electronics

MDPI

Article

Autonomous Medical Robot Trajectory Planning with Local Planner Time Elastic Band Algorithm

Arjon Turnip [1,*], Muhamad Arsyad Faridhan [1], Bambang Mukti Wibawa [1] and Nursanti Anggriani [2]

[1] Department of Electrical Engineering, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Kabupaten Sumedang 45363, Indonesia; muhamad20059@mail.unpad.ac.id (M.A.F.); b.mukti.wibawa@unpad.ac.id (B.M.W.)
[2] Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Kabupaten Sumedang 45363, Indonesia; nursanti.anggriani@unpad.ac.id
* Correspondence: turnip@unpad.ac.id

Abstract: Robots have made significant contributions across various industries due to their efficiency and effectiveness. However, indoor navigation remains challenging due to complex environments and sensor signal interference. Changes in indoor conditions and the limited range of GPS signals necessitate the development of an accurate and efficient indoor robot navigation system. This study aims to create an autonomous indoor navigation system for medical robots using sensors such as Marvelmind, LiDAR, IMU, and an odometer, along with the Time Elastic Band (TEB) local planning algorithm to detect dynamic obstacles. The algorithm's performance is evaluated using metrics like path length, duration, speed smoothness, path smoothness, Mean Squared Error (MSE), and positional error. In the test arena, TEB demonstrated superior efficiency with a path length of 155.55 m, 9.83 m shorter than the Dynamic Window Approach (DWA), which covered 165.38 m, and had a lower yaw error of 0.012 radians. TEB outperformed DWA in terms of speed smoothness, path smoothness, and MSE. In the Sterile Room Arena, TEB had an average path length of 14.84 m, slightly longer than DWA's 14.32 m, but TEB navigated 2.82 s faster. Additionally, TEB showed better speed and path smoothness. In the Obstacle Room Arena, TEB recorded an average path length of 21.96 m in 57.3 s, outperforming DWA, which covered 23.44 m in 61 s, with better results in MSE, speed smoothness, and path smoothness, highlighting superior path consistency. These findings indicate that the TEB algorithm is an effective choice as a local planner in dynamic hospital environments.

Keywords: autonomous navigation; ROS; global planer; time elastic bane

Academic Editor: Andrea Bonci

Received: 15 November 2024
Revised: 19 December 2024
Accepted: 26 December 2024
Published: 4 January 2025

Citation: Turnip, A.; Faridhan, M.A.; Wibawa, B.M.; Anggriani, N. Autonomous Medical Robot Trajectory Planning with Local Planner Time Elastic Band Algorithm. *Electronics* **2025**, *14*, 183. https://doi.org/10.3390/electronics14010183

1. Introduction

The development of autonomous robot technology has become a key focus in industry and research, particularly in the context of addressing complex, monotonous, repetitive, and physically demanding tasks. This technology offers effective solutions for handling work that requires consistency, precision, and long-term durability without physical or mental fatigue. Several important reasons for the development of this technology include efficiency, safety, and increased productivity, which directly contribute to the quality of output across various sectors, including manufacturing, agriculture, logistics, and healthcare. Repetitive and monotonous tasks, such as assembling goods on production lines, monitoring crops in agriculture, or transporting items in warehouses, require consistency that is difficult for human workers to maintain over extended periods. Research has shown that autonomous robots can perform these tasks faster and more consistently, significantly improving operational efficiency [1]. Additionally, automation can reduce human error, which often occurs in repetitive tasks due to fatigue or boredom. Autonomous robots have the potential to boost productivity by working non-stop for 24 h, a feat that is impossible for human workers. This enables industries to increase production volumes without significantly increasing the workforce. A study by Jones et al. (2021) found that autonomous robots in production lines could reduce operational costs by up to 30% due to increased productivity and reduced downtime [2].

In agriculture, the use of autonomous drones for crop monitoring and pesticide spraying has led to a 20% increase in crop yields due to optimized water and pesticide use [3]. Autonomous robots also help reduce the risk of workplace accidents, particularly in hazardous environments such as mining, construction, or areas at risk of disease transmission [4]. Robots designed for specific tasks can be equipped with sensor systems to avoid accidents and detect potential hazards, providing additional protection for people in their vicinity. As the elderly population grows in many developed countries, labor shortages have become a serious issue, especially in sectors dependent on physical labor. Autonomous robots can serve as a solution to this labor shortage by replacing or assisting human workers in tasks that require significant physical effort and long durations [5]. Autonomous robots also support sustainability initiatives, particularly in resource management. For instance, plant watering robots in agriculture equipped with moisture sensors can ensure that water is only supplied to areas that need it, thereby reducing water consumption and optimizing irrigation. A literature review by Zhang and Wang (2022) showed that sensor- and AI-based automation can reduce resource usage by up to 40%, directly contributing to environmental sustainability [6].

Robots have made significant contributions to humans in various fields, such as warehouse delivery, floor cleaning, and as household assistants. Indoor navigation technology is becoming increasingly important due to its widespread use in various applications. However, indoor navigation remains a challenge due to the complex indoor environments and signal interference from sensors [7]. Indoor robot navigation requires the development of more accurate and effective technologies because of frequent environmental changes and the limited range of GPS signals indoors [8]. The development of indoor navigation technology for robots is advancing to improve accuracy. Challenges in developing these navigation systems include measurement accuracy, data processing with various filtering techniques, the robot's limited range of movement, and many other factors. To address the challenges of autonomous indoor navigation, artificial intelligence can be used to develop algorithms and systems capable of tasks such as decision-making, pattern recognition, and machine learning [9,10]. To tackle these issues, previous research has presented various approaches to indoor robot navigation. Zulfan J. and Rafly developed an autonomous robot using Marvelmind sensors, LiDAR, IMU, and rotary encoders with the Adaptive Monte Carlo Localization method and Extended Kalman Filter [11,12]. The results showed that the robot could avoid obstacles with low error rates. LiDAR was used for real-time environmental mapping but had limitations in measuring large distances and interference from surrounding objects.

Currently, much research is focused on the development of autonomous navigation robots. Arjon Turnip et al. [13] developed a medical assistant robot to replace nurses in patient isolation at hospitals, creating a robot that moves autonomously, interacts, and detects faces. Research by Rivai et al. [14] introduced a 2D mapping technique using an omnidirectional moving robot equipped with LiDAR, while a study by Hai Wang et al. [15] proposed a real-time vehicle detection algorithm based on the combination of vision and LiDAR point cloud. The implementation of interactive medical robots was explored by Turnip et al. [16], who examined the use of deep learning-based voice chatbots. This research highlighted the potential integration of chatbot technology to enhance interaction and functionality in medical robots. Furthermore, Suryawan et al. [17] demonstrated significant progress in this field. However, the research by Suryawan et al. used teleoperated manual control, which is considered ineffective due to the need for an operator and vulnerability to environmental disturbances in hospital settings. Alternatively, the medical assistant robot by Nurhayati [18–20] uses a line-following system, though it is difficult to implement in hospitals due to the need for obstacle-free paths. Zhao et al. [20] utilized the Robot Operating System (ROS) with the Karto SLAM algorithm for mapping, A* for global planning, and DWA for local planning. LiDAR, IMU, and odometers were used for accurate navigation, but the system was slow in building maps and tracking the robot's position.

Recent studies on mobile robot navigation emphasize advanced technologies and applications, including the use of LIDAR, indoor GPS, emerging trends, and developments in autonomous navigation systems [21–23]. Path planning is essential for enabling autonomous navigation and optimizing efficiency in robotic systems. Various approaches have been developed to address challenges such as navigating cluttered environments, achieving energy efficiency, and ensuring precision in diverse terrains. Advanced techniques, including reinforcement learning, hybrid algorithms, and motion capture technologies, demonstrate significant progress in both theoretical and practical applications. These advancements highlight the growing potential for enhancing navigation capabilities in robotics for both ground and aerial applications [24–29].

This study on an autonomous indoor navigation system for medical robots presents several novel contributions, particularly its focus on dynamic hospital environments. By integrating sensors such as Marvelmind, LiDAR, IMU, and odometers with the Timed Elastic Band (TEB) algorithm, this study addresses challenges like signal interference and environmental changes. Unlike many previous studies, this research evaluates TEB in real-world medical scenarios, demonstrating superior performance over the Dynamic Window Approach (DWA) in metrics such as path smoothness, speed smoothness, Mean Squared Error (MSE), and positional error. When compared to related work, this study stands out for its specialized application. For instance, "Comparison and Improvement of Local Planners on ROS for Narrow Passages" focuses on optimizing DWA and TEB for narrow environments but lacks the focus on dynamic human-centric conditions found in hospitals [30]. Similarly, while "Comparison of ROS Local Planners with Differential Drive Heavy Robotic System" analyzes planners in heavy robotics, it doesn't address the intricacies of hospital environments with frequent human interaction [31]. Papers such as "Enhancing Social Robot Navigation with Integrated Motion Prediction" explore social navigation, but the current study emphasizes hospital-specific metrics and real-time adaptability [32].

This research also differs from works like "Resilient Timed Elastic Band Planner for Collision-Free Navigation", which enhances TEB for high-obstacle-density environments using Voronoi maps, and "Improved Timed Elastic Band Algorithm for AGVs", which optimizes energy efficiency during turns [33,34]. In contrast, this medical robot study prioritizes navigation efficiency and obstacle avoidance in dynamic settings. Studies such as "A Study on the Effect of Parameters for ROS Motion Planners" and "Navigation for Mobile Robots to Inspect Aircraft" are broader in scope, focusing on parameter effects and aircraft inspection scenarios, respectively, rather than medical environments [35]. Finally, while "A Comparison Study Between Traditional and DRL Algorithms" highlights reinforcement learning, this medical robot study confines itself to traditional algorithms, with an emphasis on practical, dynamic, and human-interactive scenarios [36].

This study significantly advances medical robotics by addressing key challenges in indoor navigation and presenting an efficient, validated solution for real-world healthcare settings. It introduces a novel framework integrating multi-sensor fusion—Marvelmind, LiDAR, IMU, and odometer sensors—with the Time Elastic Band (TEB) algorithm, tailored for medical robots to overcome GPS limitations and adapt to dynamic indoor environments. By systematically evaluating TEB's performance across metrics such as path length, duration, speed smoothness, path smoothness, Mean Squared Error (MSE), and positional error, this study demonstrates its superiority over the Dynamic Window Approach (DWA) in navigation efficiency, accuracy, and reliability. The research further proposes a hybrid navigation strategy, leveraging TEB for narrow spaces and DWA for open areas, optimizing adaptability to varying hospital conditions. Validated across multiple controlled environments, such as the Test Arena, Sterile Class Arena, and Obstacle Class Arena, the system proves its practical applicability. By effectively combining sensor data and showcasing robust performance in dynamic scenarios, this work provides a practical and efficient solution for advancing autonomous medical robotics, setting a benchmark for local navigation systems in healthcare environments.

The remainder of this paper is organized as follows: Section 2 presents the proposed methodology and the experimental setup, detailing the integration of multi-sensor fusion and the TEB algorithm including the controlled environments used for validation. Section 3 discusses the results, benchmarking the TEB algorithm against DWA across various performance metrics. Finally, Section 4 concludes the paper by summarizing the findings, emphasizing the practical implications for medical robotics, and suggesting potential directions for future research.

2. Methods

This research was conducted at PPBS Laboratory C, 3rd Floor, Padjadjaran University, West Java, Indonesia. The testing arena was designed to represent a nurse's and a patient's room, as shown in Figure 1. The robot trajectory path is marked with seven waypoints, namely waypoints 1 to 7, where waypoints 1–6 are patient rooms. In the test scenario, the robot is designed to be able to deliver medicine or food to three patient rooms in sequence as well as take biosignal measurements, such as blood pressure, body temperature, and oxygen saturation, in a single trip. The robot's ability to perform these various tasks simultaneously aims to mimic the daily activities of a nurse in a hospital in Indonesia.

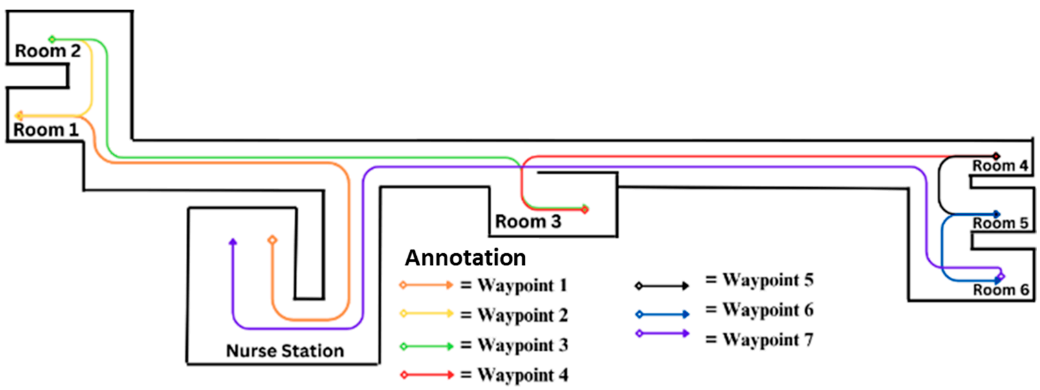

Figure 1. Waypoint for Arena PPBS C.

The testing process was conducted in 5 scenarios, with a total of 7 waypoints to each patient room. The robot moved sequentially from room to room, navigating automatically using the Trajectory Elastic Band (TEB) algorithm. An additional class-shaped test arena was used to test the effectiveness of the TEB algorithm in handling dynamic environments.

This arena includes conditions that represent environmental changes and moving obstacles, as shown in Figure 2. Figure 2a shows a narrow room condition, while Figure 2b depicts a narrow room with additional dynamic obstacles. This test aims to evaluate the ability of the TEB algorithm to plan safe and efficient paths under various scenarios with different navigation challenges. With the addition of the testing room, the robot's local planner system can be tested in more depth to assess its effectiveness in planning a path through narrow passageways, selecting the most efficient path to navigate, and observing the robot's movement characteristics in a confined area. In the initial stage, tests were conducted in an arena without obstacles (sterile conditions) using 8 setpoints to observe the basic characteristics of robot movement. Next, tests were conducted in a more complex arena with additional obstacles (Figure 2b) using 6 setpoints to evaluate the robot's ability to avoid obstacles and make navigation adjustments according to environmental conditions.

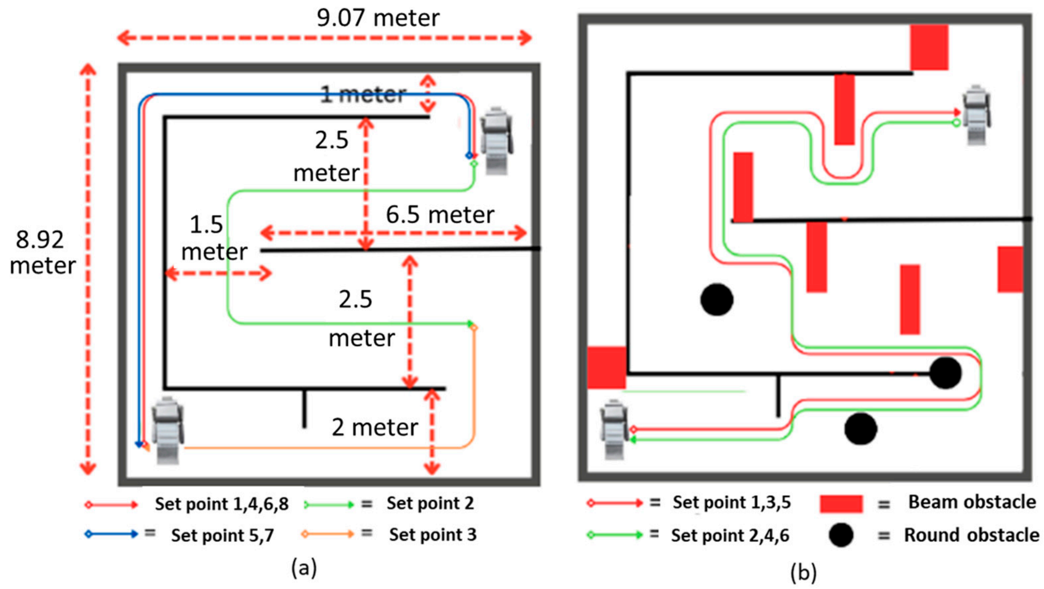

Figure 2. Waypoint for dynamic narrow room simulation: (**a**) with, (**b**) without Obstacle.

Figure 3 illustrates the circuit schematic of the medical robot, highlighting the integrated relationship between its components. The robot is powered by a 24-volt LiFePO4 battery, which is recharged using a battery charger and protected by a fuse to prevent power surges. The battery's power is distributed through step-down converters: one regulates the voltage to 19 Volts to power the Intel NUC mini PC, while another converts it to 5 Volts for the control PCB. A terminal block ensures efficient power distribution across the subsystems.

Control and processing are managed by the Intel NUC, a compact computer responsible for high-level processing and decision-making, and the ESP32 microcontroller, which handles communication and low-level control. A custom PCB control board centralizes connections and manages components such as motor drivers and sensors. The robot's sensor system consists of LiDAR, Marvelmind, IMU, and odometer sensors, working together to provide accurate navigation, orientation, and position data. The LiDAR sensor connects directly to the Intel NUC and integrates with the Robot Operating System (ROS). It uses a laser to measure object distances through light reflection with high accuracy, enabling obstacle detection and avoidance. The Marvelmind system connects to the mini PC via an HW receiver modem, receiving ultrasonic wave data from super beacons. This system calculates distances and delivers highly accurate x and y coordinate data, ensuring the robot stays on its intended path.

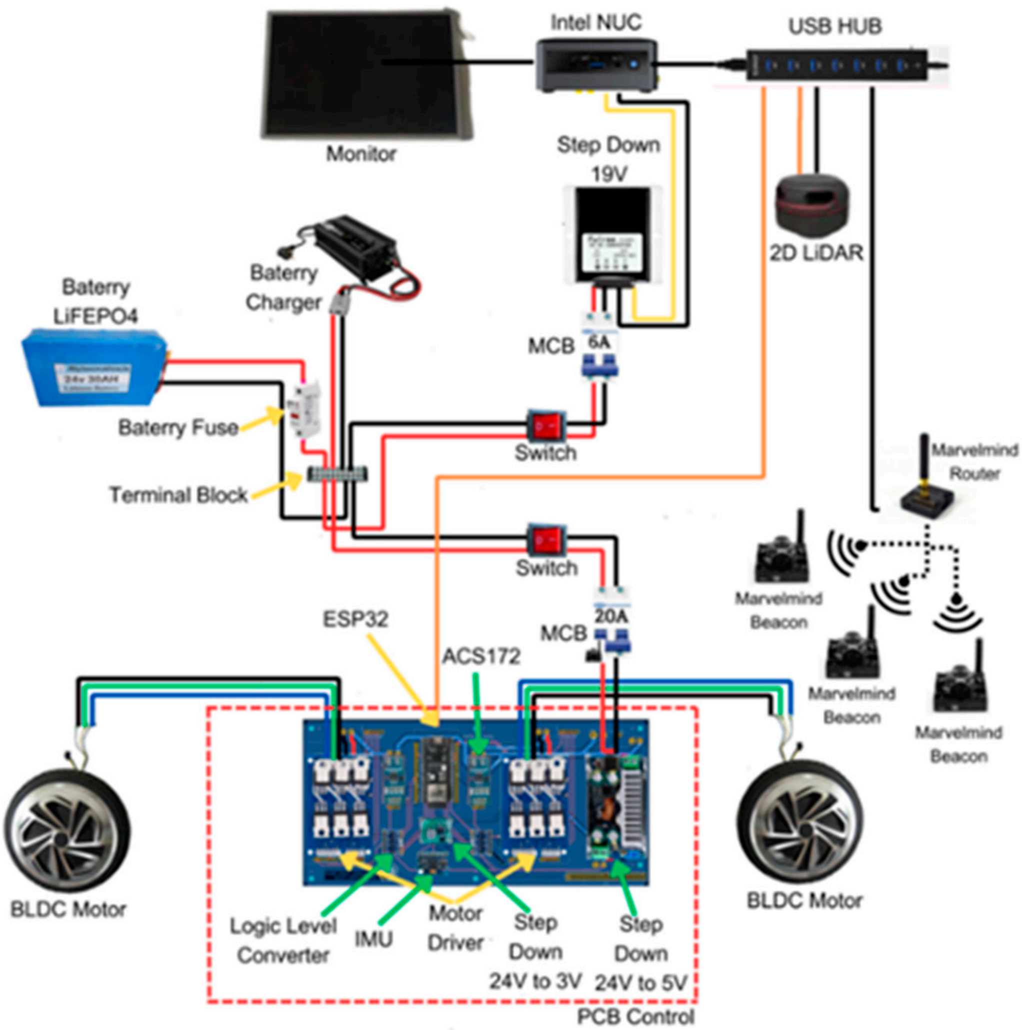

Figure 3. Robot Schematics.

Actuation is powered by two BLDC motors controlled by a motor driver that regulates their speed and direction. The system incorporates switches and miniature circuit breakers (MCBs) to ensure safe operation by protecting circuits from overload. Additionally, a USB hub connects peripherals to the Intel NUC, and a monitor displays system information for user monitoring. This integrated design supports the efficient, reliable, and safe operation of the medical robot in dynamic indoor environments.

The Inertial Measurement Unit (IMU) is connected to the ESP32 microcontroller to read the linear and angular acceleration values, which allows measuring and tracking the robot's orientation in three-dimensional space. The IMU sensors consist of accelerometers, gyroscopes, and magnetometers that provide motion and orientation data, used to correct and stabilize the robot's movements for greater precision. A rotary encoder is mounted on the BLDC motor to measure and calculate the rotation of the wheel or motor shaft. This magnetic incremental encoder produces output pulses that provide information about the wheel rotation and direction of rotation. This data, generated in real-time, supports more accurate control of the robot's position and speed, improving efficiency and precision in navigation and movement control. This interconnected set of components gives the robot the ability to perform better navigation and orientation, supporting its application in medical tasks that require high precision in movement and positioning in operational environments.

In Figure 4, the autonomous navigation system is composed of several integrated stages designed to determine the robot's position, create a map, and plan its route effectively.

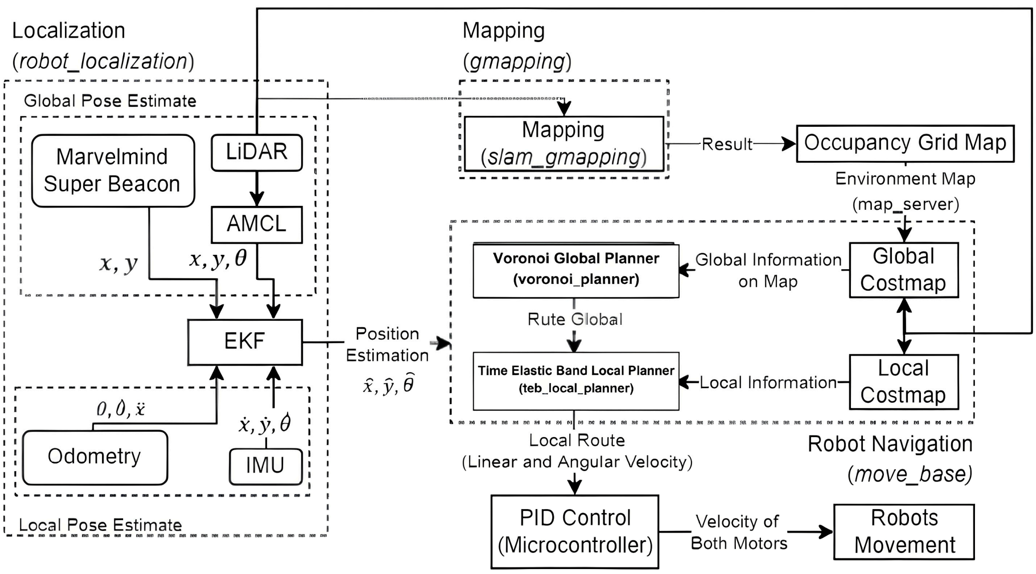

Figure 4. Autonomous Navigation System.

1. Localization Stage: This stage leverages the robot_localization package and various sensors to estimate the robot's position accurately. The Marvelmind Super Beacon provides global position estimates in x and y coordinates, while the LiDAR sensor generates distance data. This distance data is combined with the Adaptive Monte Carlo Localization (AMCL) method to estimate the robot's position in three dimensions (x, y, θ). Additionally, odometry tracks local position changes (Δx, Δy, Δθ), and the IMU supplies acceleration and rotation data. These data inputs are fused using an Extended Kalman Filter (EKF), which minimizes errors arising from the limitations of individual sensors, resulting in a more precise position estimate.
2. Mapping Stage: Data from the LiDAR sensor and the position estimates generated during localization are processed by the SLAM_GMapping algorithm to create an Occupancy Grid Map. This map identifies obstructed and navigable areas, serving as the foundation for autonomous navigation and obstacle avoidance.
3. Path Planning Stage: With the Occupancy Grid Map in place, the Voronoi Global Planner calculates a global route from the robot's starting point to its target destination. This global path is further refined by the TEB Local Planner, which incorporates dynamic factors such as the robot's speed, direction changes, and shifting environmental conditions, to create an optimal local route.
4. Speed Control: In the final stage, the PID Controller embedded in the robot's microcontroller uses the data from the local route to set the linear and angular velocities of the robot's motors. These velocity instructions are sent in the form of cmd_vel or command velocity commands, which enable the robot to move responsively along the planned route.

Overall, this autonomous navigation system seamlessly integrates advanced technologies across localization, mapping, path planning, and control. This ensures precise, efficient navigation while adapting to dynamic environmental changes, making the system highly effective in complex indoor scenarios.

2.1. Local Planner Time Elastic Band

Time-Elastic Band Algorithm is a path planning algorithm used to control the motion of autonomous robots with differential drive kinematics models. This algorithm optimizes the path based on several objectives, such as speed, safety, and energy efficiency, and is suitable for use in complex, dynamic environments and spaces with limited space. In robots with a -*-differential drive kinematics model, the speed of the left wheel and right wheel are used to determine the linear velocity (V) and angular velocity (ω) of the robot. The equation of velocity and acceleration is shown in Equation 1.

$$u(t) = \begin{bmatrix} V\ (t) \\ \omega(\mathrm{t}) \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{D} & \frac{1}{D} \end{bmatrix} \begin{bmatrix} V_L \\ V_R \end{bmatrix} \tag{1}$$

where V_L dan V_R are the left and right wheel speeds of the differential drive robot, and D is the distance between the two wheels. TEB increases the time interval between robot movements from one pose to the next based on the Elastic Band algorithm. These poses and time intervals are optimized using the G2o package Equations (2) and (3), resulting in smoother and more efficient trajectories.

$$\begin{cases} Q = \{S_i\}, i = 1, 2, \ldots,\ nn\ \in\ \mathbb{N} \\ \tau = \{\Delta T_i\}\ i = 1, 2, \ldots,\ nn - 1 \end{cases} \tag{2}$$

$$B\ := (Q, \tau) = [S_1, \Delta T_i, S_2, \Delta T_2, \ldots, \Delta T_{n-1,} S_n] \tag{3}$$

The objective constraint function in the TEB algorithm focuses on criteria such as speed and acceleration limits, obstacle avoidance, and time efficiency. These criteria ensure that the planned path is not only optimal but also safe and efficient under various operational conditions, both for static and dynamic obstacles. The penalty function in Equation (4) in the TEB algorithm is used to penalize violations of the constraints that robot path planning must adhere to. This function measures the extent to which a value violates certain constraints.

$$e_\tau(x, x_r, \varepsilon, S, n) \approx \begin{cases} \left(\frac{x-(x_r-\varepsilon)}{S}\right)^n & , x > x_r - \varepsilon \\ 0 & , x \leq x_r - \varepsilon \end{cases} \tag{4}$$

Punishment function e, plays a role in measuring the degree of violation of a value x against predetermined limits. Thus, this function provides a penalty value that is proportional to the amount of violation that occurs, so that it can direct the optimization process so that the resulting solution meets the existing constraints. In the Time- Elastic Band (TEB) algorithm, the multi-objective optimization function in Equation (5) is used to combine several criteria that need to be met in robot path planning. These criteria include speed, acceleration, obstacle avoidance, and travel time constraints. By considering all these criteria, the TEB algorithm aims to produce a path that is optimal and safe, while being efficient in achieving the desired goal.

$$f(B) = \sum_k \gamma_k f_k(B) \tag{5}$$

Optimization in the Time-Elastic Band algorithm aims to achieve a balance between various desired objectives or constraints. Each criterion $f_k(B)$ in path planning is given a weight of γ_k. The magnitude of the value γ_k determines how much influence the criteria have on the optimization results. In determining the value of γ_k, and the handling of each $f_k(B)$ system ensures that the final result not only meets the mathematical requirements but also represents the desired parameters of each weight with the aim that the robot navigation system runs as safely and efficiently as possible.

2.2. Local Planner Assessment Parameters

To test the performance of the TEB algorithm, it is necessary to process data from the ROS topic so that it can be analyzed regarding the characteristics of robot movement in implementing the algorithm used. Some of the performance assessments of the TEB and DWA local planners that will be tested include: (i) Mean Square Error (MSE) value to get a quantitative measure of how accurate the robot is in achieving position and orientation on the optimal path. The MSE value is calculated based on the error of x, y position, and yaw orientation on the robot against the optimal path. The calculation of the error value is shown in Equation (6).

$$\begin{aligned} \text{Error}_x &= \text{X}_{goal} - \text{X}_{odom}\ ; \\ \text{Error}_y &= \text{Y}_{goal} - \text{Y}_{odom}\ ; \\ \text{Error}_{yaw} &= \theta_{goal} - \theta_{odom} \end{aligned} \tag{6}$$

The MSE value is calculated as the mean square of the error through Equation (7).

$$\begin{aligned} \text{MSE}_x &= \tfrac{1}{n}\textstyle\sum_{i=1}^{n}\left(\text{Error}_{xi}\right)^2 \\ \text{MSE}_y &= \tfrac{1}{n}\textstyle\sum_{i=1}^{n}\left(\text{Error}_{yi}\right)^2 \\ \text{MSE}_{yaw} &= \tfrac{1}{n}\textstyle\sum_{i=1}^{n}\left(\text{Error}_{yawi}\right)^2 \end{aligned} \tag{7}$$

By calculating the MSE value for each of the x, y, and yaw variables the robot's performance in achieving the optimal position and orientation can be analyzed and compared; (ii) Navigation Duration is used to evaluate the robot's navigation efficiency in achieving the predefined target. This duration shows how fast the robot can reach the destination from the initial position after receiving the command as shown in Equation (8).

$$T = t_N - t_1 \tag{8}$$

where T is the navigation duration, t_N, is the time when the robot reaches the goal, and t_1 is the time when the robot receives the command to move; (iii) Path Length is needed to evaluate the efficiency of the path taken by the robot in achieving the predetermined goal. The path length shows the total distance traveled by the robot from the starting position until it reaches the goal.

$$Path\ Length = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\left(x_{i+1} - x_i\right)^2 + \left(y_{i+1} - y_i\right)^2} \tag{9}$$

where x_i and y_i are the coordinates of the robot's position at the i-th point, and n is the total number of coordinate points recorded from the start to the destination. By calculating the path length, it can be evaluated whether the robot takes the optimal path or there is a significant deviation from the shortest path it should take; (iv) Path Smoothness is used to assess the quality of robot movement. The local planner algorithm is essential in navigating the robot from one point to another considering the surrounding conditions. Path smoothness calculates the vector displacement difference between two consecutive points in two dimensions (x and y) as shown in Equation 10.

$$f_{ps} = \sum_{i=2}^{N-1}\left\|\Delta x_{i+1} - \Delta x_i\right\|^2 \tag{10}$$

Path smoothness ensures the robot moves in a controlled and predictable manner, reducing sharp tums and sudden movements, maintaining stability, reducing component wear, and improving navigation safety and efficiency, especially in environments with obstacles or in tasks that require precision; (v) Velocity Smoothness to measure the smooth and controlled acceleration and deceleration of the robot over time. Smoother acceleration means that the robot moves with more stable changes in speed, without sudden acceleration or deceleration. The calculation of the velocity smoothness parameter is shown in Equation 11.

$$f_{vs} = \frac{1}{N-1}\sum_{i=1}^{N-1}\left|\frac{v_{i+1}-v_i}{t_{i+1}-t_i}\right| \quad (11)$$

where v is the robot speed at the i-th time, t is the time at the i-th point, and N is the total number of data points. These metrics assess the change in velocity between points, helping to ensure stable and controlled movement of the robot.

3. Results and Discussion

In autonomous robot navigation, the system generates a map from the map topic, which is then used to create both local and global costmaps. The mapping is achieved using the SLAM-Gmapping algorithm, which processes lidar data from the /scan topic and hall sensor data from the /odom topic. The resulting maps are saved in Portable Gray Map (PGM) format within the /map_server topic and are further processed by the Voronoi Planner algorithm to generate safe and efficient global paths. During mapping, the robot is manually controlled through the teleop_twist_keyboard ROS package, which sends motion commands to the /cmd_vel topic based on keyboard input. Using lidar sensors and encoders, the robot maps the PPBS Building area and each room, capturing detailed environmental data. The mapping results for the PPBS rooms are shown in Figure 5a, while those for a smaller room representing limited space with dynamic obstacles are displayed in Figure 5b. This study compares two local planner algorithms: the Dynamic Window Approach (DWA) from the move_base package and the Time Elastic Band (TEB), known for its time efficiency and path-planning capability. Testing was conducted across three scenarios. Scenario 1 involves navigating the environment in Figure 5a along the path in Figure 1. Scenarios 2 and 3 involve navigating the space shown in Figure 5b, with paths depicted in Figure 2a (without obstacles) and Figure 2b (with obstacles), respectively. In Scenario 1, the robot traverses seven set points, each corresponding to those listed in Figure 1. The algorithm performance is assessed by comparing DWA and TEB to identify the more effective algorithm for navigating the PPBS rooms.

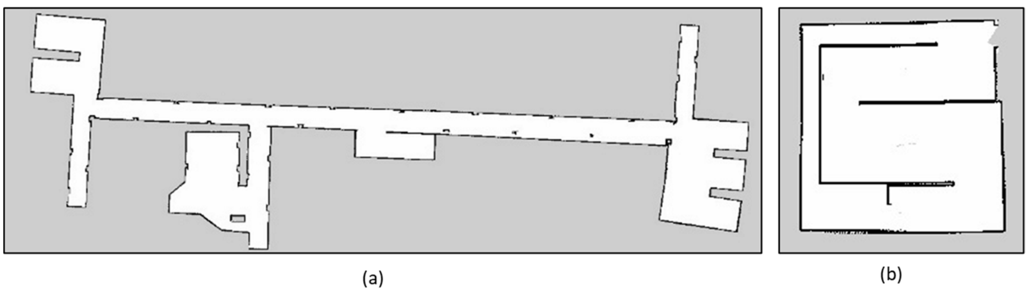

Figure 5. Map of Testing room: (**a**) Nurse's and a patient's room area (PPBS rooms), (**b**) Dynamic narrow room.

Robot navigation plots across five test scenarios as shown in Figure 6 compare the trajectories generated by the DWA and the TEB algorithms for autonomous navigation. Both graphs illustrate the robot's path along the X-Y plane in five different navigation scenarios. Figure 6a shows the trajectories produced by the DWA algorithm. In Section A, which highlights the robot's navigation in a complex, narrow region, DWA demonstrates the difficulty in maintaining consistent and smooth trajectories, exhibiting sharp turns and minor deviations. Section B depicts the robot's navigation of an initial curved segment.

While DWA successfully follows the route, there are observable variations in path accuracy and smoothness. Section C corresponds to a long, relatively straight path where the DWA trajectories exhibit oscillations, indicating inefficiencies in maintaining a steady course.

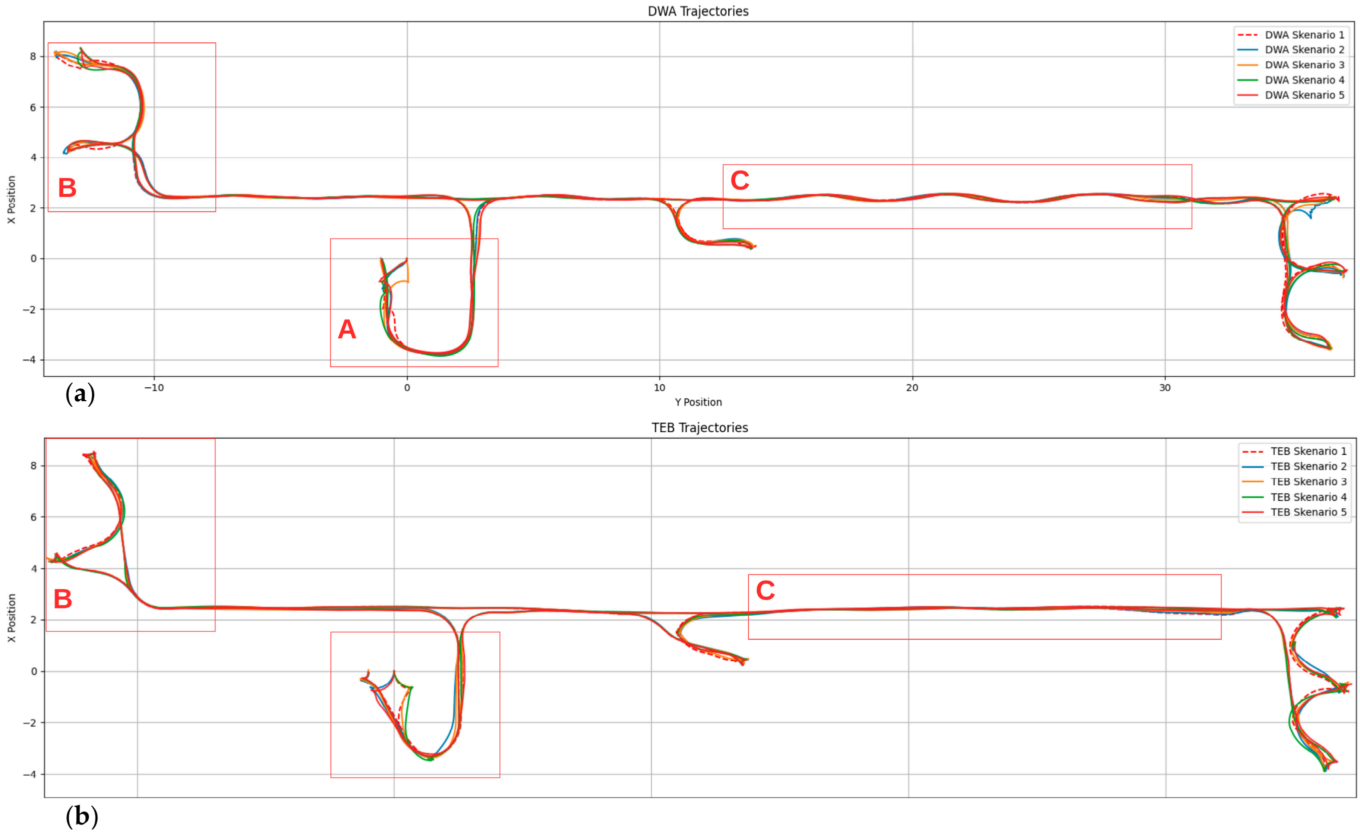

Figure 6. Plot Trajectory on Scenario 1.

Figure 6b presents the trajectories generated by the TEB algorithm. In Section A, TEB performs more consistently in the narrow region, with smoother and more uniform paths across scenarios compared to DWA. Section B shows the robot's effective handling of the initial curved segment, with improved path consistency and reduced variation. Section C highlights TEB's significant advantage in maintaining a steady trajectory along the straight path, with minimal oscillations or deviations. The comparison reveals that TEB consistently outperforms DWA in terms of path smoothness, accuracy, and uniformity across all scenarios. The smoother paths generated by TEB indicate better handling of dynamic obstacles and environmental complexities, making it a more reliable algorithm for navigation in dynamic environments. The results for the five PPBS scenarios are provided in Table 1.

Table 1 presents a comparative analysis of two navigation algorithms DWA and TEB across multiple path-planning scenarios. The metrics measured include path length, duration, velocity smoothness, path smoothness, and Mean Squared Error (MSE), each averaged across scenarios for both algorithms. Based on Path Length category, TEB consistently produces shorter paths (mean of 155.5 m) compared to DWA (mean of 165.38 m), suggesting that TEB can find more efficient paths for navigation. TEB takes slightly longer in three out of five scenarios and, on average, has a similar duration to DWA (429 s for TEB vs. 428.2 s for DWA). The velocity smoothness values highlight a notable advantage for TEB, with a mean of 0.0599 m/s^2 compared to 0.0871 m/s^2 for DWA. Lower values indicate smoother accelerations and decelerations along the path, which is beneficial in applications requiring steady movement, such as autonomous vehicles carrying sensitive cargo. This smoother velocity profile could improve passenger comfort or load stability in practical scenarios. Both algorithms exhibit similar path smoothness (0.0017 m^2 for TEB vs. 0.00183 m^2 for DWA on average). This metric suggests that the paths generated by both algorithms avoid abrupt turns, although TEB has a slight advantage. Higher path smoothness is often desired to reduce the wear on mechanical components by preventing sharp direction changes. The MSE values are comparable between the algorithms, with TEB at 0.60 and DWA at 0.613 on average. The MSE provides insight into the accuracy and reliability of the path in adhering to the desired trajectory. Here, TEB's slight improvement suggests better adherence to the planned path, which may result in a more predictable route.

Table 1. Performance Evaluation of Local Planners in Scenario 1.

Scenario/Rooms		Path Length (m)		Duration (s)		Velocity Smoothness (m/s^2)		Path Smoothness (m^2)		MSE (m^2)	
		DWA	**TEB**	**DWA**	**TEB**	**DWA**	**TEB**	**DWA**	**TEB**	**DWA**	**TEB**
1	1	166.07	156.5	423	446	0.0803	0.0740	0.00179	0.0018	0.620	0.61
	2	166.78	154.7	441	413	0.0976	0.0620	0.00196	0.0017	0.623	0.59
	3	165.41	154.4	415	421	0.0784	0.0299	0.00180	0.0016	0.636	0.60
	4	164.89	156.4	439	443	0.0886	0.0676	0.00184	0.0017	0.606	0.61
	5	163.75	155.5	423	426	0.0909	0.0662	0.00176	0.0017	0.578	0.58
Means		165.38	155.5	428.2	429.8	0.0871	0.0599	0.00183	0.0017	0.613	0.60

The comparison indicates that TEB generally performs better in terms of path length, velocity smoothness, and slightly improved path smoothness and MSE. These metrics suggest that TEB may be a preferable choice in applications where efficient path planning and smoother velocity profiles are critical. The trade-off, however, is a marginally increased travel time in some scenarios, which may or may not be significant depending on the use case. Overall, TEB appears to offer superior optimization in terms of navigation efficiency and smoothness, making it potentially more suitable for tasks requiring high precision and smooth navigation. DWA, while slightly less optimized in these areas, may still be preferable in situations where marginally faster response times are required and path efficiency is less critical.

In Scenario 2, it is made in the Room Arena which contains narrow and wide aisles according to the results of SLAM-Gmapping as many as eight set points according to Figure 2. Table 2 shows the performance assessment of the TEB and DWA algorithms for each goal as many as eight goals with six goals passing through a narrow hallway and two goals passing through a wide hallway.

In comparing the DWA and TEB methods across different metrics for a series of path goals in Scenario 2, several key insights emerge: The DWA method generally resulted in shorter path lengths, with an average of 14.32 m compared to TEB's 14.84 m. In most cases, DWA found a more direct route to the goal, which may suggest that it is more efficient in terms of distance traveled. However, TEB occasionally provided shorter paths, particularly noticeable in Goal 2 and Goal 3, which suggests TEB's flexibility in optimizing the path under certain conditions. While DWA and TEB performed similarly in terms of time, DWA showed a slightly longer average duration (46.12 s) than TEB (43.3 s). This difference could indicate that TEB was able to complete paths more quickly despite slightly longer distances, potentially due to a smoother trajectory or fewer adjustments during navigation. TEB generally achieved lower goal errors in the x and y directions. Across most goals, TEB demonstrated more consistent accuracy in reaching the target with lower error values. On average, TEB had a mean x-error of 0.071 m compared to DWA's 0.082 m and a mean y-error of 0.160 m versus DWA's 0.168 m, indicating its advantage in positional accuracy. A notable difference was observed in the yaw error. TEB consistently achieved a significantly lower yaw error, with an average of 0.066 radians, whereas DWA's average yaw error was much higher at 1.2485 radians. This suggests that TEB is more effective in aligning the orientation towards the target, which is essential for applications where precise final orientation is critical. TEB outperformed DWA in positional accuracy and orientation alignment while maintaining a competitive duration. This makes TEB more suitable for tasks requiring high precision in both position and orientation. Conversely, DWA may still be beneficial for applications prioritizing shorter paths, though with a trade-off in orientation precision. While DWA offers slight advantages in path length, TEB's accuracy in positioning and orientation makes it a more reliable option for precision navigation tasks. The selection of the approach should, therefore, depend on whether the priority is minimizing distance or achieving high positional and orientational accuracy.

Table 2. Performance evaluation of local planners in Scenario 2 for each goal.

Scenario/Goals		Path Length (m)		Duration (s)		Goal Error x (m)		Goal Error y (m)		Goal Error Yaw (rad)	
		DWA	TEB	DWA	TEB	DWA	TEB	DWA	TEB	DWA	TEB
2	1	15.97	16.95	56	56	0.039	0.086	0.193	0.222	1.552	0.097
	2	14.42	12.83	48	41	0.183	0.118	0.137	0.016	3.166	0.100
	3	9.48	8.15	29	26	0.178	0.171	0.062	0.022	0.074	0.058
	4	15.77	16.77	49	46	0.021	0.065	0.140	0.195	1.526	0.063
	5	14.86	15.91	44	44	0.066	0.022	0.198	0.193	0.045	0.021
	6	14.53	16.04	37	42	0.040	0.039	0.191	0.192	1.628	0.005
	7	15.23	16.04	48	46	0.052	0.019	0.111	0.219	0.003	0.081
	8	8.03	16.04	58	46	6.516	0.052	0.315	0.227	1.994	0.108
Mean		14.32	14.84	46.12	43.3	0.082	0.071	0.168	0.160	1.2485	0.066

The TEB algorithm demonstrates superior performance over the DWA regarding velocity smoothness. TEB provides more controlled and gradual acceleration, particularly in confined spaces, significantly reducing sudden oscillations and improving maneuverability. In terms of path smoothness, TEB also surpasses DWA by enabling safer, more fluid movements, which are essential for complex or narrow environments. Furthermore, the lower Mean Square Error (MSE) value achieved by TEB indicates greater precision and reliability in adhering to the intended path, reflecting its robustness in path-following tasks. Figure 7 illustrates the trajectories toward each target for the Scenario 2 tests, visually highlighting TEB's smoother navigation and its advantage in dynamic path adjustments. These findings underscore TEB's suitability for applications requiring high levels of accuracy and smoothness in real-time navigation, such as autonomous robotic systems operating in cluttered or constrained environments. The TEB algorithm's ability to maintain stable, smooth motion trajectories minimizes abrupt directional changes, which can be critical in environments where safety, control, and efficiency are paramount.

Table 3 compares the performance of DWA and TEB path-planning algorithms across metrics like path length, duration, and goal errors (in x, y, and yaw) over six goals. On average, TEB produced slightly shorter path lengths (21.96 m) than DWA (23.44 m), with DWA requiring longer paths in early goals, though both converged to similar lengths (around 22 m) in later goals, suggesting TEB's efficiency in minimizing path length. TEB also showed faster completion times, with a mean of 57.3 s compared to DWA's 61 s, notably outperforming DWA in goal 1 (51 vs. 66 s), hinting at TEB's ability to generate smoother trajectories. In goal errors, TEB demonstrated greater accuracy, with a mean error of 0.147 m on the x-axis versus DWA's 0.26 m and outperforming DWA significantly in Goal 1 (0.155 vs. 0.728 m). TEB maintained a lower y-axis error (0.097 m) than DWA (0.842 m), especially in Goal 1, where DWA's error was over three times TEB's (3.831 vs. 0.214 m). In orientation (yaw), TEB had a superior mean error of 0.09 radians compared to DWA's 0.877 radians, with a striking performance in Goal 5 (0.003 vs. 0.074 radians). Overall, TEB achieved shorter paths, faster completion, and lower position and orientation errors, suggesting it is better suited for applications requiring high accuracy and efficiency, while DWA's higher errors, particularly in y-axis positioning and orientation, may limit its use in precision-dependent navigation.

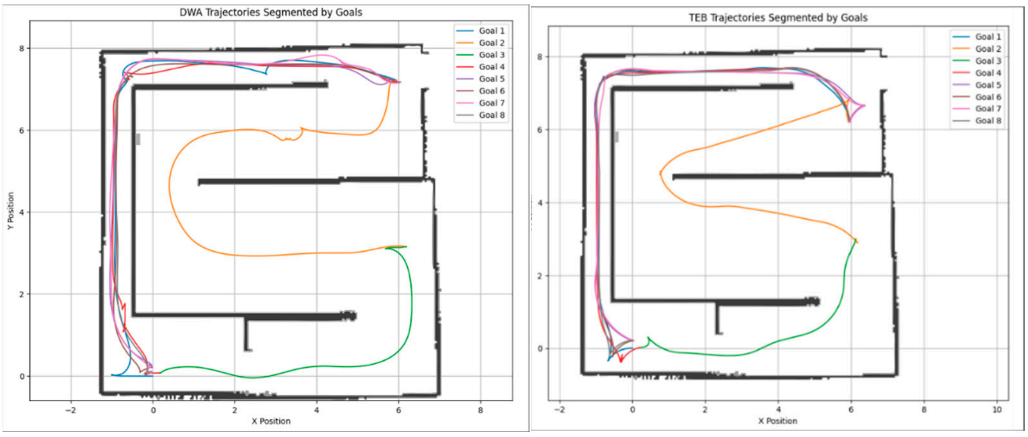

Figure 7. Scenario 2 with 8 Goal Waypoint.

Table 3. Performance evaluation of local planners in the narrow with obstacle scenario.

Scenario/Goals		Path Length (m)		Duration (s)		Goal Error x (m)		Goal Error y (m)		Goal Error Yaw (rad)	
		DWA	TEB	DWA	TEB	DWA	TEB	DWA	TEB	DWA	TEB
3	1	12.28	21.51	66	51	0.728	0.155	3.831	0.214	1.074	0.094
	2	15.52	22.94	50	73	0.151	0.146	0.140	0.025	1.596	0.248
	3	23.69	22.04	64	57	0.120	0.178	0.067	0.257	0.007	0.062
	4	23.68	22.04	64	54	0.112	0.067	0.168	0.023	1.638	0.085
	5	22.96	21.76	61	56	0.189	0.213	0.007	0.018	0.074	0.003
	6		21.49		53		0.124		0.046		0.053
Means		23.44	21.96	61	57.3	0.26	0.147	0.842	0.097	0.877	0.090

The performance comparison between the DWA and TEB local planners across scenarios with and without obstacles highlights key distinctions in path length, duration, velocity smoothness, path smoothness, and Mean Squared Error (MSE) as shown in Table 4. In terms of path length, DWA covered shorter distances in both scenarios—108.30 m vs. TEB's 118.79 m without obstacles and 98.86 m vs. TEB's 131.38 m with obstacles—suggesting that DWA generates more direct paths. TEB, however, completed the obstacle-free path faster (347 s vs. DWA's 369 s), although this advantage diminished in obstacle scenarios, with DWA taking 305 s compared to TEB's 344 s. Velocity smoothness favored TEB, which maintained lower acceleration values in both settings (e.g., 0.08828 m/s^2 vs. DWA's 0.11441 m/s^2 without obstacles), indicating smoother speed transitions, crucial in complex environments. For path smoothness, both planners were similar in obstacle-free conditions, but with obstacles, DWA's path smoothness dropped significantly (0.1421 m^2 vs. TEB's 0.00154 m^2), reflecting more abrupt movements around obstacles. TEB also consistently recorded a lower MSE, particularly in obstacle scenarios (0.0004 m^2 vs. DWA's 4.3462 m^2), underscoring its alignment with the desired path and adaptability. Overall, while DWA's shorter paths and durations are advantageous in open spaces, TEB's superior path smoothness, velocity consistency, and accuracy make it better suited for navigating obstacle-rich environments.

Table 4. Performance Evaluation of Local Planners in the narrow room without (scenario 2) and with (scenario 3) Obstacle.

Scenario	Path Length (m)		Duration (s)		Velocity Smoothness (m/s^2)		Path Smoothness (m^2)		MSE (m^2)	
	DWA	TEB	DWA	TEB	DWA	TEB	DWA	TEB	DWA	TEB
2	108.30	118.79	369	347	0.11441	0.08828	0.001531	0.00153	0.0220	0.0202
3	98.861	131.38	305	344	0.14014	0.08318	0.1421	0.00154	4.3462	0.0004

TEB algorithm successfully covered six goals with a total length of 131.38 m, while the DWA only reached three goals. TEB excelled in the velocity smoothness parameter due to its quick reaction to obstacles, keeping the velocity smooth and controlled. In path smoothness, TEB was more effective in plotting a path outside the global map than DWA, which was slower in adjusting the path. The MSE value of TEB is also much smaller, indicating consistency and efficiency in selecting the optimal path and adapting to the dynamic environment. The plot of each goal's journey for the Obstacle Class arena test is shown in Figure 8.

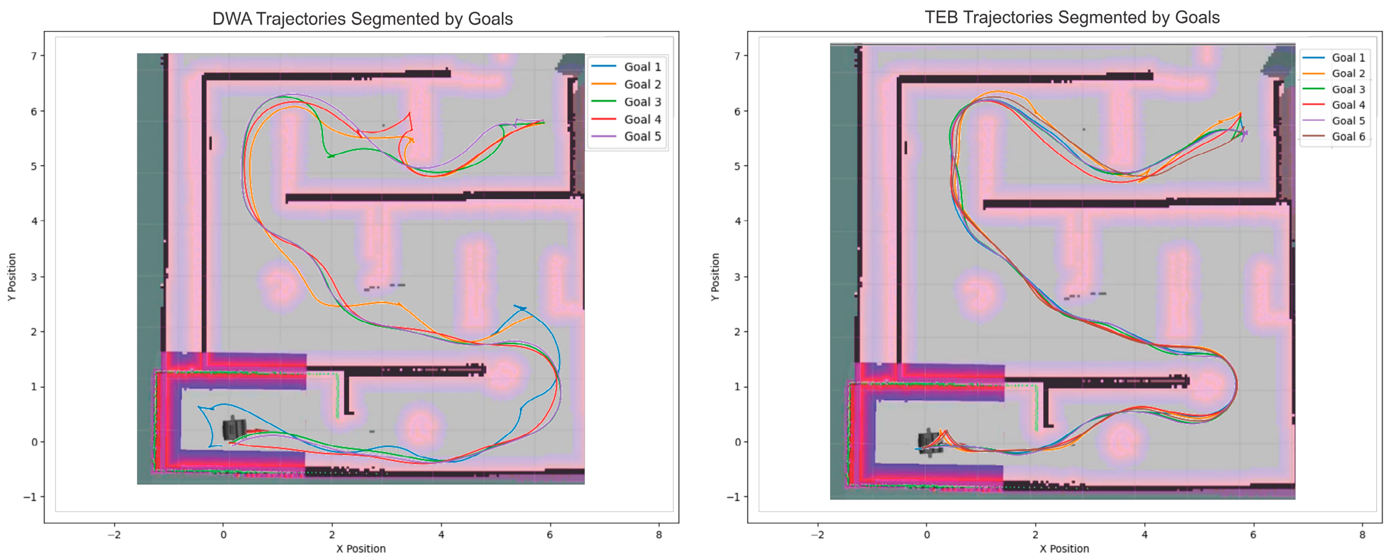

Figure 8. Scenario 3 with 6 Goal Waypoint.

TEB algorithm performed better than the DWA in the face of new obstacles. TEB instantly adapts by planning a new path, avoiding collisions, and improving navigation efficiency. In contrast, DWA often almost hit obstacles that were not recorded on the global map and took longer to perform recovery behaviors, such as backtracking and re-planning the path. The TEB successfully completed six goals, while the DWA had problems with the first two goals and delayed reactions to obstacles. This shows that TEB excels in environments with many obstacles and dynamic conditions, while DWA is more suitable for simple environments. Overall, the navigation results show that the TEB algorithm is superior in various aspects of navigation performance compared to the DWA algorithm. TEB is not only more effective in avoiding obstacles and maintaining smooth speed but also more efficient in path planning, especially in dynamic and complex environments. The

TEB algorithm shows better ability in achieving navigation goals with safer and optimized paths, making it a better choice for applications in diverse and challenging environments.

4. Conclusions

The performance evaluation of local planner algorithms demonstrates that the Time Elastic Band (TEB) algorithm outperforms the Dynamic Window Approach (DWA) across various test scenarios. TEB consistently exhibits better path efficiency, smoothness, and navigation consistency. In Scenario 1, TEB achieved a shorter path length, reduced position and yaw errors, and improved velocity and path smoothness compared to DWA. Similarly, in Scenario 2, TEB generated shorter paths with faster completion times and smoother trajectories. In Scenario 3, TEB continued to excel with shorter path lengths, reduced durations, and significantly smoother and more consistent navigation metrics. These results highlight TEB's superior performance, particularly in environments with obstacles and dynamic conditions, making it a more reliable choice for autonomous navigation tasks.

Author Contributions: Conceptualization, A.T.; Methodology, A.T.; Software, M.A.F.; Validation, M.A.F.; Formal analysis, B.M.W.; Investigation, B.M.W.; Data curation, A.T. and M.A.F.; Writing—original draft, A.T.; Writing—review & editing, B.M.W.; Supervision, A.T.; Project administration, N.A.; Funding acquisition, N.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded through the Academic Leadership Grant, No. 1518/UN6.3.1/PT.00/2024 awarded to Nursanti Anggriani and supported by the Department of Electrical Engineering, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Indonesia.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Huang, Y.; Chen, X.; Liu, J. Automation in repetitive tasks in industrial settings. *J. Ind. Eng.* **2019**, *45*, 123–136.
2. Jones, T.; Smith, R.; Johnson, K. Cost reduction and productivity enhancement with autonomous robots. *Int. J. Autom.* **2021**, *52*, 232–245.
3. Smith, A.; Johnson, B. Autonomous drones in agriculture: Yield improvements and sustainability. *Precis. Agric.* **2020**, *10*, 298–315.
4. Lee, H.; Kim, S.; Park, M. Improving worker safety through autonomous robotics in hazardous environments. *Saf. Sci.* **2020**, *64*, 412–420.
5. Dufresne, T.; Dubois, L. Autonomous Robots in Hazardous Environments: Replacing Human Labor in Dangerous and Physically Demanding Tasks. *J. Robot. Mechatron.* **2019**, *31*, 220–232.
6. Zhang, L.; Wang, T. Environmental benefits of autonomous robots in sustainable agriculture. *J. Environ. Sci. Technol.* **2022**, *58*, 98–110.
7. Romanov, A.M. A navigation system for intelligent mobile robots. In Proceedings of the IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Saint Petersburg, Russia, 28–31 January 2019; pp. 652–656.
8. Yang, H. A multi-robot formation platform based on an indoor global positioning system. *Appl. Sci.* **2019**, *9*, 1165. [CrossRef]
9. Chempolil, S.S. Design of a medical prototype robot for nurse assistance. In Proceedings of the 2021 Seventh International Conference on Bio Signals, Images, and Instrumentation (ICBSII), Chennai, India, 25–27 March 2021; pp. 1–5.
10. Turnip, A.; Tampubolon, G.M.; Ramadhan, S.F.; Nugraha, A.V.; Trisanto, A.; Novita, D. Development of medical robot Covid-19 based 2D mapping LIDAR and IMU sensors. In Proceedings of the 2021 IEEE International Conference on Health, Instrumentation & Measurement, and Natural Sciences (InHeNce), Medan, Indonesia, 14–16 July 2021; pp. 1–4.
11. Manuky, Z.J. Lokalisasi Berbasis Marvelmind-Lidar-Imu Menggunakan Extended Kalman Filter Untuk Sistem Navigasi Otonom Robot Medis. Bachelor's Thesis, Universitas Padjadjaran, Bandung, Indonesia, 2023.
12. Rafly. Deteksi Rintangan Untuk Menghindari Tabrakan di Robot Medis Otonom Dengan Kamera LiDAR. Bachelor's Thesis, Universitas Padjadjaran, Bandung, Indonesia, 2023.

13. Turnip, A. Development of autonomous medical robot based artificial intelligence and Internet of Things. *Int. J. Artif. Intell.* **2024**, *22*, 20–37.
14. Rivai, M.; Hutabarat, D.; Nafis, Z.M. 2D mapping using omni-directional mobile robot equipped with LiDAR. *TELKOMNIKA Telecommun. Comput. Electron. Control.* **2020**, *18*, 1467–1474. [CrossRef]
15. Wang, H.; Lou, X.; Cai, Y.; Li, Y.; Chen, L. Real-time vehicle detection algorithm based on vision and LiDAR point cloud fusion. *J. Sens.* **2019**, *2019*, 8473980. [CrossRef]
16. Turnip, A.; Sudrajat, A.W.; Le Hoa, N.; Dharma, A.; Joelianto, E. Interactive development of medical robot using voice chatbot based on deep learning. In Proceedings of the 2023 29th International Conference on Telecommunications (ICT), Toba, Indonesia, 8–9 November 2023; pp. 1–6. [CrossRef]
17. Suryawan, D.; Adinandra, S.; Arifianto, J.; Nugroho, E.S.; Masykur, L.A.; Purnama, R.H. Rancang bangun robot pelayan medis untuk pasien karantina COVID-19 dengan kendali berbasis Android. *JTT J. Teknol. Terapan* **2021**, *7*, 68–76. [CrossRef]
18. Nurhayati, S. Sistem navigasi robot pembawa nampan obat pasien berbasis Internet of Things. In Proceedings of the Electro National Conference (ENACO) Politeknik Negeri Sriwijaya, Palembang, Indonesia, 1 June 2021; pp. 248–255.
19. Zhao, J.; Liu, S.; Li, J. Research and implementation of autonomous navigation for mobile robots based on SLAM algorithm under ROS. *Sensors* **2022**, *22*, 4172. [CrossRef] [PubMed]
20. Looi, C.Z.; Ng, D.W. A study on the effect of parameters for ROS motion planner and navigation system for indoor robot. *J. Electr. Comput. Eng. Res.* **2021**, *1*, 29–36.
21. Chikurtev, D.; Chivarov, N.; Chivarov, S.; Chikurteva, A. Mobile Robot Localization and Navigation Using LIDAR and Indoor GPS. *IFAC PapersOnLine* **2021**, *54*, 351–356. [CrossRef]
22. Sharma, N.; Pandey, J.K.; Mondal, S. A Review of Mobile Robots: Applications and Future Prospect. *Int. J. Precis. Eng. Manuf.* **2023**, *24*, 1695–1706. [CrossRef]
23. Loganathan, A.; Ahmad, N.S. A Systematic Review on Recent Advances in Autonomous Mobile Robot Navigation. *Eng. Sci. Technol. Int. J.* **2023**, *40*, 101343. [CrossRef]
24. Liu, L.; Wang, X.; Yang, X.; Liu, H.; Li, J.; Wang, P. Path Planning Techniques for Mobile Robots: Review and Prospect. *Expert. Syst. Appl.* **2023**, *227*, 120254. [CrossRef]
25. Hu, H.; Zhang, K.; Tan, A.H.; Ruan, M.; Agia, C.G.; Nejat, G. A Sim-to-Real Pipeline for Deep Reinforcement Learning for Autonomous Robot Navigation in Cluttered Rough Terrain. *IEEE Robot. Autom. Lett.* **2021**, *6*, 6569–6576. [CrossRef]
26. Al-Kamil, S.J.; Szabolcsi, R. Optimizing Path Planning in Mobile Robot Systems Using Motion Capture Technology. *J. Robot. Syst.* **2024**, *22*, 102043. [CrossRef]
27. Balasubramanian, E.; Elangovan, E.; Tamilarasan, P.; Kanagachidambaresan, G.R.; Chutia, D. Optimal Energy Efficient Path Planning of UAV Using Hybrid MACO-MEA* Algorithm: Theoretical and Experimental Approach. *J. Ambient. Intell. Hum. Comput.* **2022**, *25*, 1–21. [CrossRef]
28. Karur, K.; Sharma, N.; Dharmatti, C.; Siegel, J.E. A Survey of Path Planning Algorithms for Mobile Robots. *Vehicles* **2021**, *3*, 448–468. [CrossRef]
29. Liang, J.; Wang, S.; Wang, B. Online Motion Planning for Fixed-Wing Aircraft in Precise Automatic Landing on Mobile Platforms. *Drones* **2023**, *7*, 324. [CrossRef]
30. Yuan, H.; Li, H.; Zhang, Y.; Du, S.; Yu, L.; Wang, X. Comparison and Improvement of Local Planners on ROS for Narrow Passages. In Proceedings of the 2022 International Conference on High Performance Big Data and Intelligent Systems HPBD&IS, Tianjin, China, 10–11 December 2022. [CrossRef]
31. Naotunna, I.; Wongratanaphisan, T. Comparison of ROS Local Planners with Differential Drive Heavy Robotic System. In Proceedings of the 2020 International Conference on Advanced Mechatronic Systems ICAMechS, Hanoi, Vietnam, 10–13 December 2020. [CrossRef]
32. Canh, T.N.; HoangVan, X.; Chong, N.Y. Enhancing Social Robot Navigation with Integrated Motion Prediction and Trajectory Planning in Dynamic Human Environments. *arXiv* **2024**, arXiv:2411.01814. [CrossRef]
33. Kulathunga, G.; Yilmaz, A.; Huang, Z. Resilient Timed Elastic Band Planner for Collision-Free Navigation in Unknown Environments. *arXiv* **2024**, arXiv:2412.03174.
34. Wu, J.; Ma, X.; Peng, T.; Wang, H. An Improved Timed Elastic Band (TEB) Algorithm of Autonomous Ground Vehicle (AGV) in Complex Environment. *Sensors* **2021**, *21*, 8312. [CrossRef] [PubMed]

35. Mansakul, T.; Fan, I.-S.; Tang, G. Navigation for a mobile robot to inspect aircraft. In Proceedings of the 7th International Young Engineers Forum (YEF-ECE), Lisbon, Portugal, 7 July 2023. [CrossRef]
36. Arce, D.; Solano, J.; Beltrán, C. A Comparison Study between Traditional and Deep-Reinforcement-Learning-Based Algorithms for Indoor Autonomous Navigation in Dynamic Scenarios. *Sensors* **2023**, *23*, 9672. [CrossRef] [PubMed]