A distributed ROS2 architecture for low-cost navigation based on virtual boundary logic

Sergio García González[1[0000−0002−0338−0817]], David Cruz García[1[0000−0002−1922−7040]], Arturo Álvarez Sánchez[1[0009−0001−7705−6139]], and Gabriel Villarrubia González[1[0000−0002−6536−2251]]

Expert Systems and Applications Laboratory (ESALAB), Faculty of Science, University of Salamanca, Salamanca, Spain
sergio.gg@usal.es

Abstract. This article presents the design and implementation of a low-cost modular autonomous navigation system for conventional wheelchairs, focusing on a distributed architecture. Unlike traditional centralised systems, the proposed solution leverages the ROS2 Data Distribution Service (DDS) protocol to establish dynamic node discovery between an external processing unit and a local unit. This approach allows SLAM and complex trajectory planning tasks to be delegated to an external computing station, while the local unit manages the acquisition of data from the LD-19 LiDAR and the control of the actuators in real time using an STM32 board. The navigation logic employs a sequential waypoint approach in which a boundary system is implemented to prevent skipping points and maintain a straight trajectory. The results validate that this distributed configuration ensures low latency and high scalability, offering a robust and economical technical alternative for transforming standard mobility devices into intelligent assistance platforms.

Keywords: Autonomous navigation · Assistive robotics · ROS2 · DDS · Distributed Computing · Low-cost robotics · Embedded systems

1 Introduction

Autonomous mobility is an essential requirement for participation in society and the psychological well-being of individuals. However, demographic change in Europe poses an urgent challenge to social and healthcare support systems. According to official statistical reports from Eurostat, population ageing is a structural trend in the European Union. It is projected that the proportion of people aged 80 and over will multiply by 2.5 by the year 2100, reaching 14.6% of the total population [1]. This scenario foreshadows an imminent saturation of the human resources available for the care and transport of people with reduced mobility.

To mitigate this pressure, the scientific community has consistently focused on autonomous system solutions, such as the development of smart wheelchairs. Despite significant advances reviewed in recent literature, the actual adoption

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_24

of these technologies in clinical and domestic environments remains marginal [2]. The primary barriers to widespread implementation are the high cost of commercial robotic platforms and the reliance on centralised and proprietary hardware architectures that hinder maintenance and adaptation.

The current technical challenge lies in endowing low-cost devices possessing limited computational resources with advanced autonomy capabilities (such as localisation, mapping, and dynamic navigation). State-of-the-art reviews indicate that the transition from classic robotic systems to modular architectures based on ROS2 is critical for achieving the necessary interoperability in modern service robots [3,4].

However, executing robust navigation algorithms entails a high computational cost that rapidly depletes the batteries of standard embedded systems. Recent research has demonstrated that, for energy-constrained robots, the optimal solution is computation offloading, delegating intensive tasks to external nodes via the network [5]. Despite this evidence, many current low-cost solutions continue to opt for monolithic architectures. Recent studies on autonomous guidance validate the use of 2D LiDAR sensors in wheelchairs but highlight latency and safety issues when all processing is forced onto local hardware [6].

Aligned with these trends, this work proposes a distributed architecture based on ROS2 that overcomes these limitations. Our main contribution is a modular system that integrates hardware, including the reuse of low-cost personal mobility motors [7], via the DDS protocol. This approach enables the execution of complex navigation algorithms by offloading the computational burden to a remote device, while retaining only the reactive control logic on the wheelchair to provide a robust, safe, and economically accessible assistive solution.

2 Architecture

The platform design was conceived under a modular approach, prioritising the integration of commercial or recycled components to guarantee replicability. The hardware is structured around a distributed topology that interconnects a reused traction platform from a *hoverboard* with 350 W BLDC motors, an STM32F103-based controller with FOC firmware [7], a local Raspberry Pi 4 unit in charge of hardware abstraction, two solid-state LD-19 LiDAR sensors (12m, 10Hz), and a 36V Li-Ion battery powering the entire assembly.

As detailed in Figure 1, the communication architecture implements a hierarchical topology designed to isolate real-time actuation control from high-level navigation logic. The Raspberry Pi 4 operates as a link node (*Edge Gateway*), providing data acquisition and command distribution among the subsystems.

At the physical level, the central processing unit manages environment perception via a direct USB connection with the LiDAR sensor, ensuring sufficient bandwidth for the transmission of high-frequency point clouds. Simultaneously, an asynchronous serial link (UART) is established at 115200 baud with the STM32 microcontroller, which acts as the power stage controller. Through this channel, the Raspberry Pi transmits the linear (v) and angular (ω) velocity vec-

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_24

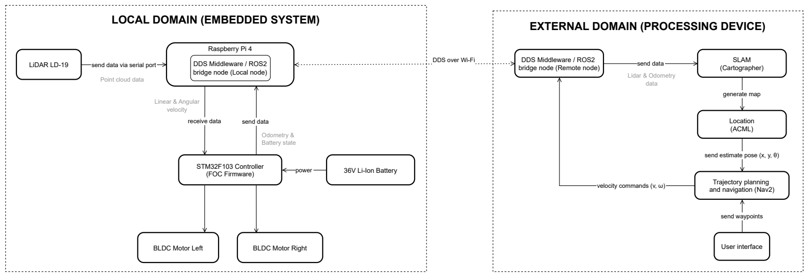

Fig. 1. Block diagram of the system architecture.

tors, which the microcontroller processes to regulate the three-phase currents of the motors [8].

Finally, integration with the distributed computing infrastructure is achieved through the 5GHz wireless interface (Wi-Fi 802.11ac). The Raspberry Pi encapsulates local telemetry under the DDS middleware, exposing ROS2 topics to the external processing device. This configuration allows for the decoupling of intensive computational load from the embedded hardware, delegating complex navigation algorithms to the remote node without compromising control loop latency [9].

3 Methodology

The architecture is based on ROS2 (Humble) and adopts a computational offloading strategy [5]. To justify this distributed approach over a monolithic one, the hardware constraints of low-cost embedded systems must be considered. While the local STM32 microcontroller is optimal for real-time motor control, it lacks the capacity to process high-frequency LiDAR point clouds or run a full ROS2 micro-agent. Conversely, running the complete navigation stack locally on the Raspberry Pi 4 is unfeasible due to its CPU and RAM limitations, which would introduce critical latency into the control loop. Therefore, the Raspberry Pi strictly acts as a high-bandwidth DDS gateway managing data acquisition and safety under *Best Effort* QoS policies [10]. Meanwhile, the external domain executes heavy navigation processing (*Nav2*, AMCL) [11] and global planning (A*). Dynamic obstacle avoidance is concurrently handled by the external *Nav2* local costmap via the relayed `/scan` topic.

For mission management, instead of fixed-radius tolerances that cause stops, we implement a virtual boundary criterion. Let P_{rob} be the robot's position, W_i the current objective, and v_{perp} the unit vector perpendicular to the desired trajectory segment $\overline{W_{i-1}W_i}$. We define the crossing condition $\mathcal{C}$ based on the 2D cross product:

$$\mathcal{C}(P) = (P_x - W_{ix}) \cdot v_{perp,y} - (P_y - W_{iy}) \cdot v_{perp,x} \quad (1)$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_24

The system automatically switches to the next objective W_{i+1} when the robot crosses the half-plane defined by the virtual boundary, that is, when the sign of its relative position coincides with that of the next objective:

$$\text{Switch\_if:} \quad \text{sgn}(\mathcal{C}(P_{rob})) = \text{sgn}(\mathcal{C}(W_{i+1})) \tag{2}$$

This formulation ensures that the robot geometrically validates the waypoint without needing to stop, guaranteeing fluid movement (Figure 2).

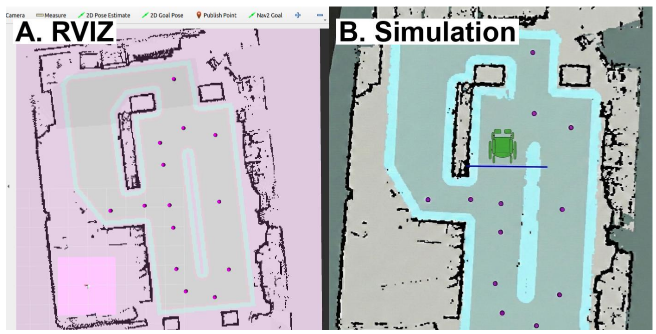

Fig. 2. Visual validation. (A) Rviz showing the waypoint sequence. (B) Simulation showing spatial coherence and costmaps.

4 Results

The experimental validation was carried out in a real industrial environment (Polígono del Montalvo, Salamanca, Spain), characterised by long concrete corridors and mixed obstacles (static and dynamic), exceeding the complexity of a controlled laboratory.

4.1 Experimental protocol and data acquisition

A test circuit was designed in the main corridor, recording telemetry via `ROS2 bag` in MCAP [1] format. Critical topics (`/odom`, `/scan`, `/tf`, `/plan`, and `/amcl_pose`) were stored for subsequent *offline* analysis. The protocol established three longitudinally aligned destinations to verify stability:

- **Destination A (Short):** 5 metres in a straight line.
- **Destination B (Medium):** 10 metres with an approach manoeuvre.
- **Destination C (Long):** 15 metres traversing the corridor to the end.

[1] MCAP is an open-source, serialization-agnostic container file format optimized for recording and playing back multimodal robotics data streams, such as high-frequency ROS2 topics.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_24

To guarantee consistency, the initial pose (x_0, y_0, θ_0) was standardised using marks on the floor. Thus, the final reported error aggregates both the deviation accumulated by navigation (odometry and AMCL) and the residual error of manual positioning.

4.2 Accuracy and efficiency analysis

20 sequential tests were performed, alternating destinations in a pseudo-random manner. In each trial, two fundamental metrics were recorded: total execution time (T_{exec}) and final positioning error (E_{pos}), measured as the Euclidean distance between the centre of the wheel axis and the coordinate of the physical target marked on the ground.

$$E_{pos} = \sqrt{(x - x_{obj})^2 + (y - y_{obj})^2} \quad (3)$$

Table 1 details the results obtained for each iteration. It is observed that the system successfully completed 100% of the proposed missions, maintaining notable consistency in arrival times.

Data analysis yields a global mean error of **7.98 cm**. This value validates the system's suitability for industrial and hospital environments, where the free passage width (doors and corridors) usually offers safety margins exceeding 20 cm. It is relevant to highlight the positive correlation between distance travelled and accumulated error: whilst in short paths (5m) the mean error sits around 4.3 cm, in long paths (15m) it rises to 11.8 cm, a behaviour consistent with the particle dispersion typical of the AMCL algorithm in the absence of distinctive geometric features during long corridor sections [12].

To correctly illustrate this navigation process, Figure 3 presents the performance analysis in the complex long-range scenario towards **Target C**. The costmap configuration defines two navigation corridors ("Upper Lane" and "Lower Lane") of approximately 2.4 metres in width, separated by a central virtual barrier simulating an operational restriction in the warehouse.

The robot initiates movement confined within the upper lane, maintaining a stable rectilinear trajectory ($y \approx 1.5m$) parallel to the shelving infrastructure. Upon surpassing the longitudinal extension of the virtual limit ($x > 11m$), the global planner traces a smooth transition curve to converge towards the target, situated on the symmetry axis of the warehouse ($y = 0$).

The comparison between global planning (grey dashed line) and real odometry (solid red line) demonstrates high tracking fidelity, even in the face of mechanical perturbations. As observed in the magnified box ("Initial turn detail"), the system effectively corrects the orientation deviation produced by the starting torque in the first 1.5 metres, converging rapidly towards the reference line. Following this stabilisation, the mean lateral error during the rectilinear phase remains below 5 cm, and in the curve phase, the robot adheres precisely to the smoothed profile. The dynamic robustness of the system, verified in this trial, extends to multiple test scenarios, the complete documentary record of which is available in [13].

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_24

Table 1. Experimental results of the 20 navigation tests. Averages are calculated independently for each target distance.

Trial (#)	Target	Distance (m)	Time (s)	Final Error (cm)
1	A	5.0	11.2	4.5
2	A	5.0	10.9	3.8
3	B	10.0	22.5	7.2
4	C	15.0	35.1	11.4
5	B	10.0	23.0	8.1
6	A	5.0	11.5	4.2
7	C	15.0	36.4	12.5
8	C	15.0	34.8	10.1
9	B	10.0	21.9	6.9
10	A	5.0	11.0	4.9
11	A	5.0	10.8	5.1
12	B	10.0	24.2	9.5
13	C	15.0	35.5	11.8
14	C	15.0	36.1	10.6
15	B	10.0	22.8	7.4
16	A	5.0	11.3	3.5
17	C	15.0	37.0	13.2
18	B	10.0	22.4	7.0
19	A	5.0	11.1	4.6
20	C	15.0	35.3	13.3
Average	**A**	**5.0**	**11.1**	**4.37**
Average	**B**	**10.0**	**22.8**	**7.68**
Average	**C**	**15.0**	**35.7**	**11.84**

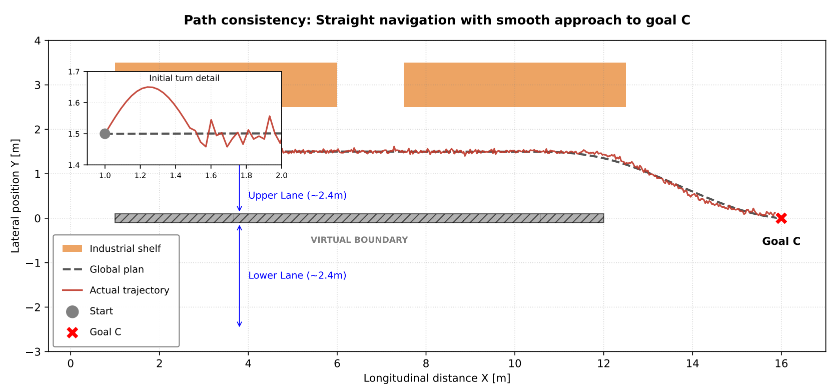

Fig. 3. Trajectory recorded during the approach manoeuvre to Target C. The graph illustrates navigation through the upper lane delimited by shelves (orange blocks) and the virtual wall (grey block).

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_24

5 Conclusions

This work has validated the design and implementation of an assistive robotic platform that manages to facilitate access to autonomous mobility through the use of low-cost hardware and open software architectures. The central contribution of the research lies in the empirical demonstration that computational decoupling in ROS2 allows for endowing resource-constrained devices with advanced navigation intelligence, such as a motorised wheelchair using reused *hoverboard* technology.

The experimental results obtained in the industrial environment confirm that the *Computation Offloading* strategy is technically viable and safe, despite outsourcing critical processes such as localisation (AMCL) and path planning (`Nav2`) to a remote node. This distributed architecture not only relieved the processing load of the local device but also guaranteed a network latency compatible with the vehicle's dynamics, even during complex long-range manoeuvres exceeding 15 metres in travel.

In terms of accuracy and safety, the trajectory analysis evidenced the efficacy of virtual boundaries to confine the robot's movement. The system rigorously respected the margins of the 2.4-metre navigation lanes defined in the costmap, achieving convergence towards the target with a final mean deviation of 7.98 cm. This level of accuracy validates the platform's suitability for operation in real infrastructures, such as hospital corridors or care homes, where reliability in route tracking is an indispensable requirement.

With a view to future research lines, the work will be oriented towards improving human-machine interaction and environment perception. It is planned to leverage the available computing surplus in the external node to integrate Large Language Models (LLMs), allowing the user to command the vehicle using natural voice instructions instead of graphical interfaces. Likewise, the incorporation of RGB-D sensors is foreseen to enrich the navigation map with three-dimensional information, mitigating the limitations of 2D LiDAR regarding suspended obstacles or glazed surfaces. Finally, the scalable nature of the DDS protocol lays the foundations for exploring multi-robot fleet coordination, managing the simultaneous traffic of multiple units in shared environments.

Acknowledgments. Sergio García González's work is subsidised by PREP2023-151701OB, funded by MCIN/AEI/10.13039/501100011033/FEDER and FSE+. This work was supported by the project Self-adaptive platform based on intelligent agents for the optimization and management of operational processes in logistic warehouses (PLAUTON), PID2023-151701OB-C21, funded by MCIN/AEI/10.13039/-501100011033/FEDER, EU.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_24

References

1. Eurostat: Ageing Europe — Looking at the lives of older people in the EU (2020 edition). Publications Office of the European Union, Luxembourg (2020). Available at: https://ec.europa.eu/eurostat/web/products-statistical-books/-/ks-02-20-655
2. Sahoo, S.K., Choudhury, B.B.: Autonomous navigation and obstacle avoidance in smart robotic wheelchairs. Journal of Decision Analytics and Intelligent Computing **4**(1), 47–66 (2024). https://doi.org/10.31181/jdaic10019022024s
3. Bonci, A., Gaudeni, F., Longhi, S., Monteriù, A., Prist, M.: Robot Operating System 2 (ROS2)-Based Frameworks for Increasing Robot Autonomy: A Survey. Applied Sciences **13**(23), 12796 (2023). https://doi.org/10.3390/app132312796
4. Macenski, S., Foote, T., Gerkey, B., Clay, C., Woodall, W.: Robot Operating System 2: Design, architecture, and uses in the wild. Science Robotics **7**(66), eabm6074 (2022). https://doi.org/10.1126/scirobotics.abm6074
5. Cacciabue, D., Marino, J., Aglieco, F., Levorato, M., Perroni, D., Risso, F.: Benchmarking Different Strategies for Offloading ROS2 Computation to the Edge. In: 2024 IEEE 10th International Conference on Network Softwarization (NetSoft), pp. 49–54. IEEE (2024). https://doi.org/10.1109/NetSoft60951.2024.10588914
6. Gallo, V., Shallari, I., Carratù, M., Laino, V., y Liguori, C. (2024). Design and Characterization of a Powered Wheelchair Autonomous Guidance System. *Sensors*, *24*(5), 1581. https://doi.org/10.3390/s24051581
7. Dias, N.F., Dyonisio, J.D.S., Quadros, L.F.M., et al.: SHARK: Stable Hoverboard-driven Autonomous Robot Kit. In: Proceedings of the V Brazilian Humanoid Robot Workshop (BRAHUR) and VI Brazilian Workshop on Service Robotics (BRASERO). Even3 (2024). https://doi.org/10.29327/v-brahur-vi-brasero.869310
8. Song, B., Tang, S., Li, Y.: A new path planning strategy integrating improved ACO and DWA algorithms for mobile robots in dynamic environments. Mathematical Biosciences and Engineering **21**(2), 2189–2211 (2024). https://doi.org/10.3934/mbe.2024096
9. Maruyama, Y., Kato, S., Azumi, T.: Exploring the performance of ROS2. In: Proceedings of the 13th International Conference on Embedded Software (EMSOFT '16), Article 5. ACM, New York (2016). https://doi.org/10.1145/2968478.2968502
10. Gutiérrez, C.S.V., San Juan, L.U., Ugarte, I.Z., Vilches, V.M.: Towards a distributed and real-time framework for robots: Evaluation of ROS 2.0 communications for real-time robotic applications. arXiv preprint arXiv:1809.02595 (2018). Available at: http://arxiv.org/abs/1809.02595
11. Macenski, S., Martín, F., White, R., Ginés Clavero, J.: The Marathon 2: A Navigation System. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2718–2725. IEEE (2020). https://doi.org/10.1109/IROS45743.2020.9341207
12. He, S., Song, T., Wang, P., Ding, C., Wu, X.: An Enhanced Adaptive Monte Carlo Localization for Service Robots in Dynamic and Featureless Environments. Journal of Intelligent & Robotic Systems **108**, 5 (2023). https://doi.org/10.1007/s10846-023-01858-7
13. García González, S., et al.: Low-Cost Autonomous Navigation: Experimental Tests and Demonstration. YouTube Video (2024). Available at: https://www.youtube.com/watch?v=-GdxQIHB9m4

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_24