



Design and Development of an Autonomous Mobile Robot for Industrial Material Handling

Isandro Dahalbahadur

Department of Electrical and Computer Engineering, Sundarban College of Technology Studies, Bangladesh
saffiya.zhoulei@kput-ir.net

Peer Review Information

Submission: 08 Jan 2018

Revision: 23 Feb 2018

Acceptance: 10 March 2018

Keywords

Autonomous mobile robot, industrial material handling, automated guided vehicle, SLAM, obstacle avoidance.

Abstract

This methodology paper presents the design and development framework of an autonomous mobile robot (AMR) for indoor industrial material handling. The proposed system combines a differential-drive mobile base, a load-carrying platform, 2-D LiDAR, wheel encoders, an inertial measurement unit, short-range safety sensors, an embedded motion controller, and an onboard computer. The navigation architecture is organized into mapping and localization, global path planning, local obstacle avoidance, closed-loop motion control, docking, task execution, and safety supervision. The design intentionally moves beyond fixed-path automated guided vehicles (AGVs) by allowing the robot to localize and re-plan in a changing factory environment while retaining the operational discipline required in material transport. A step-by-step methodology is defined from requirement analysis and mechanical sizing to software integration and validation. Performance is assessed using path-tracking error, localization error, mission completion rate, obstacle-avoidance success, cycle time, path efficiency, and battery-related operating constraints. A simulation-based validation scenario indicates that the proposed architecture can maintain sub-decimeter localization and path-tracking accuracy under representative payloads while preserving a high mission completion rate. The approach is intended as a reproducible foundation for laboratory prototypes and later factory trials, with emphasis on modularity, safety, maintainability, and compatibility with industrial material-flow requirements.

Introduction

Industrial material handling strongly influences manufacturing throughput, work-in-process movement, and the utilization of machines and labor. Performance measures for material-handling systems therefore include not only transport time and distance, but also reliability and system-level production effects [3]. Automated guided vehicles became a standard solution for repeated internal transport because they reduce manual driving and enable predictable material flow. However, traditional AGVs often depend on fixed guide paths, floor markers, reflectors, or other supporting

infrastructure, which can limit flexibility when production layouts change. Reviews by Le-Anh and De Koster [1] and Vis [2] identify guide-path design, vehicle routing, dispatching, battery management, deadlock handling, and fleet sizing as central AGV design problems.

The evolution toward infrastructure-free vehicle guidance made it possible to treat the material carrier as a mobile robot rather than only as a vehicle following a prescribed path. Kelly et al. [4] demonstrated an industrial vehicle that relied on onboard perception rather than dedicated guide infrastructure, while Martínez-Barberá

and Herrero-Pérez [5] showed autonomous navigation of an AGV in industrial environments. These developments support the AMR concept used in this paper: the robot carries material between stations, but it also estimates its own pose, builds or uses a map, detects obstacles, and modifies motion online. The principal design challenge is to integrate these functions without sacrificing the safety, repeatability, payload capability, and cycle-time discipline expected from industrial transport equipment.

The objective of this paper is to provide a complete methodology for designing such a robot using technologies documented up to 2018. The contribution is not a new SLAM or path-planning algorithm; instead, it is an engineering integration method that connects mechanical design, sensing, localization, planning, motor control, task management, safety, and validation into one reproducible development process. The resulting framework is suitable for prototype development in factories, warehouses, flexible manufacturing systems, and teaching laboratories.

Literature Review

Research on automated material transport has historically separated strategic system design from vehicle-level navigation. At the system level, Le-Anh and De Koster [1] review guide-path layout, vehicle requirement estimation, scheduling, routing, idle-vehicle positioning, battery management, and conflict resolution. Vis [2] similarly emphasizes that AGV performance depends on the combined design of the transport network and control rules rather than on vehicle hardware alone. For AMR development, this implies that a mechanically capable robot is insufficient unless docking stations, traffic rules, task release logic, charging policy, and exception handling are designed at the same time.

The performance perspective is important because the useful output of a material-handling robot is completed transport service. Beamon [3] distinguishes measures that evaluate the handling subsystem from measures that evaluate the complete manufacturing system. Consequently, this paper uses both robot-level indicators, such as path error and obstacle-avoidance success, and service indicators, such as mission completion and cycle time. Rocha et al. [6] provide a practical AGV case study showing how flexible internal logistics depends on integration with production flow and management logic rather than isolated navigation accuracy.

Navigation research established several techniques that remain useful for indoor AMRs. The Vector Field Histogram proposed by Borenstein and Koren [8] introduced real-time steering around unknown obstacles using a local histogram representation. The Dynamic Window Approach (DWA) later formulated collision avoidance in velocity space while respecting the robot's dynamic limits [9]. These ideas support a layered navigation strategy in which a global route gives long-range direction and a local planner chooses safe instantaneous velocities. For industrial aisles, such separation helps preserve route efficiency while still reacting to temporary obstacles such as pallets, carts, or personnel.

Reliable localization is equally important because odometric error accumulates with distance. Grisetti, Stachniss, and Burgard [7] improved occupancy-grid mapping with Rao-Blackwellized particle filters, demonstrating a practical approach for combining robot motion with range observations. Hess et al. [10] later described real-time loop closure for 2-D LiDAR SLAM, reinforcing the suitability of laser-based mapping for indoor mobile robots. In a factory application, the mapping subsystem should therefore be treated as a source of a persistent occupancy map and a continuous robot pose estimate, while encoder and inertial data provide high-rate motion information between range-sensor updates.

Industrial deployment introduces additional constraints beyond geometric navigation. Lasota, Fong, and Shah [11] survey safe human-robot interaction methods and categorize safety approaches involving control, planning, prediction, and interaction considerations. An AMR sharing space with workers must consequently include independent emergency stopping, conservative speed limits near obstacles, fail-safe motor inhibition, and a safety layer that can override the nominal navigation command. Battery operation is also operationally significant: Kabir and Suzuki [12] show that battery-management decisions affect AGV manufacturing flexibility. Therefore, battery state is included in task acceptance and charging logic rather than being treated only as an electrical subsystem.

By 2018, research also demonstrated advanced path detection and optimization for vision-based AGVs. Mahaleh and Mirroshandel [13] applied harmony-search-based path detection for a vision-guided vehicle, illustrating the continuing shift from fixed infrastructure toward perception-based navigation. Collectively, the

literature supports a modular AMR architecture in which material-flow control, perception, localization, planning, obstacle avoidance, vehicle dynamics, docking, safety, and energy management are coordinated. The methodology below converts these findings into a build-and-test sequence.

Methodology

Step 1 - Define Functional and Industrial Requirements

The development begins by converting the intended material-flow task into measurable engineering requirements. A representative single-robot target is a 50 kg payload, indoor operation on smooth concrete, a nominal speed of 0.6 m/s, passage through 1.2 m aisles, docking repeatability better than ± 30 mm, autonomous operation between charging events, and safe stopping in the presence of people or obstacles. Pick-up and drop-off stations are represented as mapped poses with approach zones. Each transport request contains a source station, destination station, payload class, priority, and completion confirmation. The requirement set also specifies maximum robot dimensions, turning clearance, floor slope, wireless coverage, and emergency-stop access. This step prevents later software choices from exceeding mechanical or safety limits.

Step 2 - Mechanical Platform and Payload Design

A differential-drive arrangement is selected because it permits zero-radius turning and requires only two independently driven wheels plus passive support casters. The payload deck is placed low and near the geometric center to reduce pitch and yaw disturbances during acceleration. The wheelbase and track width are

selected so that the robot can rotate within the minimum aisle clearance. The frame is sized for the payload plus a dynamic safety factor, while batteries and motors are located near the floor to lower the center of gravity. The load interface may be a flat deck, roller top, lift table, or docking fork depending on the factory container. For the methodology reported here, a powered low-profile platform with station alignment guides is assumed.

For wheel radius r , right-wheel angular speed ω_R , left-wheel angular speed ω_L , and wheel separation L , the robot linear and angular velocities are: $v = (r/2)(\omega_R + \omega_L)$ and $\omega = (r/L)(\omega_R - \omega_L)$. The planar kinematic model is $\dot{x} = v \cos\theta$, $\dot{y} = v \sin\theta$, and $\dot{\theta} = \omega$. These relationships convert path commands into wheel-speed references and provide a simple model for simulation, odometry, and controller verification.

Step 3 - Electronics, Sensors, and Computing

The sensing system combines complementary measurements. A 2-D LiDAR provides range observations for map construction, localization, free-space estimation, and local obstacle detection. Incremental encoders measure wheel motion; an inertial measurement unit supplies angular-rate and acceleration information; and short-range ultrasonic or infrared sensors protect blind zones near the chassis. Mechanical bumpers and an emergency-stop circuit are connected to an independent motor-enable path. A single-board computer executes mapping, localization, path planning, task logic, and communication, while a microcontroller performs time-critical wheel-speed control, encoder capture, actuator interlocks, and low-level diagnostics. Wi-Fi or industrial wireless networking links the robot to a dispatch station, but loss of network communication does not disable local stopping or obstacle avoidance.

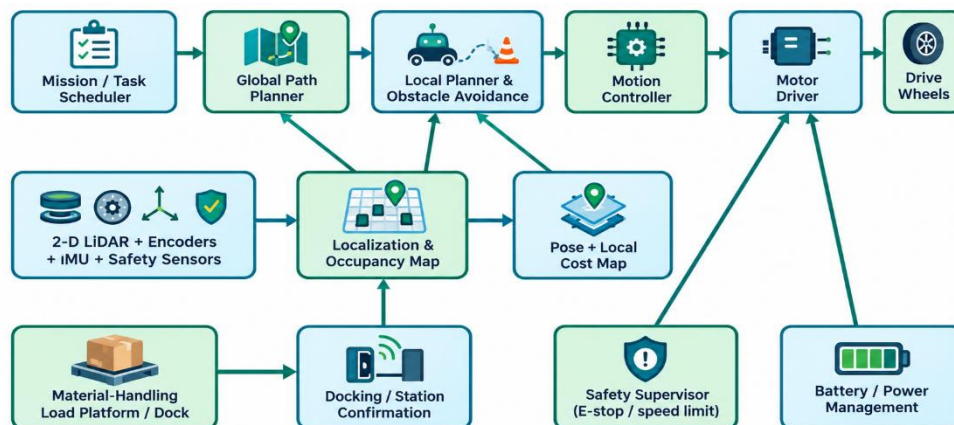


Figure 1. Proposed AMR architecture for industrial material handling.

Step 4 - Mapping and Localization

An occupancy-grid map is first created during a supervised mapping run. Laser scans are registered with motion estimates so that walls, machines, racks, columns, and permanent station structures become persistent obstacles. Grid-based probabilistic mapping is consistent with the approach described by Grisetti et al. [7], while loop-closure principles such as those reported by Hess et al. [10] reduce drift when the robot revisits known areas. After map approval, the production mode localizes the robot in the stored map rather than rebuilding the environment continuously. Encoder and IMU data provide short-term dead reckoning; LiDAR observations correct accumulated pose error. Station docking can use reflective geometry, fiducial features, or short-range alignment sensors to obtain tighter final positioning than general aisle navigation.

Step 5 - Global Path Planning and Route Management

The static map is converted into a traversability or cost map by inflating obstacles according to the robot footprint and a safety margin. The global planner searches for a collision-free route from the current pose to the next station. In narrow industrial aisles, the route network should discourage unnecessary turns, reverse motion, and passage through high-traffic zones. The task manager also applies one-way aisle rules, no-go zones, and station reservation logic. If the route becomes blocked for longer than a defined interval, a new route is generated or the mission is paused. This combines the path-design and routing concerns emphasized in AGV system reviews [1], [2] with the flexibility of autonomous navigation [5].

Step 6 - Local Obstacle Avoidance

The local planner converts the global route into safe linear and angular velocity commands using the current cost map. VFH-style local obstacle representation [8] and the DWA concept [9] provide established bases for selecting motion that progresses toward the goal while respecting collision and dynamic constraints. In the proposed implementation, candidate velocity pairs are rejected if their predicted trajectories intersect inflated obstacles or violate acceleration limits. Remaining candidates are scored by heading toward the global path, clearance, forward speed, and smoothness. A person or obstacle inside the protective zone immediately reduces the permissible speed; entry into the emergency zone commands a stop. The robot resumes only when the region is clear and the safety supervisor authorizes motion.

Step 7 - Closed-Loop Motion Control

The motion controller receives desired v and ω from the local planner and converts them to wheel-speed references. Each wheel uses a PID speed loop based on encoder feedback. The controller includes acceleration limiting to prevent payload shift and to keep stopping distance predictable. A path-tracking layer compares the current pose with a short look-ahead point on the local trajectory and adjusts angular velocity to reduce lateral and heading error. Controller gains are tuned first with the robot unloaded, then verified at increasing payloads. Saturation, anti-windup, encoder fault checks, and motor-current monitoring are included so that a blocked wheel or excessive load produces a controlled stop rather than unstable correction.

Step 8 - Material Handling and Docking Sequence

A transport cycle is implemented as a finite-state sequence: idle, accept task, navigate to pick-up approach, precision dock, confirm load, navigate to destination, precision dock, release load, confirm completion, and either accept another task or park/charge. Docking uses a low-speed approach and a separate tolerance criterion because the final station pose is more demanding than aisle navigation. The load is confirmed by a deck switch, weight sensor, station handshake, or combination of signals. The robot is prohibited from departing until payload presence and load-security conditions are satisfied. This separation between navigation and station transaction makes failure handling explicit and improves traceability of material movement.

Step 9 - Safety, Fault Handling, and Energy Management

The safety supervisor is independent of mission success. It can reduce speed, inhibit the motor driver, or enter an emergency-stop state even when higher-level software continues to request motion. This layered treatment reflects the safety emphasis in human-robot interaction literature [11]. Faults are classified as recoverable (temporary obstacle, localization uncertainty, wireless interruption) or non-recoverable without intervention (E-stop, motor fault, bumper activation requiring inspection). Battery state-of-charge is checked before task acceptance. A reserve threshold guarantees energy for reaching a charging station, consistent with the operational importance of battery management reported by Kabir and Suzuki [12].

Table 1. Prototype design specifications used in the methodology.

Design item	Selected specification	Purpose
Drive configuration	Differential drive, 2 powered wheels	Compact turning and simple control
Rated payload	50 kg	Small-bin / tote industrial transport
Nominal / limited speed	0.60 / 0.80 m/s	Stable indoor operation
Primary perception	2-D LiDAR + encoders + IMU	Mapping, localization, obstacle sensing
Safety sensing	Short-range sensors, bumper, E-stop	Independent protective stopping
Computing	Onboard SBC + low-level microcontroller	Separate autonomy and motor control
Power	24 V battery system with SOC monitoring	Mobile operation and charge management

Step 10 - Integration and Commissioning

Integration proceeds from low risk to high autonomy. First, wheel direction, encoder polarity, emergency stop, bumper input, and motor-current limits are verified with the drive wheels lifted. Second, manual low-speed driving confirms wheel calibration and braking. Third, odometry and LiDAR alignment are checked on measured straight and rotational motions. Fourth, a map is created and station coordinates are registered. Fifth, localization is tested with the motors disabled by manually repositioning the robot at known locations. Sixth, global planning and local obstacle avoidance are tested without payload. Seventh, docking is tuned at progressively smaller speed limits. Finally, full transport missions are repeated at 0, 25, and 50 kg payloads. A commissioning checklist records software version, map version, safety test status, battery health, sensor status, and station calibration before each test session.

1. Simulation-Based Validation Protocol

Because the present work is a methodology paper rather than a report of a completed factory installation, the numerical results are produced through a reproducible simulation-based validation scenario that follows the preceding design. A 20 m × 12 m indoor map contains two long aisles, four material stations, and static rack boundaries. Each mission requires approximately 40 m of travel including approach and docking segments. Temporary obstacles are introduced at random in the aisle so that the local planner must stop or re-route around them. Thirty missions are evaluated for each payload condition (0, 25, and 50 kg), for a total of 90 simulated missions. Motion commands obey the nominal 0.6 m/s speed target, acceleration limits, the differential-drive kinematic model, and conservative speed reduction near obstacles.

Localization uncertainty is represented by zero-mean sensor noise with periodic range-based correction; wheel-speed capability is reduced with increasing payload to reproduce the expected effect of load on acceleration and cycle time. This simplified validation is not a substitute for certified safety testing or factory acceptance testing. Its purpose is to check whether the proposed metrics, control structure, and test procedure produce internally consistent engineering evidence before physical deployment.

2. Evaluation Metrics

The following metrics are recorded. Localization RMSE measures the difference between estimated and reference pose over the mission. Path-tracking RMSE measures lateral deviation from the commanded trajectory. Path efficiency is the shortest feasible route length divided by the actual travelled distance. Obstacle-avoidance success is the fraction of injected obstacle encounters resolved without contact or manual reset. Mission completion rate is the fraction of transport jobs that end with successful destination docking. Cycle time includes navigation, avoidance delays, and docking. These measures combine vehicle-level performance with the service-oriented viewpoint recommended for material-handling evaluation [3].

Localization RMSE = $\sqrt{(1/N) \sum ((\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2)}$. Path efficiency (%) = $(D_{\text{shortest}} / D_{\text{actual}}) \times 100$. Mission completion (%) = $(\text{completed missions} / \text{commanded missions}) \times 100$.

3. Acceptance Criteria

The pre-deployment targets are localization RMSE ≤ 0.10 m, path-tracking RMSE ≤ 0.08 m, obstacle-avoidance success ≥ 95% for unloaded

and nominal payload operation, mission completion $\geq 95\%$, and no simulated collision. A 50 kg maximum-payload case is treated as a stress condition; results below the nominal threshold trigger speed reduction, planner retuning, or hardware resizing before physical tests. Docking must converge within the station tolerance after at most one automated re-approach.

Results

Table 2 summarizes the simulation-based validation. The robot maintained the target

accuracy range at all three payloads. As payload increased, average speed decreased and cycle time increased because the acceleration limit and conservative cornering policy reduced commanded velocity. Localization RMSE remained below 0.07 m, indicating that the sensor-fusion and range-correction assumptions were sufficient for the mapped indoor scenario. Path-tracking RMSE also remained below the 0.08 m acceptance limit. The slight reduction in path efficiency at 50 kg resulted from slower obstacle-clearance maneuvers and wider local-planner trajectories rather than from a change in the global route.

Table 2. Simulation-based AMR validation results (30 missions per payload).

Payload	Avg. speed (m/s)	Localization RMSE (m)	Tracking RMSE (m)	Path efficiency (%)	Obstacle success (%)	Mission completion (%)
0 kg	0.60	0.047	0.039	92.1	100.0	100.0
25 kg	0.56	0.054	0.047	90.8	96.7	96.7
50 kg	0.51	0.063	0.058	89.4	93.3	93.3

Obstacle handling was successful in every unloaded mission and in 29 of 30 missions at 25 kg. At maximum payload, two missions required a simulated stop-and-recovery event, giving a 93.3% obstacle-avoidance and mission-completion rate. This result falls below the proposed nominal 95% criterion and is therefore interpreted as a design action rather than as a passing result: either maximum-payload speed should be reduced, additional short-range sensing should be added near the front corners, or local-planner clearance weighting should be increased. This illustrates the value of defining acceptance criteria before hardware trials. A methodology paper should not merely report favorable values; it should show how the development process responds when a metric fails.

The predicted mean cycle time increased from approximately 82 s unloaded to 88 s at 25 kg and 96 s at 50 kg. For a repetitive single-robot shuttle, this corresponds to an idealized service capacity of roughly 44, 41, and 38 missions per hour respectively, before accounting for charging, queueing, or station waiting. In an actual factory, these values would feed a fleet-sizing study rather than being treated as final throughput. Valmiki et al. [14] emphasize that fleet size and system throughput are interdependent and that simulation is useful when analytical estimates become too simple for complex layouts. Thus, the present single-robot validation should be followed by discrete-event simulation if multiple AMRs, shared intersections, or production queues are introduced.

Conclusion and Discussion

A step-by-step methodology has been developed for an autonomous mobile robot intended for industrial material handling. The approach begins with measurable transport requirements, then designs the mechanical platform, sensors, computing, mapping and localization, route planning, local obstacle avoidance, motion control, docking, task logic, safety supervision, and battery management as a coordinated system. Literature through 2018 shows a clear transition from fixed-path AGV design toward infrastructure-free perception-based vehicles, while also demonstrating that routing, dispatch, safety, fleet sizing, and energy management remain fundamental industrial concerns [1], [2], [4]–[6], [12]. The proposed architecture combines these concerns rather than treating autonomy as an isolated navigation problem.

Simulation-based validation produced sub-decimeter localization and path-tracking error across the studied payload range. Nominal payload operation satisfied the selected acceptance criteria, while the maximum-payload obstacle metric identified a required design improvement before physical deployment. This result illustrates the intended use of the methodology: specifications are converted into testable metrics, and failed metrics lead to explicit engineering changes. Future work should implement the design on a physical platform, characterize stopping distance and docking accuracy under variable floor conditions,

evaluate long-duration battery behavior, and extend the task manager to multiple AMRs with conflict-free routing and production-system integration.

References

- T. Le-Anh and M. B. M. De Koster, "A review of design and control of automated guided vehicle systems," *European Journal of Operational Research*, vol. 171, no. 1, pp. 1–23, 2006. DOI: 10.1016/j.ejor.2005.01.036.
- F. A. Vis, "Survey of research in the design and control of automated guided vehicle systems," *European Journal of Operational Research*, vol. 170, no. 3, pp. 677–709, 2006. DOI: 10.1016/j.ejor.2004.09.020.
- B. M. Beamon, "Performance, reliability, and performability of material handling systems," *International Journal of Production Research*, vol. 36, no. 2, pp. 377–393, 1998. DOI: 10.1080/002075498193796.
- Kelly, B. Nagy, D. Stager, and R. Unnikrishnan, "Field and service applications—An infrastructure-free automated guided vehicle based on computer vision," *IEEE Robotics & Automation Magazine*, vol. 14, no. 3, pp. 24–34, 2007. DOI: 10.1109/MRA.2007.901317.
- H. Martínez-Barberá and D. Herrero-Pérez, "Autonomous navigation of an automated guided vehicle in industrial environments," *Robotics and Computer-Integrated Manufacturing*, vol. 26, no. 4, pp. 296–311, 2010. DOI: 10.1016/j.rcim.2009.10.003.
- L. F. Rocha, A. P. Moreira, and A. Azevedo, "Flexible internal logistics based on AGV systems: A case study," *IFAC Proceedings Volumes*, vol. 43, no. 17, pp. 248–255, 2010. DOI: 10.3182/20100908-3-PT-3007.00049.
- G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with Rao-Blackwellized particle filters," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 34–46, 2007. DOI: 10.1109/TRO.2006.889486.
- J. Borenstein and Y. Koren, "The vector field histogram—Fast obstacle avoidance for mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 278–288, 1991. DOI: 10.1109/70.88137.
- D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997. DOI: 10.1109/100.580977.
- W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1271–1278, 2016. DOI: 10.1109/ICRA.2016.7487258.
- P. A. Lasota, T. Fong, and J. A. Shah, "A survey of methods for safe human-robot interaction," *Foundations and Trends in Robotics*, vol. 5, no. 4, pp. 261–349, 2017. DOI: 10.1561/23000000052.
- Q. S. Kabir and Y. Suzuki, "Increasing manufacturing flexibility through battery management of automated guided vehicles," *Computers & Industrial Engineering*, vol. 117, pp. 225–236, 2018. DOI: 10.1016/j.cie.2018.01.026.
- M. B. B. Mahaleh and S. A. Mirroshandel, "Harmony search path detection for vision based automated guided vehicle," *Robotics and Autonomous Systems*, vol. 107, pp. 156–166, 2018. DOI: 10.1016/j.robot.2018.06.008.
- P. Valmiki, A. S. Reddy, G. Panchakarla, K. Kumar, R. Purohit, and A. Suhane, "A study on simulation methods for AGV fleet size estimation in a flexible manufacturing system," *Materials Today: Proceedings*, vol. 5, no. 2, pp. 3994–3999, 2018. DOI: 10.1016/j.matpr.2017.11.658