the selected GMapping configuration achieved a map intersection-over-union of 96.8 ± 0.8% and reduced mapping time to 738 ± 29 s; ten AMCL trials yielded a localization RMSE of 0.0299 ± 0.0026 m. In physical mock-up tests, the bio-inspired strategy achieved 99.8 ± 0.2% geometric coverage, 4.2 ± 0.2% repeated coverage, and a duration of 78.36 ± 1.02 s across five runs. Fifteen clean-water runs at 0.08, 0.11, and 0.14 m/s produced mean deposition rates of 0.1985, 0.2008, and 0.2006 L/m², respectively, with transverse coefficients of variation of 5.37%, 5.20%, and 5.30%. Actual flow increased with speed (Bonferroni-adjusted $p < 0.001$), whereas deposition, effective swath width, and transverse uniformity did not differ among speeds (adjusted $p \geq 0.980$). Across five supervised field cycles, path-tracking RMSE was 0.0376 ± 0.0031 m, localization continuity was 99.62 ± 0.23%, and 55 of 56 animal-related blockage events were handled automatically. The results support the engineering feasibility of the integrated navigation and hydraulic-delivery workflow under the reported conditions; microbiological efficacy and animal-welfare effects were not evaluated.

Keywords: sheep-house robot; autonomous navigation; simultaneous localization and mapping; dynamic obstacle avoidance; coverage planning; spray-command control

Science4Impact Statement

This work contributes a sheep-house-specific engineering solution rather than a generic ROS demonstration. The system couples a tracked and protected hardware platform with scan-quality gating, resource-aware localization, explicit task–global–local navigation interfaces, blockage recovery, supervised stopping, and speed-dependent liquid delivery. Repeated mapping, localization, coverage, structured-obstacle, clean-water, and field-cycle records quantify the value and operating limits of these adaptations. The resulting architecture and dataset provide a reproducible engineering reference for low-speed spraying robots in constrained livestock facilities.

Abbreviations

Abbreviation	Full English Name
ROS	Robot Operating System
SLAM	Simultaneous Localization and Mapping
AMCL	Adaptive Monte Carlo Localization
GMapping	Grid-based Rao-Blackwellized particle-filter SLAM implementation
A*	A* graph-search algorithm
DWA	Dynamic Window Approach
IoU	Intersection over Union
IMU	Inertial Measurement Unit
LiDAR	Light Detection and Ranging
SSH	Secure Shell

1 Introduction

Intensive livestock production increasingly uses sensing, automation, and robotics to reduce labor requirements and improve operational consistency. Epidemic-prevention spraying is repetitive and may expose workers to disinfectants and contaminated environments; a mobile spraying platform can reduce close-range manual operation while providing more repeatable routes and command sequences (Department and Wang 2022; Feng and Wang 2020). Practical deployment,

however, depends on reliable mapping, localization, obstacle avoidance, and task execution under the spatial and environmental constraints of livestock facilities.

Sheep houses present several navigation challenges, including narrow aisles, repetitive pen geometry, uneven or slatted floors, dust, partial sensor occlusion, and moving animals. These conditions can degrade LiDAR measurements, increase odometric drift, temporarily obstruct planned routes, and interrupt spraying tasks. Manual spraying can also produce variable geometric coverage because walking speed and nozzle handling differ among operators. A practical robotic system must therefore coordinate perception, hierarchical navigation, platform motion, and spraying commands while clearly defining the scope and limitations of its field evidence.

1.1 Related Work and Research Gap

Livestock-house disinfection robots have mainly used chemical spraying, ultraviolet irradiation, or combined treatment methods. Feng et al. developed spraying robots using magnetic strips and RFID guidance to improve operational stability in low-light animal-house environments (Feng and Wang 2020; Qing-chun Feng et al. 2021; Qingchun Feng et al. 2021). UVC-based systems and combined UVC–aerosol systems have also been investigated for targeted disinfection tasks (Ma et al. 2022; Yao et al. 2023). These studies demonstrate the feasibility of robotic disinfection, but their navigation assumptions, target surfaces, and validation methods differ substantially from those required in narrow sheep-house aisles with live animals.

Research on autonomous navigation has addressed SLAM robustness, global path search, and local obstacle avoidance. Feature-fusion SLAM methods can improve pose estimation in complex indoor scenes, while improved RRT*, A*, and hybrid planning methods can reduce search time or improve path smoothness (Chang et al. 2023; Wang et al. 2024; Chen et al. 2023). In agricultural field robotics, technical contribution also arises from converting these capabilities into a verifiable system under scene-specific sensing, mobility, safety, and task constraints. Accordingly, this study focuses on the co-design of scan-quality gating, resource-aware localization, tracked motion, hierarchical replanning, and spray-command coordination for sheep-house operation.

Three gaps motivate this study. First, few studies describe how mapping, localization, global-local navigation, coverage planning, and spraying control are coordinated on a single sheep-house platform. Second, dynamic disturbances are often reported without defined obstacle trajectories, speeds, frequencies, or repeated-trial protocols. Third, geometric path coverage is sometimes treated as equivalent to disinfection effectiveness, although trajectory coverage, liquid deposition, and microbiological reduction represent different levels of evidence. This study addresses system-level integration and navigation-oriented evaluation while reserving disinfection efficacy for future experimental verification.

1.2 Objectives and Main Contributions

This study develops a tracked spraying robot and a ROS-based software framework for sheep-house operation. The objectives were to integrate LiDAR, IMU, wheel odometry, differential tracked drive, and spraying hardware; implement a mapping-to-navigation workflow based on GMapping and AMCL; coordinate task-level coverage planning, B-spline-smoothed A* global planning, and DWA local motion planning; link robot speed to a nominal pump command; and evaluate mapping, localization, path tracking, obstacle avoidance, geometric path coverage, and supervised field operability within clearly defined evidence limits.

The study makes three scenario-specific engineering contributions. First, it combines a compact tracked and dust-protected platform with LiDAR valid-return gating, locally limited scan use, sensor-watchdog stopping, and map-based pose recovery for narrow and partially occluded sheep-house aisles. Second, it defines explicit interfaces among coverage-cell selection, B-spline-smoothed A* routing, DWA avoidance, route recovery, global replanning, and supervised stopping, thereby turning separate navigation modules into a continuous livestock-house task workflow. Third, it couples ground speed to a saturation-aware flow rule and evaluates that coordination using three-speed clean-water measurements, repeated geometric coverage, structured obstacle events, and five quantified occupied-house work cycles. The contribution is the measurable scenario adaptation and coordination of the complete system.

Table 1. Sheep-house-specific engineering adaptations and evidence links.

Narrow aisles and repetitive pen geometry	Separated mapping/localization workflow and three-level task–global–local navigation	Saved occupancy map, AMCL pose, B-spline-smoothed A*, DWA, 0.20 m clearance criterion	15-run mapping comparison; 10 AMCL trials; 80 planning trials/configuration
Dust, occlusion, and invalid LiDAR returns	Scan-confidence gating and locally limited sensing range	Valid-return-ratio gate; 0.05–3.0 m usable range; sensor-loss stop logic	Noise/invalid-return simulation and mapping-configuration comparison
Uneven or slatted floor and track slip	Tracked chassis with map-based pose correction and conservative low-speed operation	Encoder–IMU–LiDAR workflow; AMCL recovery after pose disturbance	Localization RMSE, convergence time, and recovery outcome
Temporary aisle blockage and moving obstacles	Local avoidance, route recovery, global replanning, and supervised stop	DWA reaction; 0.3 m path-return rule; blockage timeout; latching and wireless stop	Static and controlled-dynamic mock-up repetitions; occupied-house event tally
Speed-dependent liquid delivery	Motion–spray command coupling with explicit saturation and no-spray gating	$Q_r = 60q_a*W_ev$; 0.35–0.90 L/min command bounds; $W_e = 0.48$ m	15 clean-water runs at three speeds; five repeated coverage runs per strategy; five quantified field cycles

The validation is staged to distinguish navigation, geometric coverage, liquid deposition, and microbiological reduction as separate evidence levels. This paper evaluates the integrated platform, navigation performance, route execution, structured-obstacle response, geometric footprint coverage, speed-dependent actual flow, effective swath width, and transverse clean-water deposition. Microbiological reduction and long-term commercial operation remain outside the present scope.

2 Tracked Robot Platform and Three-Level System Architecture

2.1 Hardware Platform

The tracked robot was assembled using a modular main-computer/slave-controller architecture. The main computer executed mapping, localization, planning, data logging, and remote visualization, while the slave controller handled low-level motor and spray commands. The platform integrated sensing, drive, spraying, communication, and power modules, as summarized in Fig. 1 and Table 2.

The main computer was a Jetson Xavier NX running Ubuntu 20.04 and ROS Noetic. Serial communication connected the main computer to the slave controller. Messages included left/right track velocity, actuator state, pump command, and pump-state information. No onboard closed-loop liquid-flow measurement is claimed.

The mobility unit used a differential tracked chassis driven by independently controlled left and right brushless DC motors with encoder feedback. Steering was achieved by regulating the speed difference between the two tracks; no steering servo was required for chassis direction control. This configuration supports zero-radius turning but may exhibit slip-induced odometry error on smooth or slatted floors.

The sensing unit included a SICK TiM781 2D LiDAR, a BMI088 IMU, and motor encoders. The LiDAR was mounted approximately 0.32 m above the ground and 0.06 m forward of the base_link origin, with its scanning plane parallel to the floor. It operated with a 270° field of view, 15 Hz scan rate, 0.33° angular resolution, and a software-limited range of 0.05–3.0 m. The BMI088 was mounted near the geometric center of the chassis at approximately 0.18 m above the ground and published data at 100 Hz. Encoder odometry was updated at 50 Hz. The adopted frame offsets were laser_link = (0.06, 0, 0.32) m and imu_link = (0, 0, 0.18) m relative to base_link, with axes aligned according to ROS REP-103.

The spraying unit consisted of an electric pump and a flat-fan nozzle. Pump start/stop and commanded flow were coordinated with robot motion. Pump-state feedback was logged separately from any direct flow measurement; the present evidence therefore supports command-level pump control and bench-calibrated flow estimates, not an onboard closed-loop flow measurement claim.

The auxiliary unit included a wireless router, lithium-battery power supply, and power-distribution module. Remote access through SSH and RViz was used for monitoring and emergency intervention during testing.

Protective covers and sealed electrical interfaces were used to reduce exposure to dust and moisture. Figure 2 shows the physical prototype. Approximate dimensions and mass are listed in Table 2 as engineering specifications and are not treated as experimental outcomes. With an overall width of approximately 0.39 m, the platform has about 0.205 m nominal lateral space on each side in a centered 0.8 m aisle.

Table 2. Main hardware and operational specifications of the prototype.

Item	Specification
Main computer	Jetson Xavier NX
Operating system	Ubuntu 20.04; ROS Noetic
LiDAR	SICK TiM781
IMU	BMI088

Item	Specification
Drive configuration	Differential tracked chassis with encoder feedback
Overall dimensions (L × W × H)	Approx. 480 × 390 × 420 mm
Total mass	Approx. 15.6 kg dry; 20.6 kg with 5 L water
Ground clearance	Approx. 55 mm
Chassis motion speed	0.10–0.30 m/s typical operating range; 0.60 m/s maximum; spray-enabled speed is constrained by Eq. (16)
Battery capacity and electrical endurance	24 V, 20 Ah; approx. 2.0 h under intermittent bench operation, excluding liquid-refill limits
Tank capacity	5 L detachable reservoir
Nozzle model and number	One ISO 80-02 flat-fan nozzle
Nozzle height and nominal footprint width	0.35 m mounting height; approx. 0.60 m nominal footprint width
Nominal pump-command range	0.35–0.90 L/min based on bench calibration; no onboard flow sensor
Track-belt width	Approx. 70 mm per track
Minimum passable corridor	Approx. 0.60 m geometric passage target; safety-clearance criterion reported separately
In-place turning radius	Approximately 0 m
Obstacle-crossing capability	Design/bench target: approx. 40 mm at low speed; not used as an evaluated outcome
Slope capability	Design/bench target: approx. 8°; not used as an evaluated outcome
Safety-stop arrangement	Latching motor/pump power cut-off and wireless remote stop

The platform used differential steering and could turn about its geometric center. The stated 0.60 m minimum corridor, 40 mm obstacle-crossing value, and 8° slope value are prototype design or bench targets unless supported by a documented repeated-test protocol. A latching emergency-stop switch interrupted motor and pump power, while a wireless remote-stop command provided supervisory intervention; these controls are prototype safeguards and are not presented as a certified functional-safety system. The 5 L reservoir provides a nominal continuous-spray duration of approximately 5.6–14.3 min across the 0.90–0.35 L/min command range, so the stated electrical endurance does not represent uninterrupted liquid autonomy.

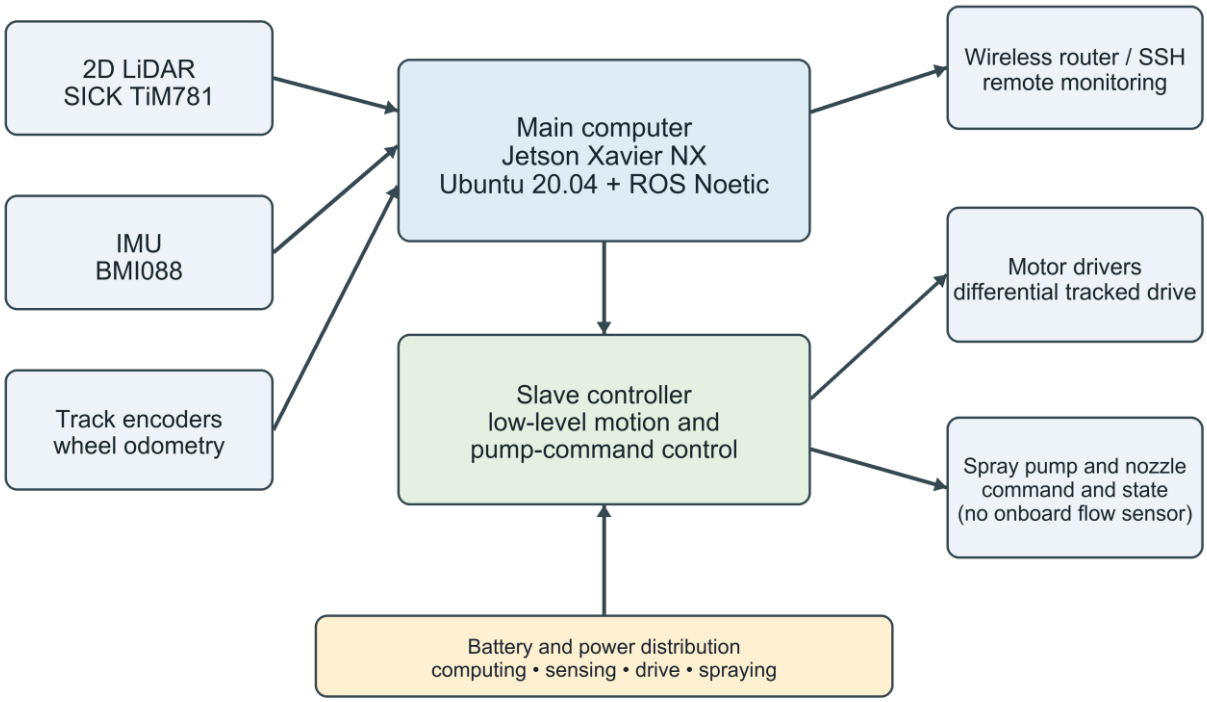

Fig. 1. Hardware architecture of the tracked spraying robot.

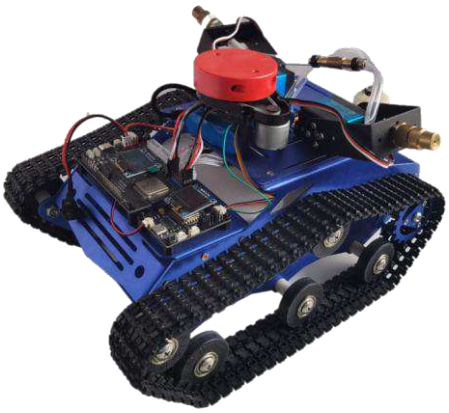

Fig. 2. Physical prototype of the tracked spraying robot.

2.2 ROS Software Architecture and Three-Level Coordination

The software platform was organized as a ROS-based sensing-state-estimation-task-planning-motion-control loop (Fig. 3). Sensor drivers published LiDAR, IMU, and encoder data; preprocessing nodes performed filtering, synchronization, coordinate transformation, and fused-odometry estimation; and GMapping or AMCL supplied the map and pose required by the planning layers.

Mapping and navigation were implemented as separate operating modes. In mapping mode, GMapping used filtered laser scans and fused odometry to construct the occupancy grid. In navigation mode, map_server loaded the saved map and AMCL estimated the robot pose. GMapping was not used to update the map during normal AMCL-based navigation.

The navigation hierarchy comprised three levels. At the task level, the bio-inspired coverage planner converted the reachable work region into an ordered set of coverage cells and task waypoints. At the global level, A* generated a route between successive task waypoints and cubic B-spline smoothing reduced abrupt heading changes while preserving the inflated obstacle-free corridor. At the local level, DWA used the current pose, local costmap, and LiDAR observations to generate collision-avoidance velocity commands. The information flow was therefore coverage task to global path to local motion command, rather than global path to coverage planning.

The control layer converted /cmd_vel commands into left- and right-track velocity references. The spray-control node calculated the required pump command from the target nominal areal rate q_a*, effective swath width W_e, and robot ground-speed command v. Pump-state feedback indicated actuator state only; it was not treated as liquid-flow feedback, and the nominal command range was not described as a closed-loop calibration range.

RViz and SSH were used to monitor pose, task waypoints, the global path, the local costmap, velocity commands, sensor status, and pump state. A supervisory node stopped motion and spraying when required sensor data were unavailable or when an obstacle blockage exceeded the recovery criterion. Figure 3 summarizes this corrected data flow.

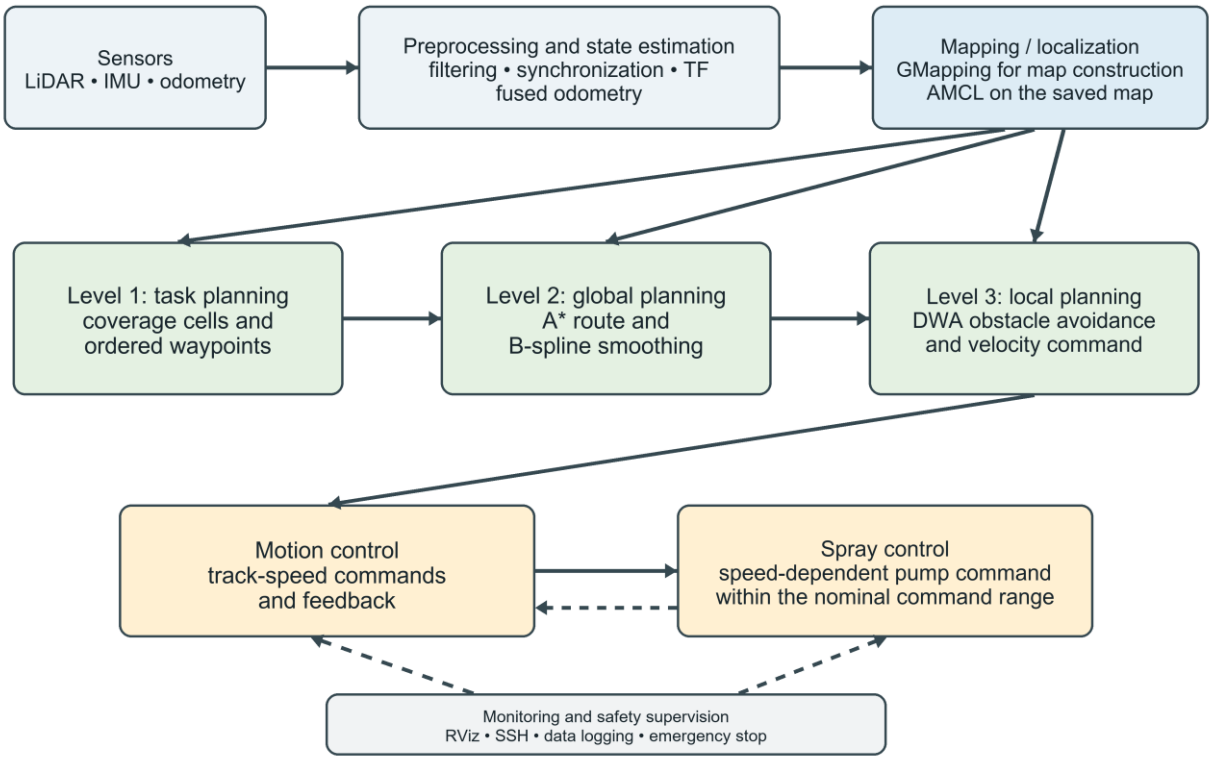

Fig. 3. ROS-based software architecture and data flow.

3 Autonomous Mapping and Map-Based Localization

3.1 Implementation of Robot Mapping

Reliable autonomous operation requires an occupancy map that is sufficiently accurate and unambiguous for localization and path planning. Dense pens, narrow aisles, dust, and intermittent occlusion can reduce scan quality and map consistency in large sheep houses. The mapping workflow was therefore selected and parameterized for robustness and computational feasibility within the tested two-dimensional layouts.

3.1.1 Selection of the Mapping Method and Scope of Applicability

For the available computing platform and the tested two-dimensional sheep-house layouts, GMapping was selected because it provides occupancy-grid mapping with moderate computational demand and can use LiDAR measurements together with odometric motion estimates. Cartographer offers graph-based optimization and loop-closure capabilities but generally requires greater computational resources, whereas Hector SLAM depends strongly on high-rate laser scans and can be sensitive to occlusion when odometry is unavailable (Jeong et al. 2025; Liu et al. 2024). This selection is an engineering trade-off for the tested environmental scale and does not imply that GMapping is universally superior in larger or more topologically complex barns.

An initial engineering-screening simulation compared GMapping, Cartographer, and Hector SLAM under three modeled sensor-interference conditions: unobstructed sensing, partial pen occlusion, and a dust/noise model (Fig. 4). The figure is retained as a descriptive configuration-screening summary because the available manuscript materials do not contain a complete repeated-run log for every algorithm-condition cell. It is therefore used to motivate the subsequent choice of GMapping, not to support a statistical claim that one SLAM method is universally superior. The separate repeated GMapping configuration assessment is reported in Section 5.1.

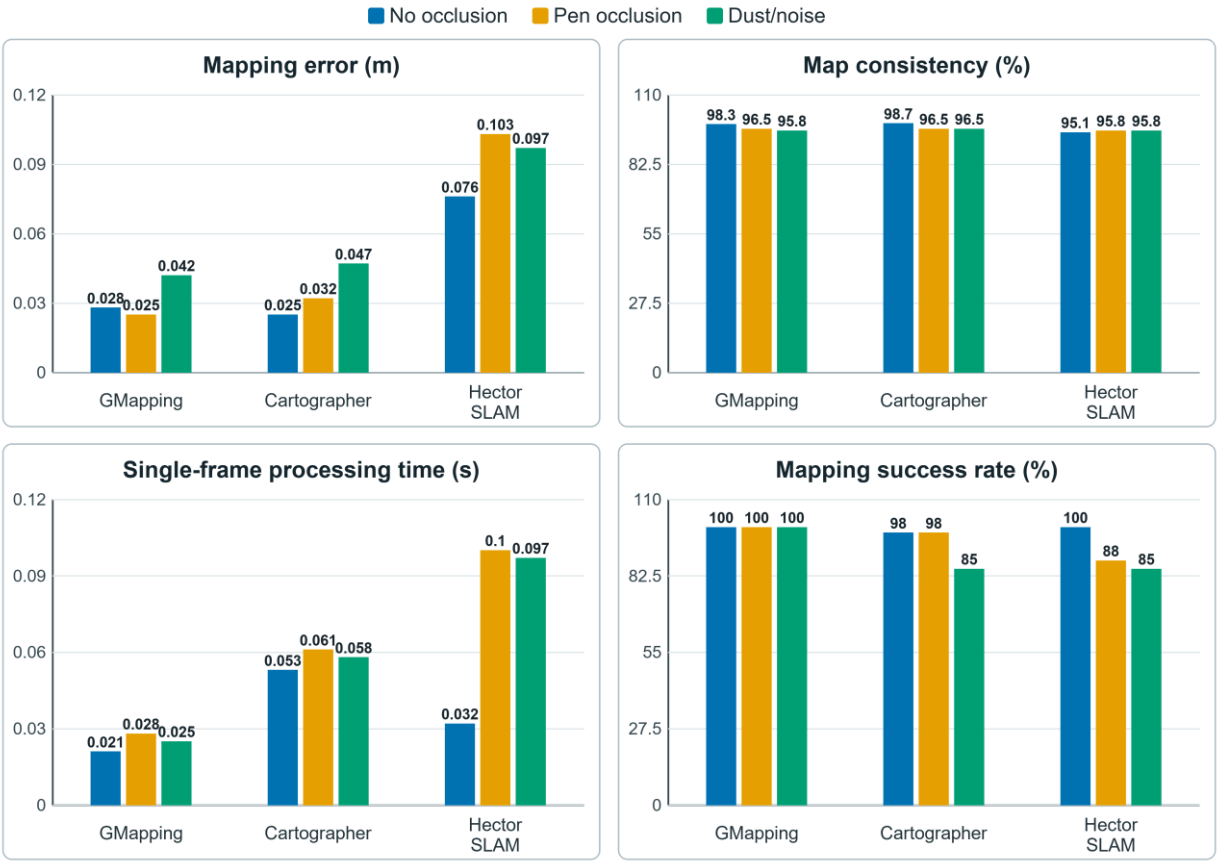

Fig. 4. Descriptive engineering screening of SLAM configurations under modeled sheep-house interference conditions.

3.1.2 Sensor Preprocessing and Odometry Fusion

LiDAR scans (15 Hz), BMI088 IMU measurements (100 Hz), and wheel-encoder odometry (50 Hz) were time-aligned using the ROS message_filters ApproximateTimeSynchronizer with a maximum synchronization tolerance of 20 ms and transformed into the base_link coordinate frame. A five-sample moving-average filter was applied to IMU angular velocity and acceleration, and the robot_localization extended Kalman filter produced fused odometry at 50 Hz from IMU and encoder inputs. GMapping subsequently combined the fused /odom estimate with the /scan measurements; raw LiDAR data were not directly fused into the EKF. Scan quality was evaluated using the valid-return ratio $r_v = \frac{N_{\text{valid}}}{N_{\text{total}}}$ after applying the 0.05–3.0 m range gate. Scans with $r_v \geq 0.65$ were accepted directly, scans with $0.55 \leq r_v < 0.65$ were processed using a three-frame median filter, and scans with $r_v < 0.55$ were temporarily rejected while the robot pose was propagated using fused odometry. The 20 ms value was a synchronization setting; inter-topic offsets were not retained as an independent timing-performance dataset.

The rule-based scan-confidence gate was defined as $g(r_v) = 1$ for $r_v \geq 0.65$, $g(r_v) = 0.5$ for $0.55 \leq r_v < 0.65$, and $g(r_v) = 0$ for $r_v < 0.55$. This gate controlled scan acceptance and filtering only; it did not directly modify the internal particle weights of GMapping. This implementation avoids presenting the preprocessing logic as a new sensor-fusion algorithm and makes the mapping workflow reproducible.

During map construction, filtered laser scans were published through /scan and the fused motion estimate through /odom, with coordinate transformations maintained through the ROS TF tree. GMapping generated an occupancy-grid map at a resolution of 0.05 m. After map construction, the map was saved and loaded through map_server. During subsequent navigation, AMCL—not GMapping—estimated the robot pose on the prebuilt map and supplied localization information to the global and local planners.

3.1.3 Autonomous Exploration and Parameterization

A breadth-first frontier strategy identified candidate boundaries between known and unknown cells. Each candidate was scored using travel cost, which incorporated obstacle clearance, path length, and heading change, together with information gain representing the surrounding unexplored area. The next target balanced low travel cost against high expected information gain. Exploration performance was assessed using exploration rate, mapping time, and map similarity rather than visual inspection alone.

The frontier travel-cost function was defined as:

$$J_{\text{cost}} = \omega_1 \cdot \frac{1}{d_{\text{obs}} + 0.1} + \omega_2 \cdot L_{\text{path}} + \omega_3 \cdot \theta_{\text{dev}} \tag{1}$$

where d_{obs} is the minimum distance from the candidate frontier point to obstacles, L_{path} is the path length from the current position to the candidate point, θ_{dev} is the deviation angle between the robot's traveling direction and the direction toward the candidate point, and $\omega_1 = 0.6$, $\omega_2 = 0.3$, $\omega_3 = 0.1$ are the weighting coefficients. The resulting cost is therefore interpreted as an empirical ranking score rather than a physical quantity.

The corresponding information-gain function was defined as:

$$J_{\text{info}} = S_{\text{unexplored}} \tag{2}$$

where $S_{\text{unexplored}}$ is the area of the unexplored region within a $5\ \text{m} \times 5\ \text{m}$ window around the candidate point.

Scenario-oriented GMapping parameters were selected from repeated configuration runs using map IoU, geometric boundary error, mapping time, CPU occupancy, and memory use. Particle number, usable LiDAR range, map-update interval, and linear/angular update thresholds were evaluated as engineering variables for the 20 × 10 m sheep-house layout. The final setting was selected by jointly considering accuracy and the computing budget of the Jetson Xavier NX.

A particle-number configuration study examined 50, 100, 200, and 300 particles with 15 runs per level. Map IoU values were 93.4 ± 0.9%, 95.6 ± 0.7%, 96.8 ± 0.8%, and 97.1 ± 0.5%, respectively; the corresponding geometric errors were 0.059 ± 0.010, 0.042 ± 0.007, 0.030 ± 0.004, and 0.028 ± 0.004 m. One-way ANOVA showed particle-number effects for IoU, error, mapping time, CPU, and memory (all $p < 0.001$). In the focused 200-versus-300 comparison, IoU ($p = 0.230$) and geometric error ($p = 0.182$) did not differ significantly, whereas mapping time, CPU occupancy, and memory use were all higher at 300 particles (all $p < 0.001$). The 200-particle setting was therefore selected as the accuracy–resource trade-off for onboard operation.

The maximum usable LiDAR range (maxRange) was set to 3.0 m to emphasize local aisle and pen boundaries and reduce the influence of distant invalid returns in the tested layouts. This value is specific to the aisle dimensions and sensor placement used in the experiments.

The map-update interval (map_update_interval) was set to 0.5 s to balance occupancy-grid publication frequency and computational stability. This parameter controls how frequently the generated map is updated; it does not imply that moving animals or other transient objects are modelled as persistent changes to a dynamic map.

The linear and angular update thresholds were set to 0.2 m and 0.1 rad, respectively. These thresholds determine when new motion and scan information is incorporated into the GMapping update process and were selected to reduce unnecessary updates caused by small chassis vibrations while retaining sufficient spatial detail.

3.2 Autonomous Exploration Workflow

The mapping workflow was implemented in Ubuntu 20.04 and ROS Noetic, with the virtual sheep-house environment constructed in Gazebo and the exploration state visualized in RViz. Figure 5 illustrates the progression from local map initialization to frontier selection, traversal of multiple pen boundaries, and completion of the occupancy-grid map.

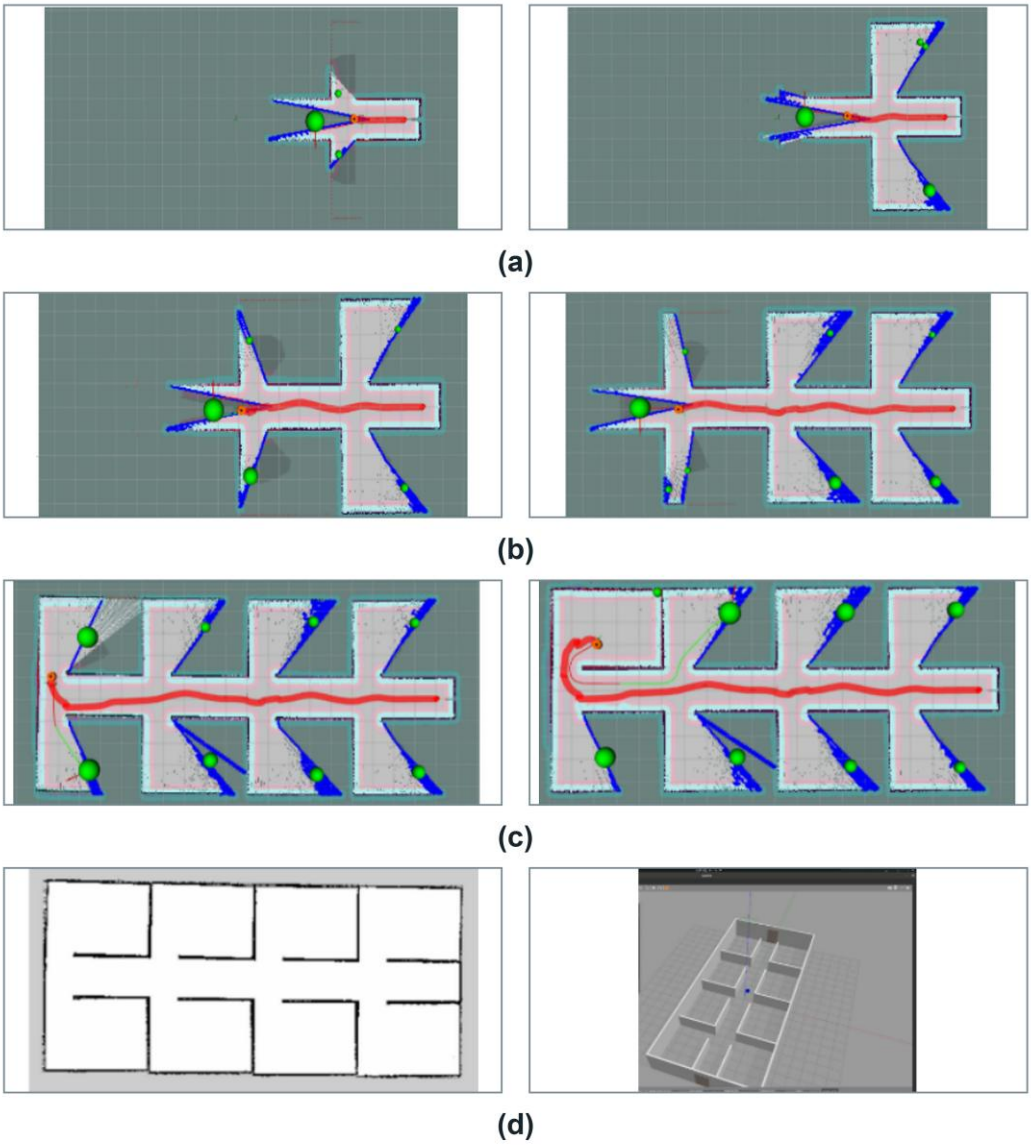

Fig. 5. Autonomous exploration and mapping workflow in the simulated sheep-house environment: (a) initial exploration; (b) intermediate exploration; (c) final exploration; and (d) generated occupancy-grid map.

At the initial stage, LiDAR and odometry data were used to establish a local map and identify the first frontier. During exploration, the robot repeatedly selected a frontier target, planned a collision-free route, and updated the occupancy grid as new boundaries were observed. At the final stage, the remaining reachable frontiers were processed and the resulting map was saved for navigation. The visual sequence in Fig. 5 demonstrates the workflow; quantitative mapping accuracy and computation results are presented in Section 5.1.

Exploration was initiated from multiple starting positions to examine sensitivity to the initial pose. Successful completion was defined by the absence of reachable frontiers above the minimum-size threshold and by the generation of a navigable occupancy grid. Map quality was evaluated against a reference layout rather than inferred solely from path smoothness or screenshot appearance.

3.3 Map Post-Processing and Transition to Navigation

After map construction, isolated occupancy-grid noise was reduced using morphological filtering, and the effective operating region was cropped to remove unused space. The processed map was saved in .pgm and .yaml formats and published through map_server. No independent RANSAC-based loop-closure module is claimed in the revised workflow; global consistency is therefore discussed within the known limitations of the selected GMapping implementation.

The mapping and navigation stages were separated. GMapping was used only while constructing the occupancy-grid map. After the map was saved, AMCL estimated the robot pose on the prebuilt map, while the global and local planners read obstacle information from the static and local costmaps. This separation avoids conflating SLAM pose estimation with map-based localization and provides a reproducible mapping–localization–planning workflow.

3.4 AMCL-Based Localization on the Prebuilt Map

Map-based localization was required for path tracking and obstacle avoidance in repetitive sheep-house layouts. After map construction, AMCL estimated the robot pose from LiDAR scans, odometry, and the static occupancy map. The evaluation considered partial scan occlusion, track slip, temporary dynamic obstacles, and recovery from a deliberately disturbed initial pose.

AMCL represents the posterior pose distribution using weighted particles and applies a motion model, laser observation model, and resampling procedure. Unlike fixed-particle Monte Carlo localization, KLD sampling adjusts the number of particles according to the estimated uncertainty. Random-particle injection supports recovery when the current scan has low likelihood under the existing particle distribution (Peavy et al. 2023).

AMCL provided map-based pose estimation after the occupancy map had been saved. The ROS Noetic implementation used the laser observation model, odometry motion model, KLD-based particle adaptation, and recovery-particle mechanism. Evaluation used the Gazebo model pose as planar ground truth and tested repeatable initialization or pose-disturbance events on the same reference trajectory.

Ten localization trials were performed in the simulated sheep-house map. Each trial followed the same reference trajectory and included a defined initialization or pose-disturbance event. Planar position RMSE was calculated against the Gazebo ground-truth pose, and convergence time was measured until the pose error remained within the configured tolerance. Trial-level RMSE and convergence values are reported as mean ± sample standard deviation; recovery is reported as a binomial outcome with a Wilson 95% confidence interval.

AMCL calculates an average likelihood from the particle importance weights:

$$\overline{\omega}_k = \frac{1}{N}\sum_{i=1}^{N} \omega_k^{(i)} \tag{3}$$

where $\bar{\omega}_k$ denotes the average particle likelihood at update k, N is the total number of particles, and $\omega_k{}^{(i)}$ is the importance weight of the i-th particle. Equations (4) and (5) then update the long- and short-term likelihood averages used by the recovery mechanism:

$$\overline{\omega}_{\text{long}} = (1-\lambda_{\text{long}})\overline{\omega}_{\text{long}} + \lambda_{\text{long}}\overline{\omega}_k \tag{4}$$

$$\overline{\omega}_{\text{short}} = (1-\lambda_{\text{short}})\overline{\omega}_{\text{short}} + \lambda_{\text{short}}\overline{\omega}_k \tag{5}$$

where ω_{long} and ω_{short} are the long- and short-term likelihood averages, and λ_{long} and λ_{short} are their update coefficients. The condition $0 < \lambda_{\text{long}} < \lambda_{\text{short}} \leq 1$ makes the short-term estimate respond more rapidly. In the standard recovery logic, the random-particle probability is $p_{\text{rand}} = \max\left(0, 1 - \frac{\omega_{\text{short}}}{\omega_{\text{long}}}\right)$ when $\omega_{\text{long}} > 0$. Because the retained files do not verify an independent parameter-optimization study, no optimized numerical values are claimed here. Equations (3)–(5) therefore explain the recovery mechanism rather than a new AMCL algorithm. The workflow is shown in Fig. 6.

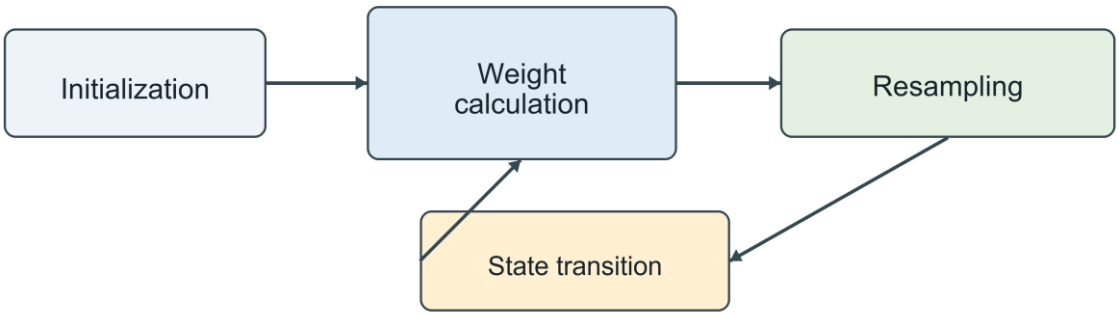

Fig. 6. AMCL particle-filter workflow.

4 Hierarchical Navigation, Coverage Planning, and Spraying Coordination

4.1 Robot Global Path Planning

4.1.1 The A Algorithm*

A* is a heuristic graph-search algorithm that combines the accumulated path cost with an estimate of the remaining distance to the goal. It is suitable for global planning on a static occupancy grid, but grid-based paths may contain unnecessary

heading changes and are not designed to react directly to moving obstacles. These characteristics motivate the use of geometric smoothing and a separate local planner.

4.1.2 B-Spline Smoothing of the A Path*

Cubic B-spline smoothing was applied to the control points extracted from the A* path to reduce abrupt heading changes. B-spline curves provide local controllability and continuity and can generate trajectories that are more compatible with the differential tracked chassis than the original grid path. The smoothing step was constrained to remain within the inflated obstacle-free corridor of the costmap.

$$P(t) = \sum_{i=0}^{n} P_i \, F_{i,k}(t) \tag{6}$$

$$F_{i,k}(t) = \frac{1}{k!} \sum_{m=0}^{k-i} (-1)^m \binom{m}{k+1} (t + k - m - i)^k \tag{7}$$

In Eqs. (6) and (7), P_i is the i-th control point, $F_{i,k}(t)$ is the i-th basis function of degree k, $t \in [0,1]$ is the local parameter of a curve segment, and n is the highest control-point index. For the cubic segment used here, $k = 3$ and $n = 3$, giving the four basis functions:

$$\mathrm{F}_{0,3}(\mathrm{t}) = \frac{1}{6}(1 - \mathrm{t})^3 \tag{8}$$

$$\mathrm{F}_{1,3}(\mathrm{t}) = \frac{1}{6}(3\mathrm{t}^3 - 6\mathrm{t}^2 + 4) \tag{9}$$

$$\mathrm{F}_{2,3}(\mathrm{t}) = \frac{1}{6}(-3\mathrm{t}^3 + 3\mathrm{t}^2 + 3\mathrm{t} + 1) \tag{10}$$

$$\mathrm{F}_{3,3}(\mathrm{t}) = \frac{1}{6}\mathrm{t}^3 \tag{11}$$

Substituting the cubic basis functions into the general B-spline expression yields the smoothed curve used to connect successive A* control points.

$$P(t) = P_0 \cdot F_{0,3}(t) + P_1 \cdot F_{1,3}(t) + P_2 \cdot F_{2,3}(t) + P_3 \cdot F_{3,3}(t) \tag{12}$$

The MATLAB comparison in Fig. 7 illustrates the geometric effect of smoothing on the selected control points. It does not by itself demonstrate collision avoidance, reduced search time, or improved execution. Those claims depend on obstacle-corridor checks and repeated path-length, clearance, and tracking measurements.

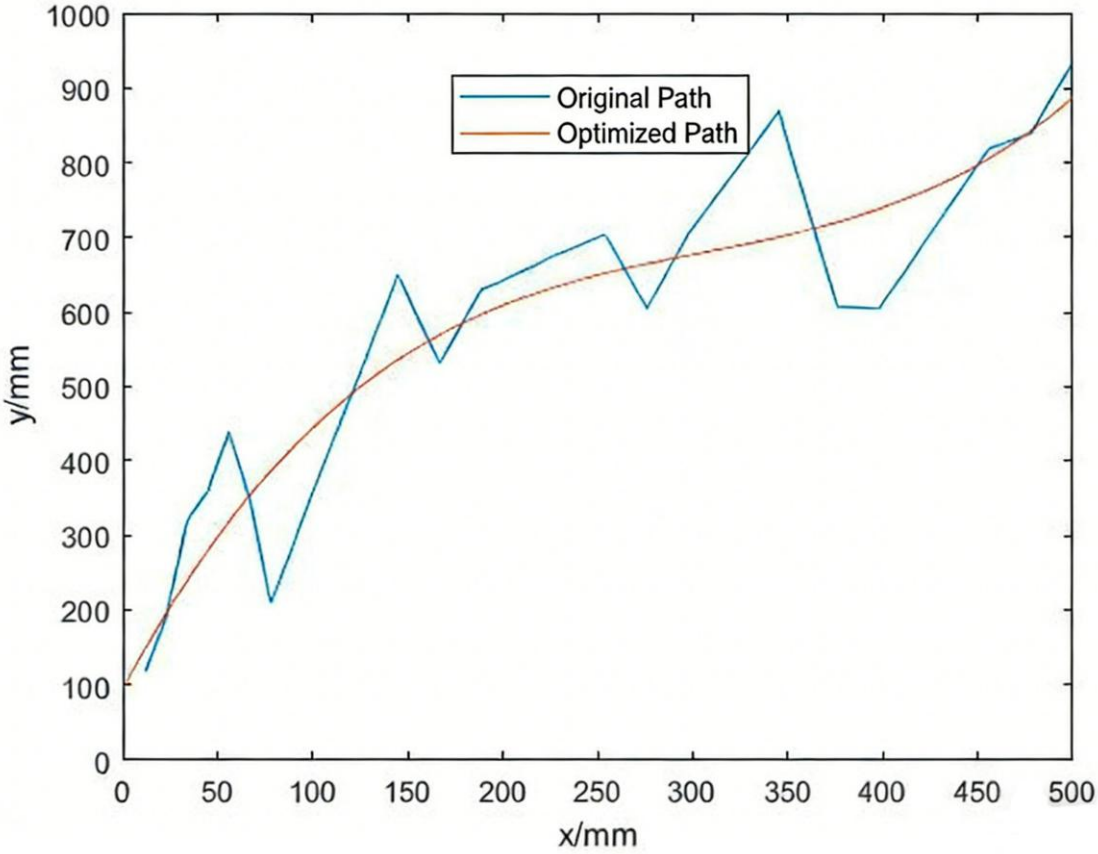

Fig. 7. Geometric comparison of the original and B-spline-smoothed A* paths.

4.2 Global-Local Navigation

4.2.1 Hybrid Path Planning Based on A and DWA*

The Dynamic Window Approach samples dynamically admissible linear and angular velocities within a finite prediction horizon. Candidate trajectories are simulated and evaluated using goal heading, path alignment, obstacle clearance, and velocity-related terms. The highest-scoring admissible velocity command is then applied to the robot. This enables real-time local avoidance while respecting the velocity and acceleration constraints of the tracked platform.

The smoothed A* path provides global guidance on the static map, whereas DWA updates the local motion command using current LiDAR observations and the local costmap. If a moving obstacle blocks the reference path, DWA selects a temporary collision-free trajectory; after the obstacle is cleared, the robot returns to the global path or requests replanning when the deviation exceeds the configured threshold. The contribution is the coordination of established global and local planners within the sheep-house task workflow.

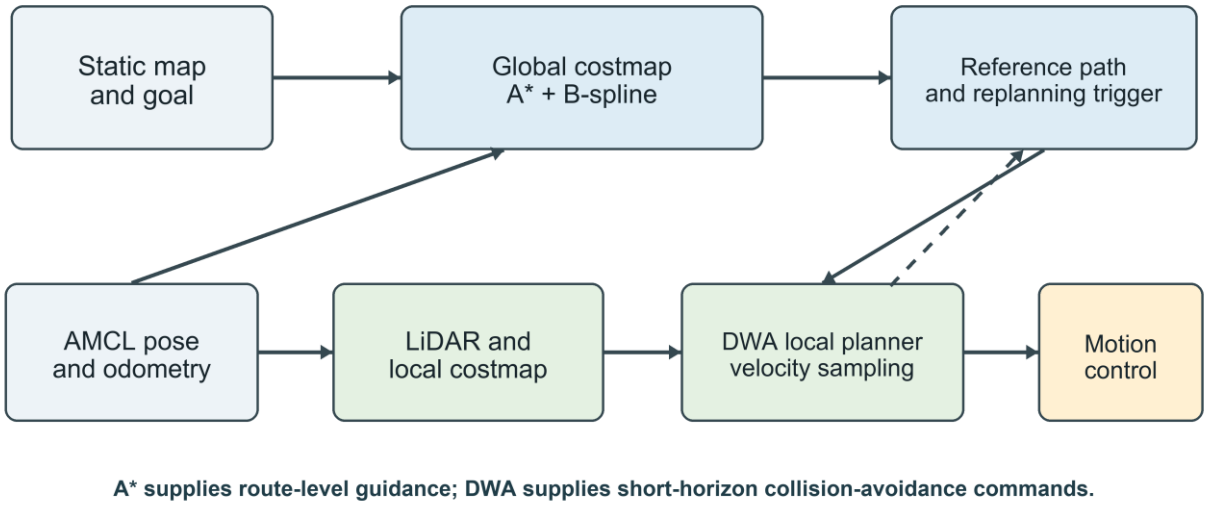

Fig. 8. Coordination of global and local path planning.

4.3 Bio-Inspired Coverage Planning and Spraying Coordination

4.3.1 Coverage-Planning Model

The coverage planner represented each occupancy-grid cell i by a dimensionless neural activity $x_i(t)$. Let $\mathcal{N}_i$ denote the finite set of cells connected to cell i. The activity state was described by the bounded shunting model (Singh and Nagla 2024):

$$\tau \frac{\mathrm{d}x_i}{\mathrm{d}t} = -a\, x_i + (x_\mathrm{U} - x_i)\left([I_i]^+ + \sum_{j \in \mathcal{N}_i} w_{ij}^+ [x_j]^+\right) - (x_i - x_\mathrm{L})\left([I_i]^- + \sum_{j \in \mathcal{N}_i} w_{ij}^- [x_j]^-\right) \tag{13}$$

In Eq. (13), $\tau > 0$ is the update time constant, $a > 0$ is the passive decay coefficient, and $x_u > 0$ and $x_l < 0$ are the upper and lower activity bounds. I_i is the external input; $[z]^+ = \max(z, 0)$ and $[z]^- = \max(-z, 0)$. The non-negative weights w^+_{ij} and w^-_{ij} denote excitatory and inhibitory coupling, respectively. Unvisited traversable cells receive positive input, whereas occupied, no-entry, and previously visited cells receive inhibitory input. Both summations are restricted to $j \in \mathcal{N}_i$; this removes the undefined summation range in the previous notation.

At planning step k, collision checking is separated from neural-state ranking. The next cell is selected as:

$$i_{k+1} = \arg\max_{j \in \mathcal{A}(i_k)} x_j \tag{14}$$

where $\mathcal{A}(i_k) \subseteq \mathcal{N}_{i_k}$ is the set of neighboring cells that are unoccupied and satisfy the robot-footprint and minimum-clearance constraints. If several candidates have equal activity within the numerical tolerance, the candidate requiring the smallest heading change is selected. Consequently, Eq. (14) defines the task-level cell choice, whereas the A* and DWA layers remain responsible for feasible route generation and short-horizon collision avoidance.

Priority is encoded as a dimensionless multiplier ρ_p on the positive input of selected traversable cells, not as a change in spraying dose. The sensitivity settings formerly described as 10%, 20%, 30%, and 40% increments are therefore written as $\rho_p \in \{1.10, 1.20, 1.30, 1.40\}$. The retained descriptive setting $\rho_p = 1.30$ was the lowest tested multiplier that advanced the selected priority cells without repeated switching. Drainage areas may be prioritized where permitted by the operating protocol; drinking-water zones and other contamination-sensitive areas are represented as no-spray masks.

4.3.2 Spraying-Task Coordination

The nominal geometric spray half-width was $R_s = 0.30$ m. For a flat-fan angle $\alpha = 80°$ and nozzle height $h = 0.35$ m, ideal planar geometry gives a footprint width $W_s = 2h \tan(\alpha/2) = 0.587$ m, rounded to 0.60 m for coverage planning. With the nominal overlap ratio $r_o = 0.20$, the effective swath width, which is also the adjacent centerline spacing used in the areal-rate calculation, is $W_e = (1 - r_o)W_s \approx 0.48$ m. Thus, the single-pass footprint width W_s and effective swath width W_e are not interchangeable.

For spray-enabled motion, the required and commanded flow rates are defined by:

$$\begin{aligned} W_\mathrm{e} &= (1 - r_\mathrm{o}) W_\mathrm{s} \\ Q_\mathrm{r}(t) &= 60\, q_\mathrm{A}^*\, v(t)\, W_\mathrm{e} \\ Q_\mathrm{p}(t) &= s(t) \cdot \mathrm{clip}[Q_\mathrm{r}(t), Q_\mathrm{min}, Q_\mathrm{max}] \end{aligned} \tag{15}$$

In Eq. (15), q_a^* is the target nominal areal application rate (L/m²), v(t) is robot ground speed (m/s), Q_r is the unconstrained required flow (L/min), and Q_p is the pump command (L/min). The clip operator limits Q_r between the bench-supported lower and upper command bounds, Q_{min} = 0.35 L/min and Q_{max} = 0.90 L/min, respectively. The binary variable s(t) is the spray-enable gate: it is zero in no-spray masks and during stopping or non-spraying transition maneuvers, and one only on an enabled coverage segment.

The speed interval over which the target areal rate can be maintained without flow saturation is:

$$v_{\min} = \frac{Q_{\min}}{60\, q_{\mathrm{A}}^{*}\, W_{\mathrm{e}}}, \quad v_{\max} = \frac{Q_{\max}}{60\, q_{\mathrm{A}}^{*}\, W_{\mathrm{e}}} \tag{16}$$

For q_a^* = 0.20 L/m² and W_e = 0.48 m, Eq. (16) gives 0.061 ≤ v ≤ 0.156 m/s. At an illustrative speed of 0.10–0.12 m/s, the required command is 0.576–0.691 L/min. These corrected values differ from the previous calculation because the earlier expression used the 0.60 m footprint width and then stated a separate 20% overlap without reducing the effective swath width. Consequently, the chassis operating range in Table 2 cannot be interpreted as a spray-enabled range at this target rate; spray-enabled segments must satisfy Eq. (16) or accept command saturation.

No online liquid-flow or deposition sensor was installed. Equations (15) and (16) therefore describe command-level coordination rather than closed-loop regulation of delivered or deposited liquid. For an enabled segment with v > 0, a command-derived nominal rate may be calculated from $Q_p/(60vW_e)$, but deposition uniformity, drift, and microbiological efficacy require independent measurements using water-sensitive paper, collection trays, surface sampling, or microbiological assays.

4.4 Experimental Design and Statistical Analysis

Continuous outcomes are reported as mean ± sample standard deviation. Two independent configuration summaries were compared with two-sided Welch t tests. Particle-number effects were evaluated by one-way ANOVA, followed by the prespecified 200-versus-300 Welch comparison. For the three speed groups and three coverage strategies, the three pairwise Welch tests for each outcome were adjusted using the Bonferroni procedure. Static and controlled-dynamic mock-up outcomes were compared with a two-sided Welch test where appropriate. Binomial completion outcomes are reported with Wilson 95% confidence intervals; planning success counts were compared using two-sided Fisher exact tests with Holm adjustment. The significance level was α = 0.05.

The experimental program covered March 2025 to March 2026. Repeated coverage records spanned 18 August–22 September 2025, and the five field cycles spanned 12 January–18 March 2026. Inference was restricted to retained trial-level values; missing observations were not reconstructed. Coverage, spray, obstacle, and field results were calculated from the author-supplied workbook using the definitions reported below.

5 Experimental Evaluation

5.1 Simulation-Based Navigation and Coverage Evaluation

5.1.1 Mapping Configuration Evaluation

The SLAM screening and original-versus-revised GMapping comparison served different purposes. Method screening motivated the selected SLAM family, whereas the configuration comparison used 15 repeated runs per configuration in the 20 × 10 m simulated layout. The latter comparison is analyzed with Welch tests; the screening results remain descriptive because their original factor allocation was not retained.

The LiDAR interference conditions were simulation noise models rather than measured physical dust concentrations. The low-interference model used zero-mean Gaussian range noise with σ = 0.01 m, 5% invalid returns, and 1% uniform outliers within ±0.10 m; the high-interference model used σ = 0.03 m, 15% invalid returns, and 3% outliers within ±0.20 m. These settings define the modeled conditions only. Because the original factor structure and paired-run allocation cannot be verified from the available materials, the previously stated ANOVA statistics are not retained.

In the 20 × 10 m layout, the revised configuration produced a map IoU of 96.8 ± 0.8%, mean geometric boundary error of 0.030 ± 0.004 m, mapping time of 738 ± 29 s, CPU occupancy of 38.2 ± 2.1%, and memory use of 128 ± 5 MB. The original configuration produced 93.6 ± 1.2% IoU and a mapping time of 988 ± 42 s. IoU increased significantly (Welch t = 8.59, df = 24.39, p < 0.001, Hedges g = 3.05), while mapping time decreased by 25.3% (Welch t = −18.97, df = 24.88, p < 0.001, Hedges g = −6.74). These results quantify the improvement for the tested layout and onboard configuration.

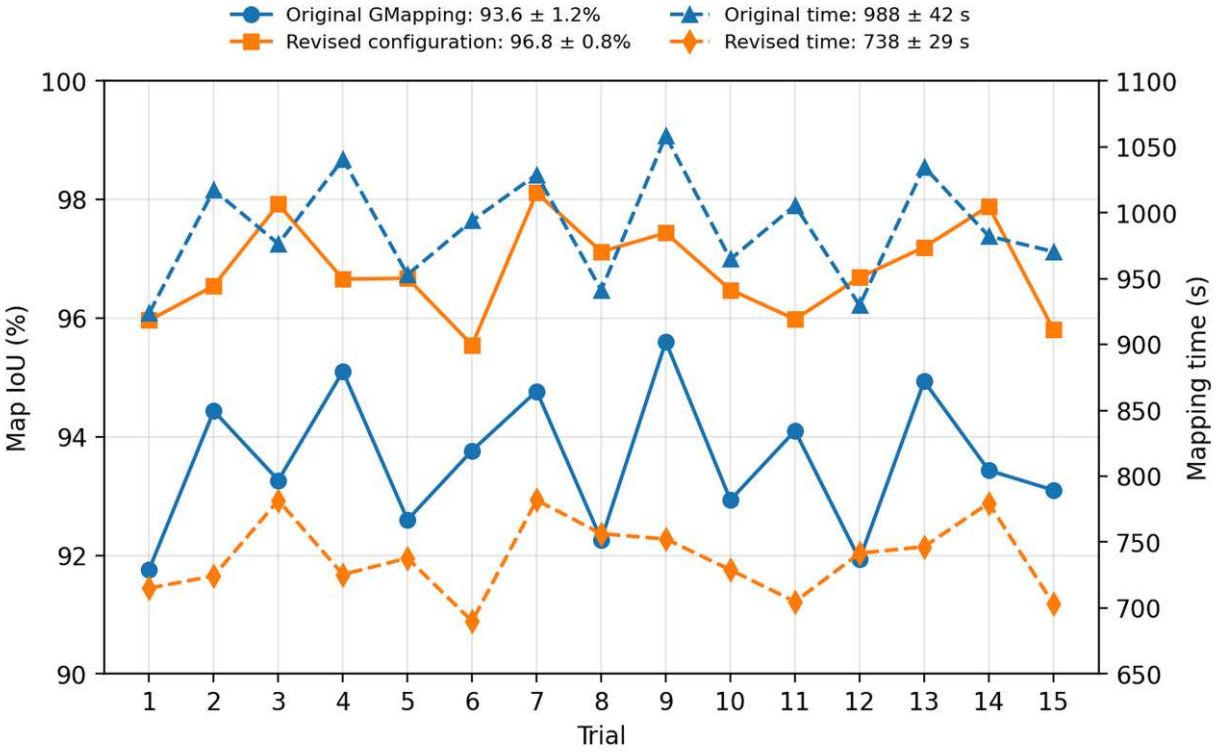

Fig. 9. Comparison of the original and revised GMapping configurations.

5.1.2 Map-Based Localization Evaluation

The localization evaluation comprised ten simulation trials using the same reference trajectory and a defined initialization or pose-disturbance event. The Gazebo model pose served as ground truth for planar position RMSE. Figure 10 shows representative particle dispersion and convergence, and Fig. 11 reports the retained trial-level RMSE, convergence time, and recovery outcome.

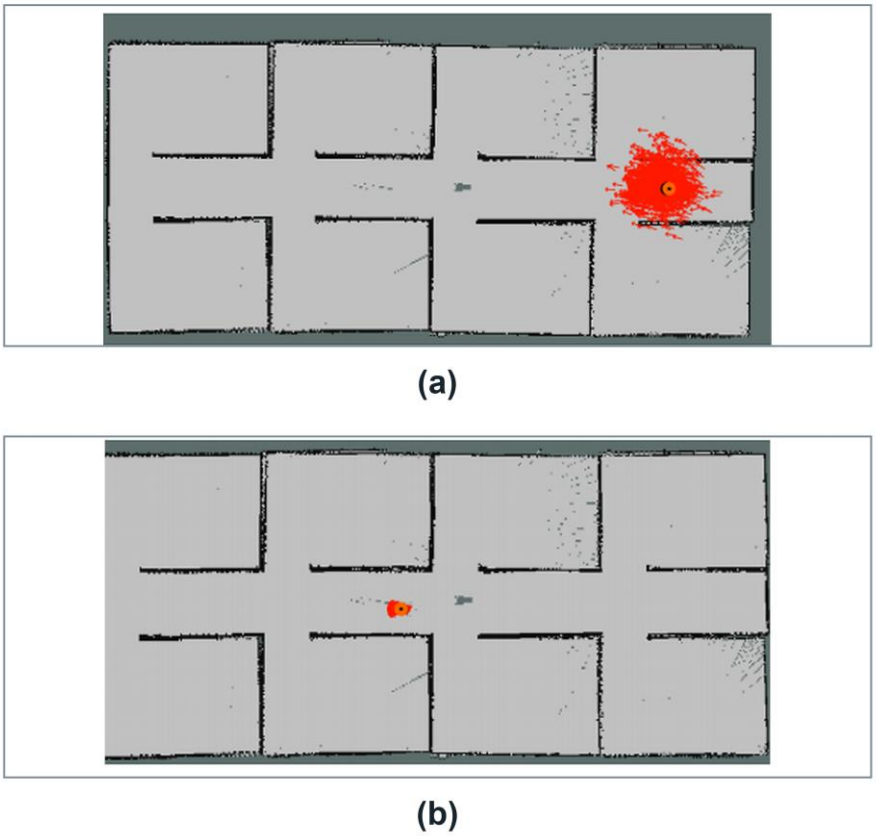

Fig. 10. AMCL localization process: (a) particle distribution after initialization; and (b) particle convergence after robot motion.

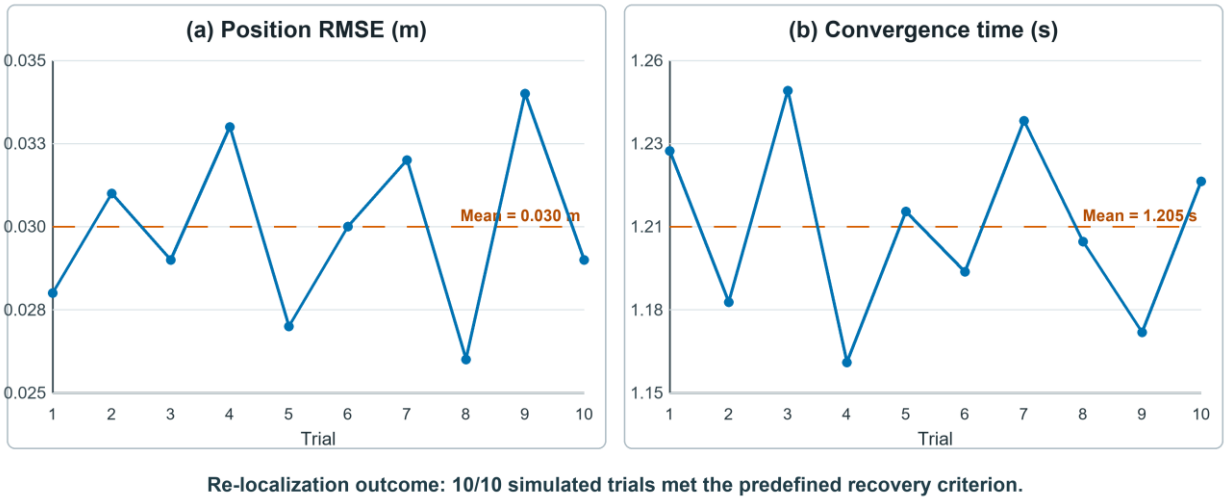

Fig. 11. Localization performance across ten simulated trials.

Across the ten simulated trials, planar localization RMSE was 0.0299 ± 0.0026 m and convergence time was 1.205 ± 0.030 s. All ten pose-disturbance trials recovered within the configured tolerance; the Wilson 95% confidence interval for 10/10 recovery was 72.2–100%. During navigation, AMCL supplied pose estimates to the planners without modifying the saved GMapping map.

5.1.3 Global-Local Planning Benchmark

The global-local planning benchmark used a simulated sheep-house layout with static pen boundaries and eight predefined dynamic-obstacle scenarios. A 0.45 m diameter, 0.70 m high cylindrical obstacle followed one transverse and one head-on trajectory at each of 0.2, 0.4, and 0.6 m/s, together with two aisle-blocking trajectories with dwell times of 5 and 10 s. Each scenario was repeated ten times, yielding 80 reported trials per configuration. Success required collision-free completion, at least 0.20 m clearance, target arrival within 1.5 times the unobstructed completion time, and no manual intervention. Path length, task completion, local-minimum occurrence, minimum clearance, and the different timing scopes were recorded separately.

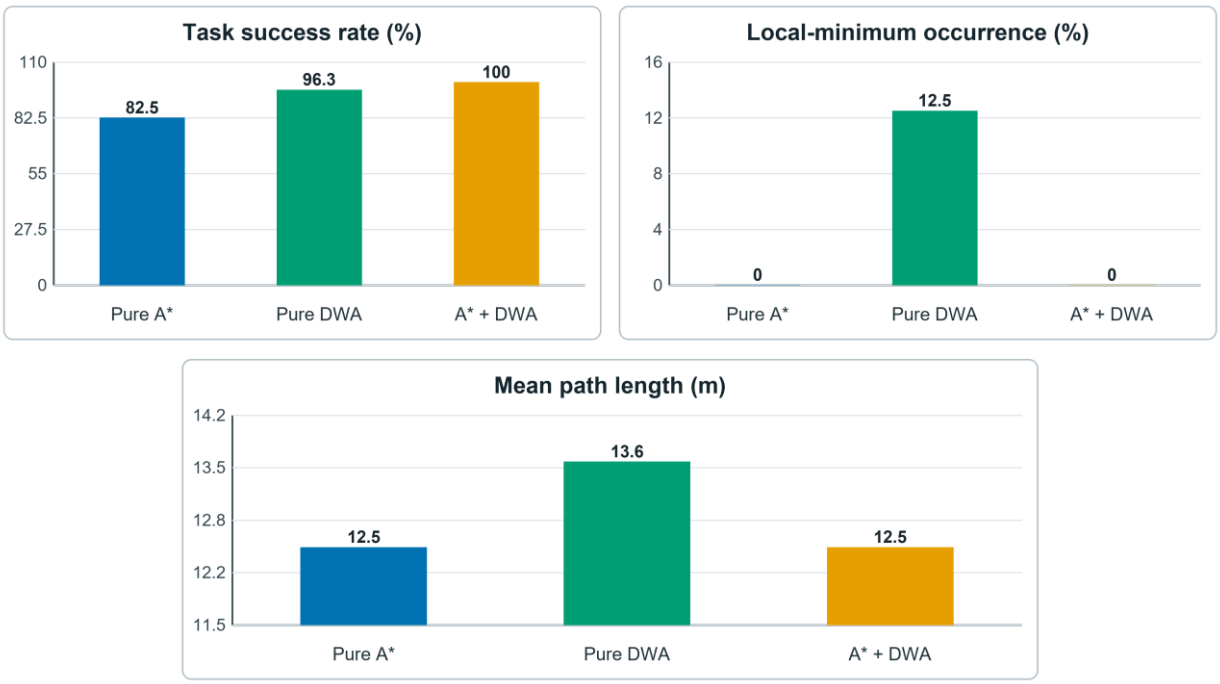

Fig. 12. Comparative performance of the path-planning configurations.

Across 80 trials per configuration, A* completed 66 trials (82.5%, Wilson 95% CI 72.7–89.3%), DWA completed 77 (96.3%, 89.5–98.7%), and the integrated A*–DWA configuration completed 80 (100%, 95.4–100%). After Holm adjustment, the integrated configuration outperformed A* (adjusted $p < 0.001$), and DWA outperformed A* (adjusted $p = 0.017$); the integrated-versus-DWA completion difference was not significant (adjusted $p = 0.245$). Pure DWA entered a local-minimum state in 10/80 trials (12.5%), whereas the integrated configuration could invoke the global route. Mean path lengths were 12.5, 13.6, and 12.5 m, respectively, and the integrated configuration maintained a mean minimum clearance of 0.31 ± 0.05 m in the specified scenarios.

5.1.4 Coverage-Planning Benchmark

The coverage-planning benchmark descriptively compared the bio-inspired planner with boustrophedon and spiral baselines in one simulated workspace. Reported coverage times were 78.3, 92.5, and 85.7 s; geometric completeness values were 100.0%, 98.7%, and 99.2%; and repeated-coverage ratios were 4.2%, 12.5%, and 8.7%, respectively. The corresponding 0.203, 0.191, and 0.197 L/m² values are retained as command-derived nominal rate indices rather than measured deposition. They should be recalculated from synchronized speed and pump-command logs using Eq. (15) before any inferential comparison. Because a common repeated-trial log and a fully specified denominator mask were unavailable, the comparison remains descriptive and does not establish an inferential ranking.

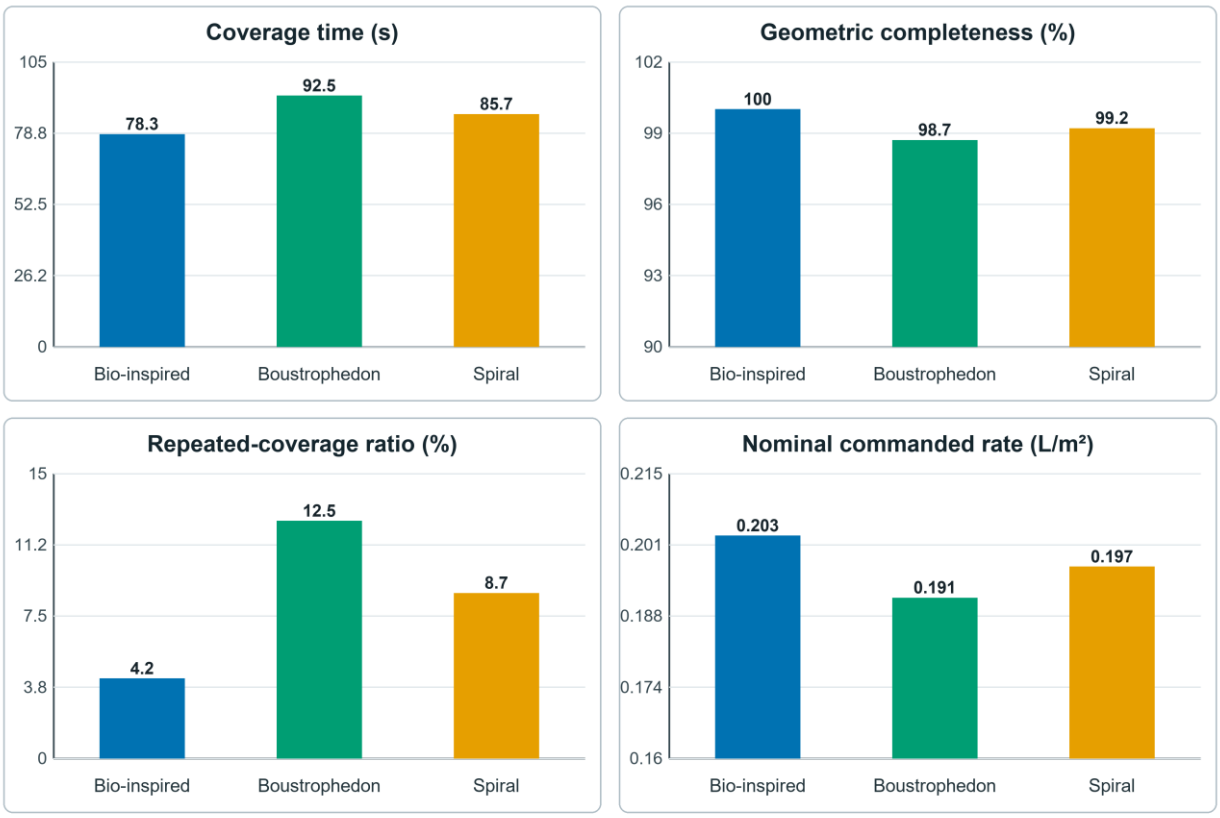

Fig. 13. Comparative path-coverage metrics in the simulated layout.

5.2 Physical Mock-Up Navigation and Geometric Coverage Evaluation

5.2.1 Multi-Waypoint Path-Following Test

A physical mock-up with narrow corridors and pen-like partitions was used to evaluate waypoint tracking and obstacle avoidance. The retained event-level dataset comprised ten static-obstacle events and ten controlled-dynamic events. Static cases included left-offset, right-offset, central, and paired-cylinder layouts. Dynamic cases included transverse crossing, head-on approach, aisle blockage, and one irregular crossing pattern, with recorded obstacle speeds of 0.10–0.60 m/s. The findings are limited to the geometry, surface, obstacle trajectories, map resolution, and costmap settings of this mock-up.

Automatic success required collision-free completion, autonomous avoidance or wait-and-resume recovery, and no manual takeover. Reaction time was measured from first detection to the first avoidance command, and minimum clearance was the closest recorded robot–obstacle separation. A continuous blockage longer than 15 s invoked supervised takeover and was counted as an automatic failure.

After the saved map was loaded, the smoothed A* planner generated the multi-waypoint reference path. When LiDAR detected an obstacle within the local costmap, DWA generated a temporary avoidance trajectory. If the robot returned within 0.3 m of the reference path after avoidance, it continued to follow the original path; otherwise, the global path was regenerated. The obstacle-detection distance was configured as 1.5 m for the reported mock-up.

All ten static events were handled automatically, compared with nine of ten controlled-dynamic events. Across all 20 events, the automatic success rate was 95.0% (19/20; Wilson 95% CI 76.4–99.1%), and no collision occurred. Reaction time increased from 0.303 ± 0.029 s under static conditions to 0.366 ± 0.035 s under controlled-dynamic conditions (Welch $p < 0.001$). Minimum clearance was 0.307 ± 0.028 and 0.289 ± 0.025 m, respectively, and the difference was not significant ($p = 0.144$). One continuous-blockage event exceeded 15 s and required supervised takeover. These results characterize structured test obstacles and do not establish performance for unrestricted livestock motion.

Table 3. Repeated structured-obstacle results in the physical mock-up (mean ± SD).

Condition	Events	Automatic success	Reaction time (s)	Minimum clearance (m)	Collision
Static	10	10/10 (100%)	0.303 ± 0.029	0.307 ± 0.028	0
Controlled dynamic	10	9/10 (90%)	0.366 ± 0.035	0.289 ± 0.025	0
Overall	20	19/20 (95%)	0.335 ± 0.045	0.298 ± 0.027	0

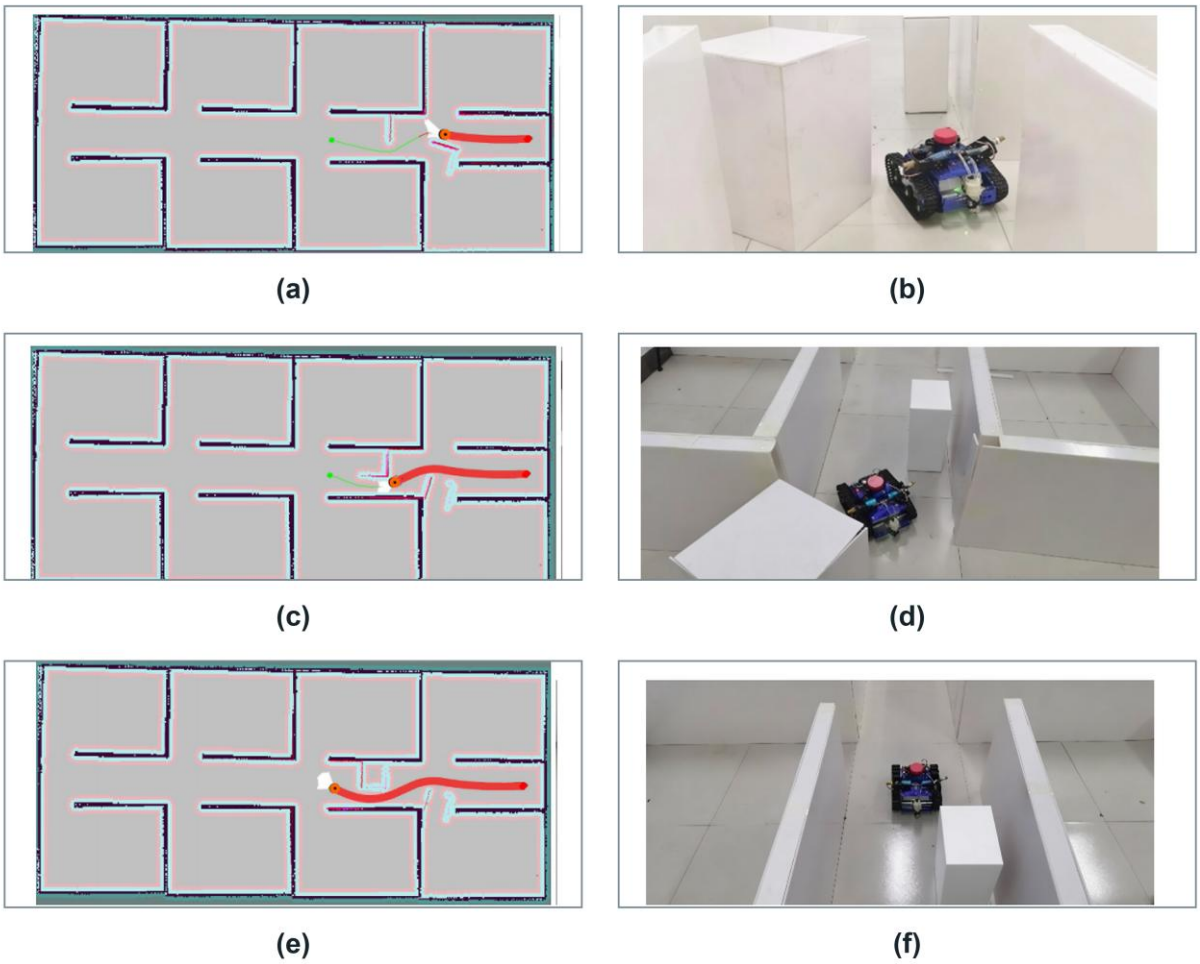

Fig. 14. Multi-waypoint navigation in the physical mock-up: (a, c, e) planned and executed trajectories; and (b, d, f) corresponding robot positions during obstacle avoidance and goal arrival.

5.2.2 Path-Coverage Planning Test

The bio-inspired, boustrophedon, and spiral strategies were each executed five times in the same 58.0 m² reachable-workspace mask. All strategies used the same obstacle inflation, termination rule, 0.60 m nominal footprint, and 20% nominal overlap, corresponding to an adjacent centerline spacing of approximately 0.48 m. This test evaluated route geometry and execution rather than biological disinfection effectiveness.

For each run, geometric coverage was calculated as covered area divided by reachable area; repeated and missed ratios used the same denominator. The bio-inspired strategy achieved 99.80 ± 0.16% coverage, 4.20 ± 0.16% repeated coverage, 0.20% mean missed area, and a duration of 78.36 ± 1.02 s. The boustrophedon strategy achieved 98.68 ± 0.19%, 12.46 ± 0.30%, 1.32%, and 92.54 ± 1.19 s, respectively; the spiral strategy achieved 99.20 ± 0.16%, 8.70 ± 0.25%, 0.80%, and 85.72 ± 0.76 s.

After Bonferroni correction, all pairwise differences in coverage remained significant (adjusted $p \leq 0.00531$). The bio-inspired strategy also had lower repeated coverage and shorter duration than both baselines (all adjusted $p < 0.001$). These inferences apply only to the common structured mock-up mask and five-run protocol.

Table 4. Five-run geometric coverage comparison in the physical mock-up (mean ± SD).

Strategy	n	Coverage (%)	Repeated (%)	Missed (%)	Duration (s)
Bio-inspired	5	99.80 ± 0.16	4.20 ± 0.16	0.20	78.36 ± 1.02
Boustrophedon	5	98.68 ± 0.19	12.46 ± 0.30	1.32	92.54 ± 1.19
Spiral	5	99.20 ± 0.16	8.70 ± 0.25	0.80	85.72 ± 0.76

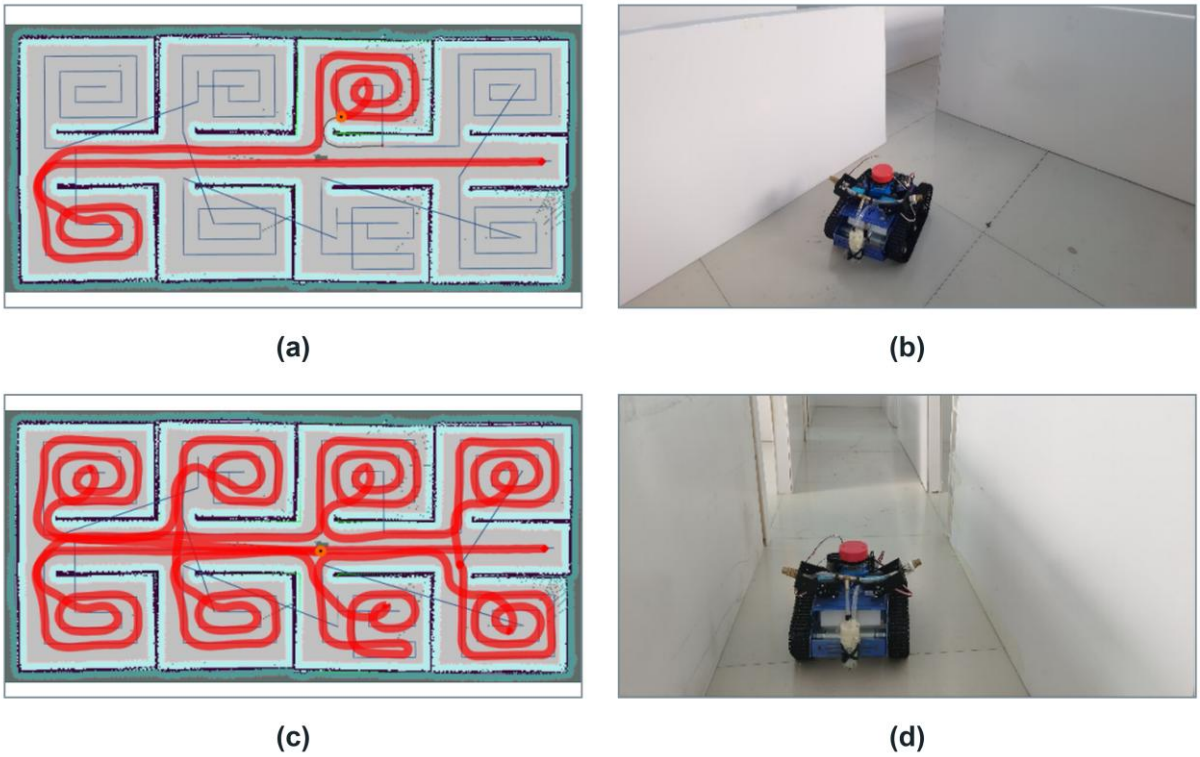

Fig. 15. Multi-waypoint and path-coverage tests in the physical mock-up: (a) planned route during continuous operation; (b) corresponding physical test; (c) coverage route; and (d) coverage-operation test.

5.2.3 Clean-Water Spray-Performance Test Protocol

A clean-water test evaluated the speed-dependent pump rule in Eq. (15). The nozzle height was 0.35 m, the target nominal application rate was 0.20 L/m², and the planned effective swath width was 0.48 m. The robot was tested at commanded speeds of 0.08, 0.11, and 0.14 m/s with five runs per speed. For each run, measured speed, discharged volume, spray-on time, geometric wetted width, and liquid collected by five transverse 100 × 100 mm trays were recorded. Actual flow was computed from discharged volume and time. Measured effective swath was calculated from the geometric wetted width using the 20% overlap, and mean tray deposition was calculated as collected liquid volume divided by tray area. The transverse coefficient of variation (CV) was the sample standard deviation divided by the mean of the five tray measurements.

Measured flow was 0.462 ± 0.008, 0.635 ± 0.009, and 0.806 ± 0.014 L/min at 0.08, 0.11, and 0.14 m/s, respectively, closely matching the theoretical commands of 0.461, 0.634, and 0.806 L/min. Flow differed among all speed pairs (Bonferroni-adjusted $p < 0.001$). In contrast, mean deposition was 0.1985 ± 0.0036, 0.2008 ± 0.0034, and 0.2006 ± 0.0033 L/m², with no pairwise difference (adjusted $p \geq 0.980$). Effective swath width remained 0.480 ± 0.002 m and transverse CV remained 5.20–5.37% across the tested range (all adjusted $p = 1.000$). Thus, the controller increased delivered flow with speed while maintaining the target deposition and transverse uniformity within the tested low-speed range.

Table 5. Clean-water spray performance at three travel speeds (n = 5 per speed; mean ± SD).

Command/measured speed (m/s)	Theoretical/actual flow (L/min)	Effective swath (m)	Deposition (L/m²)	Transverse CV (%)	Target bias (%)
0.08 / 0.080	0.461 / 0.462 ± 0.008	0.4798 ± 0.0023	0.1985 ± 0.0036	5.37 ± 0.36	−0.76
0.11 / 0.110	0.634 / 0.635 ± 0.009	0.4805 ± 0.0022	0.2008 ± 0.0034	5.20 ± 0.30	+0.40
0.14 / 0.140	0.806 / 0.806 ± 0.014	0.4802 ± 0.0015	0.2006 ± 0.0033	5.30 ± 0.15	+0.32

5.3 Preliminary Field Deployment

Five supervised work cycles were conducted in an occupied sheep house between 12 January and 18 March 2026 (Fig. 16). Their scope was basic low-speed mobility, localization continuity, route execution, stopping, and operator supervision around live animals. Runs included low- and medium-density occupied conditions; one run contained a short clean-water segment, while the remaining operations were predominantly dry-route tests.

Each cycle followed a representative aisle and pen-access route while an operator maintained direct line of sight and remote-stop authority. Route length, total duration, localization-valid time, path-tracking RMSE, maximum tracking error,

localization interruptions, animal-related blockage events, automatic responses, remote stops, collisions, and full-cycle completion were recorded.

Before each cycle, LiDAR, IMU, odometry, the latching emergency stop, and the wireless remote-stop function were checked. Disinfectant was not applied, no sheep was restrained or sampled, and the deployment was not designed as an animal-behavior or microbiological-efficacy experiment.

Across the five cycles, mean route length was 102.76 ± 0.57 m and mean duration was 1201.6 ± 29.5 s (20.03 ± 0.49 min). Localization continuity was 99.62 ± 0.23%, path-tracking RMSE was 0.0376 ± 0.0031 m, and maximum tracking error was 0.138 ± 0.018 m. The five cycles contained 1.0 ± 0.7 localization interruptions per run, each followed by continued supervised operation.

The robot handled 55 of 56 animal-related blockage events automatically (98.2%; Wilson 95% CI 90.6–99.7%). One continuous blockage in the fourth cycle required a remote stop; no collision occurred. All five cycles reached the prespecified end point (5/5).

Table 6. Summary of five supervised field work cycles in an occupied sheep house.

Metric	Result	Metric	Result
Route length	102.76 ± 0.57 m	Duration	1201.6 ± 29.5 s
Localization continuity	99.62 ± 0.23%	Tracking RMSE	0.0376 ± 0.0031 m
Maximum error	0.138 ± 0.018 m	Interruptions	1.0 ± 0.7 per cycle
Automatic responses	55/56 (98.2%)	Remote stops/collisions	1 / 0
Full-cycle completion	5/5 (100%)	Evaluation scope	Basic supervised operation

No quantitative animal-behavior dataset was analyzed. Physiological stress, longer-term behavior, body-edge separation, animal-welfare outcomes, and microbiological reduction were not measured. Accordingly, the field results establish only basic supervised system operation under the recorded conditions.

The five-cycle record adds run-level route, timing, localization, tracking, and obstacle-response measurements to the event tally. It does not establish unattended operation, animal-welfare neutrality, deposition on animal-house surfaces, or disinfection efficacy.

Fig. 16. Supervised preliminary field deployment in an occupied sheep house.

6 Discussion

6.1 Engineering Contribution and Relation to Previous Work

The principal contribution is the co-design of a complete sheep-house operating workflow: protected tracked hardware, scan-confidence management, resource-aware localization, map-to-localization transition, task–global–local navigation,

route recovery, supervised stopping, and speed-dependent liquid delivery. The clean-water results demonstrate that the pump command scaled actual flow across the tested speeds while maintaining approximately 0.20 L/m² deposition and 5.2–5.4% transverse CV. Together with repeated coverage and quantified field cycles, these measurements make the scenario adaptation assessable as a system rather than a collection of software modules.

6.2 Effect of Dynamic Disturbances on Navigation

The controlled simulation and mock-up benchmarks varied obstacle speed, trajectory, and blockage duration and therefore support conclusions for structured dynamic obstacles. The longer reaction time for controlled-dynamic obstacles ($p < 0.001$) shows the additional local-planning demand, while the non-significant clearance difference ($p = 0.144$) indicates that the configured safety margin was not measurably reduced in this small structured dataset. The 55/56 supervised field response count is consistent with stop–wait–resume operation around animal-related blockage, but it is not a livestock behavior or unattended-safety trial.

6.3 Path Coverage, Spray Deposition, and Disinfection Efficacy

Geometric footprint coverage, liquid deposition, and microbiological reduction remain successive evidence levels. The repeated route test established geometric performance, and the three-speed clean-water test added direct external measurements of actual flow, effective swath, and transverse deposition. Significant flow increases accompanied stable deposition (adjusted $p \geq 0.980$), effective width, and CV, supporting the speed–flow rule over 0.08–0.14 m/s. These findings do not quantify surface-specific retention, droplet-size distribution, drift, chemical activity, or microbial reduction.

6.4 Limitations and Future Improvements

The remaining limitations are concentrated in four areas. First, coverage and spray tests used five repetitions per group in one mock-up, so uncertainty across facilities and floor conditions is unknown. Second, the obstacle test used structured objects and only 20 events; the results cannot be generalized to unrestricted flock motion. Third, the field dataset contains five supervised cycles in one sheep house and does not establish unattended reliability. Fourth, microbiological reduction, animal-welfare effects, droplet-size distribution, and surface-specific deposition were outside the study scope.

Future work should extend the completed low-cost spray and navigation tests to multiple sheep houses, seasons, floor conditions, flock densities, and longer operating periods; add a calibrated onboard flow sensor and an explicit slip observer or LiDAR–inertial odometry comparison; and evaluate surface-specific deposition before undertaking microbiological reduction studies. Animal-behavior and welfare studies should use a separately approved protocol with prespecified independent runs.

7 Conclusions and Future Work

This study developed a sheep-house-specific tracked spraying robot and a three-level ROS navigation framework. The selected mapping configuration achieved 96.8 ± 0.8% IoU, AMCL achieved 0.0299 ± 0.0026 m RMSE, and the integrated planner completed 80/80 specified simulation trials. In five-run physical comparisons, the bio-inspired strategy achieved 99.80 ± 0.16% geometric coverage with 4.20 ± 0.16% repeated coverage. Fifteen clean-water runs showed that actual flow increased from 0.462 ± 0.008 to 0.806 ± 0.014 L/min as travel speed increased from 0.08 to 0.14 m/s, while measured deposition remained 0.1985–0.2008 L/m² and transverse CV remained 5.20–5.37%. Five supervised field cycles achieved 99.62 ± 0.23% localization continuity and 0.0376 ± 0.0031 m tracking RMSE, with 55/56 animal-related blockage events handled automatically.

The results demonstrate the engineering feasibility of the integrated sheep-house navigation, coverage, and hydraulic-delivery workflow under the reported test conditions. The contribution is a measurable scenario-specific system integration and validation dataset, not a new standalone SLAM or planning algorithm. Microbiological disinfection efficacy, animal-welfare effects, and unattended long-term operation require separate future studies.

Acknowledgments

This work was financially supported by the Key Research and Development Special Project of Henan Province (251111110400), the Scientific and Technological Breakthrough Foundation of Henan Province (262102320256), and the Key Research Project of Higher Education Institutions in Henan Province (25A220004, 25B460006).

Data Availability Statement

Processed data and retained configuration summaries supporting the reported descriptive results are available from the corresponding author upon reasonable request. Complete raw logs are not available for all legacy configuration comparisons, as stated in the Methods and Results.

Ethics Statement

The field component involved non-invasive observation of equipment operation under normal farm conditions. Prior permission to conduct the five supervised field work cycles in the occupied sheep house was obtained from the manager responsible for the sheep house at Henan Agricultural University. No sheep was restrained, sampled, dosed, deliberately sprayed, surgically treated, or subjected to a prespecified behavioral intervention, and direct supervisory stopping authority was maintained throughout.

Declaration of Generative AI and AI-Assisted Technologies in the Writing Process

During preparation of this work, the authors used ChatGPT (OpenAI) to assist with English-language editing, structural revision, and draft figure layouts and captions. The authors subsequently reviewed and edited the content and take full responsibility for the publication.

References

[1] Banjanovic-Mehmedovic, L, I Karabegovic, J Jahic, and M Omercic. 2021. "Optimal Path Planning of a Disinfection Mobile Robot Against COVID-19 in a ROS-Based Research Platform." *Proceedings of the Conference on Automation Science and Engineering (CASE), Lyon, France*, 23–27.

[2] Chang, Zhanyuan, Honglin Wu, and Chuanjiang Li. 2023. "YOLOv4-Tiny-Based Robust RGB-d SLAM Approach with Point and Surface Feature Fusion in Complex Indoor Environments." *Journal of Field Robotics* 40 (3): 521–34.

[3] Chen, Lifang, Huogen Yang, Zhichao Chen, and Zhicheng Feng. 2023. "Research on Intelligent Disinfection-Vehicle System Design and Its Global Path Planning." *Electronics* 12 (7): 1514.

[4] Department, Editorial, and Xu Wang. 2022. "The 14th Five-Year Plan for the Development of Agricultural Mechanization in China." *China Animal Husbandry*.

[5] Feng, Qing Chun, and Xiu Wang. 2020. "Design of Disinfection Robot for Livestock Breeding." *Procedia Computer Science* 166: 310–14.

[6] Feng, Qing-chun, Xiu Wang, Quan Qiu, et al. 2021. *Design and Test of Disinfection Robot for Livestock and Poultry House.*

[7] Feng, Qingchun, Bowen Wang, Wanhao Zhang, and Xiaoming Li. 2021. "Development and Test of Spraying Robot for Anti-Epidemic and Disinfection in Animal Housing." *2021 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*, 24–29.

[8] Jeong, Wonseok, Chanho Lee, Namyeong Lee, Seungwoo Hong, Donghyun Kang, and Donghyeok An. 2025. "Improving Sensor Adaptability and Functionality in Cartographer Simultaneous Localization and Mapping." *Sensors* 25 (6): 1808.

[9] Li, Hongwei. 2016. "Key Points of Epidemic Prevention Technology in Large-Scale Mutton Sheep Farms." *Journal of Animal Husbandry and Veterinary Medicine Information*, no. 6: 55.

[10] Liu, Xintong, Gu Gong, Xiaoting Hu, Gongyu Shang, and Hua Zhu. 2024. "Cognitive Enhancement of Robot Path Planning and Environmental Perception Based on Gmapping Algorithm Optimization." *Electronics* 13 (5): 818.

[11] Ma, Ye, Ning Xi, Yuxuan Xue, Siyu Wang, Qingyang Wang, and Ye Gu. 2022. "Development of a UVC-Based Disinfection Robot." *Industrial Robot: The International Journal of Robotics Research and Application* 49 (5): 913–23.

[12] Peavy, Matthew, Pileun Kim, Hafiz Oyediran, and Kyungki Kim. 2023. "Integration of Real-Time Semantic Building Map Updating with Adaptive Monte Carlo Localization (AMCL) for Robust Indoor Mobile Robot Localization." *Applied Sciences* 13 (2): 909.

[13] Singh, Raj Kumar, and KS Nagla. 2024. "Reduction in Trajectory Error by Generating Smoother Trajectory for the Time-Efficient Navigation of Mobile Robot: RK Singh, KS Nagla." *MAPAN* 39 (3): 687–99.

[14] Sun, Jie. 2025. "Research on the Impact of Digital Economy on the Upgrading of Animal Husbandry Industrial Structure." *Feed Research* 48 (8).

[15] Wang, Haotian, Xiaolong Zhou, Jianyong Li, Zhilun Yang, and Linlin Cao. 2024. "Improved RRT* Algorithm for Disinfecting Robot Path Planning." *Sensors* 24 (5): 1520.

[16] Xu, Jun. 2025. "Construction and Management Optimization of Animal Epidemic Prevention and Control System in Large-Scale Farms." *Medicine and Public Health* 1 (7).

[17] Yao, Zifan, Na Ma, and Youdong Chen. 2023. "An Autonomous Mobile Combination Disinfection System." *Sensors* 24 (1): 53.

[18] Zhang, Chenyu, Haohao Qin, Shiying Sun, et al. 2024. "JPMDP: Joint Base Placement and Multi-Configuration Path Planning for 3D Surface Disinfection with a UV-c Robotic System." *Robotics and Autonomous Systems* 174: 104644.

[19] Zhao, Yongyu. 2016. "Disease Prevention and Control Technology for House-Feeding Sheep." *Abstracts of Chinese Animal Husbandry and Veterinary Medicine*, no. 3: 123.

[20] Zhou, Luyang, Linqiu Hao, Yingjun Xiong, Huanhuan Qin, Aoran Bao, and Zikang Chen. 2026. "Research Progress of Robotic Technologies and Applications in Smart Pig Farms." *Agriculture* 16 (3): 334.