

Article

Identification of Differential Drive Robot Dynamic Model Parameters

Michał Siwek * , Jarosław Panasiuk , Leszek Baranowski , Wojciech Kaczmarek , Piotr Prusaczyk , and Szymon Borys 

Faculty of Mechatronics, Armament and Aerospace, Military University of Technology, Kaliskiego 2 Street, 00-908 Warsaw, Poland

* Correspondence: michal.siwek@wat.edu.pl

Abstract: The paper presents the identification process of the mathematical model parameters of a differential-drive two-wheeled mobile robot. The values of the unknown parameters of the dynamics model were determined by carrying out their identification offline with the Levenberg–Marguardt method and identification online with the Recursive least-squares method. The authors compared the parameters identified by offline and online methods and proposed to support the recursive least squares method with the results obtained by offline identification. The correctness of the identification process of the robot dynamics model parameters, and the operation of the control system was verified by comparing the desired trajectories and those obtained through simulation studies and laboratory tests. Then an analysis of errors defined as the difference between the values of reference position, orientation and velocity, and those obtained from simulations and laboratory tests was carried out. On its basis, the quality of regulation in the proposed algorithm was determined.

Keywords: mobile robot; ROS; AGV; identification; adaptation; trajectory tracking



Citation: Siwek, M.; Panasiuk, J.; Baranowski, L.; Kaczmarek, W.; Prusaczyk, P.; Borys, S. Identification of Differential Drive Robot Dynamic Model Parameters. *Materials* **2023**, *16*, 683. <https://doi.org/10.3390/ma16020683>

Academic Editors: Piotr Wróblewski and Raul D.S.G. Campilho

Received: 8 December 2022

Revised: 30 December 2022

Accepted: 3 January 2023

Published: 10 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

By Industry 4.0 standards, internal transportation is becoming one of the most important areas undergoing transformation. Due to the increasingly scarce labor resources and the increasing cost of transporting products in the production process, transportation is considered in the category of losses. Therefore, there is a need to minimize the costs associated with the internal transportation by reducing the distances and times of product movement. One of the solutions to this problem is the introduction of autonomous transport robots that will accurately, quickly, and safely carry out the tasks assigned to them. Automated guided vehicles are mainly used to support material handling operations—in the warehouse, in production and at the intersection of these areas [1]. This can involve a range of tasks, from the timely delivery of parts to the production line to round-the-clock transportation. Full automation of the production process makes it possible to ensure smooth manufacturing of products along the entire process line. Therefore, in line with the idea of Industry 4.0, more and more often, mobile cart units or AGV (Automated Guided Vehicle) robots appear at the end of the production line to transport and manage products. The market for mobile robots is still at an early stage of development. However, for several years there has been an exponential increase in the implementation of transport mobile robots in the industrial sector. This is due to the fact that more and more companies are noticing the possibilities of this type of vehicles to automate internal transport (Figure 1).

When deciding to integrate mobile robots into a production process at a company, it is necessary to define the efficiency of the transport system based on the following assumptions: the number of robots, the number of logistics points, the location of the navigation paths along which they will move and the frequency of changeovers.

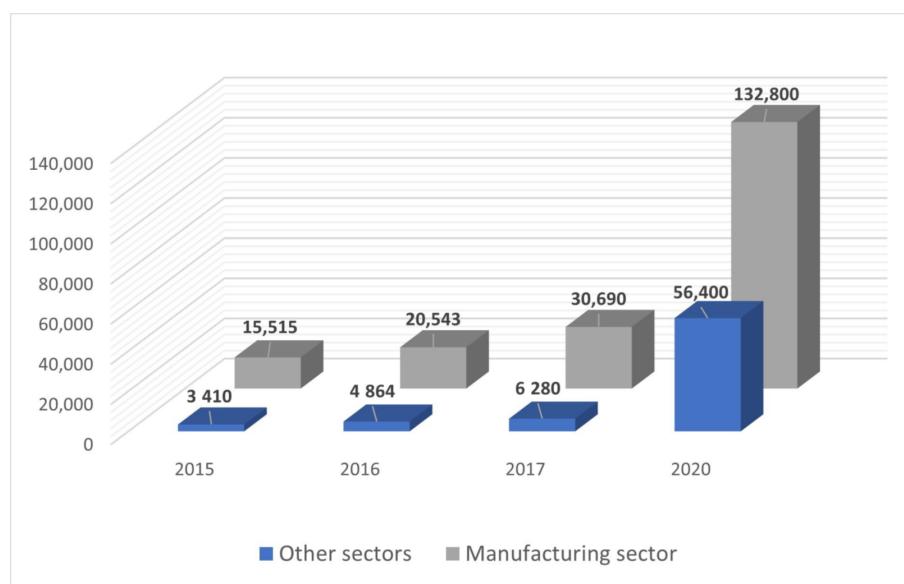


Figure 1. The number of mobile robots used in industry [2].

The first automatic carts implementing operator-free driving appeared in the 1950s. These were single implementations primarily in the automotive industry. Over time, the advantages of AGV deployments were increasingly recognized and automatic vehicles began to appear in other industries as well, especially in applications related to in-plant logistics. Since then, AGV robot technology has been developing rapidly, as indicated by the fact that there are several types of AGVs [3,4]. The main characteristic of AGV robots is their driving system. One can distinguish:

- three-wheeled systems (two independent driven wheels, one non-driven castor wheel);
- four-wheeled systems:
 - two independent driven standard wheels mounted on the front of the robot and two castor wheels mounted on the rear of the robot;
 - four independently controlled Swedish wheels;
- six-wheeled systems (two independent driven standard wheels mounted in the center of the robot frame and four castor wheels mounted on the corners of the robot frame).

All vehicles for automatic load handling are regulated by ISO 3691-4:2020 and called automated guided vehicle (AGV) [5]. This standard distinguishes specific names for different types of AGV introduced by manufacturers of these machines. At the beginning, the main division into AGV and AMR (Autonomous Mobile Robots) can be distinguished. AGV technology refers to automated vehicles that move indoors along physically defined paths, requiring the installation of lines or navigation points. This imposes the need to navigate only along a designated path. In case of encountering an obstacle, it requires stopping and waiting for its removal. Changing the path requires changing the navigation line and other markings. An AMR solution involves autonomous vehicles, often called intelligent vehicles, equipped with 2D or 3D scanners, moving based on a self-created map along a self-determined path between destination points, with the ability to avoid obstacles. The use of AMR technology allows for improved production flexibility. The robot itself determines the optimal route for a given task [6]. Changing the tasks or destinations assigned to it is very quick and does not require physical interference in the space in which the robot moves.

This work proposes a control system for the tracking motion of a mobile robot with two differentially controlled drive wheels and two support wheels. The control system was realized using a kinematic and dynamic controller, where the input and output signals are linear and angular velocity. In the presented approach, the problem is the parameters of the mathematical model of the robot dynamics, which are not available to the user. Therefore,

their identification is carried out. In this work, the authors performed offline identification using the Levenberg–Marguardt method based on a pre-recorded trajectory, and online identification using the Recursive least-squares (RLS) method while moving along a desired trajectory. To verify the correctness of the identified parameters, simulation studies were carried out in the MATLAB/Simulink environment, followed by laboratory tests using a real robot. During the tests, the effect of random initial parameters in the identification process on the accuracy of trajectory tracking was investigated. The parameters identified by online and offline methods were compared. The novelty of proposed method is the usage of the parameters identified by the Levenberg–Marguardt method as the initial parameters in the online identification. The effect of combination of online and offline methods on the accuracy of tracking the desired trajectory was investigated.

In Section 2, the authors presented related works. Section 3 presents the selected research object for which the control system, kinematic and dynamic model were developed. The Section 4 contains the results of the identification process, the design of the control system, and the results of trajectory tracking considering the identified parameters. In Section 5, the authors summarized the work carried out.

2. Related Works

AGV robots are a good solution for plants that are characterized by low variability of the production process and not very large distances between logistic destinations. In the case of frequent changes in the production process, dynamically changing environment in the plant (e.g., moving workers) [7], disturbances [8,9] and obstacles [10], a safe and flexible control system is required to allow for environmental changes [11–13] or reprogramming of the carts for other tasks. Searching for the most appropriate path planning algorithm according to user requirements can be challenging given the large number of examples existing in the literature [14–19]. Furthermore, in the case of AGV robots, the precision of the final positioning is crucial for successful loading and unloading of transported goods in typical industrial applications [20]. Designers of robotics systems use advanced tools to assist in the design process of control systems [21–25] to provide the expected functionality of the robot. The working environment has a particular impact on the change in the operating parameters of the robot. Especially when it is difficult [26,27] such as dusty rooms or industrial and sewage pipelines or mines [28].

A way to obtain unknown values of robot model parameters is to identify them. The work [29–32] presents a methodology for offline identification based on an inverse dynamics model of robots. The offline identification of the dynamics parameters is sufficient only in the case of robots that will not be expanded with additional modules or will not transport loads. In the case of AGV robots transporting loads, their mass and moment of inertia change, which means that the parameters of the model are not constant. When designing the control algorithm, it is necessary to take into account the possibility of changing the model parameters based on their identification using adaptive methods (online identification) [33–38]. The most widely used adaptive methods is the Recursive least squares (RLS) method, as presented in papers [39,40]. The example is for trajectory tracking of an autonomous ground vehicle. To estimate the unknown parameters of the system, the authors of [39] proposed a set-membership-based estimator that provides a nonincreasing estimation error. In the proposed predictive controller (MPC) scheme, the authors used the RLS method to improve prediction accuracy by introducing a robustness constraint on the algorithm to parametric and additive uncertainties. The robustness factor is determined offline based on a set of invariant parameters. The size of this set is updated online. The proposed method is a bit computationally complex, so it requires a high-performance on-board computer. Another example of the use of the RLS method is the work [41], whose authors presented a positioning system for a mobile robot using a stereoscopic camera, an IMU, and an Ultra Wideband (UWB) network containing five positioning markers. The system can be extended with additional positioning markers, whose unknown position in the robot's environment is estimated using the RLS method and Kalman filter. The proposed

system works well in general locating the robot in an unknown environment. However it comes with a large positioning error, which unfortunately is not acceptable for transportation tasks. An interesting approach, consistent with the authors of this work, was demonstrated by the authors of papers [41–44]. They presented the problem of trajectory tracking by a nonholonomic mobile robot without a global positioning system. The main difficulty pointed out in this work is localizing the robot using only its on-board sensors. The authors proposed an adaptive trajectory tracking controller without orientation and velocity measurements. Their method is not based on the estimation of the robot's position and velocity from the image of an omnidirectional camera mounted on the robot. The paper [43] presents a fusion of an inertial sensor of six degrees of freedom (6-DoF) which comprises the 3-axis of an accelerometer and the 3-axis of a gyroscope, and a vision to determine a low-cost and accurate position for an autonomous mobile robot. For vision, a monocular vision-based object detection algorithm speeded-up robust feature (SURF) and random sample consensus (RANSAC) algorithms were integrated and used to recognize a sample object in several images taken. In contrast to the work of [42], the authors of this work, present a control concept that not use robot-mounted peripheral devices such as laser scanners and cameras. The tracking control concept is based only on odometry data and inertial measurements.

3. Materials and Methods

The considerations presented in this paper include the TURTLEBOT 2 laboratory robot (Figure 2) [45], which is a low-cost training set, used to building and testing wheeled mobile robot control systems. The platform is equipped with a two-wheeled differential drive and basic sensors such as a laser scanner, a 640×480 resolution RGB video camera (used for image processing, color, and texture recognition), an IR camera (depth information processing, distance measurement in the range of 0.4–6.5 m), four directional microphones, and an accelerometer. An advantage of using the robot in this task is its open control system based on ROS (Robot Operating System) [46–48], which allows quick launch of the platform and integration with sensors [45]. In addition, the drive system of the TURTLEBOT 2 robot is the basis for expansion to the drive systems discussed in the introduction, commonly used in industrial AGVs. Transferring and adapting the control system to another chassis will be extremely fast and simple.

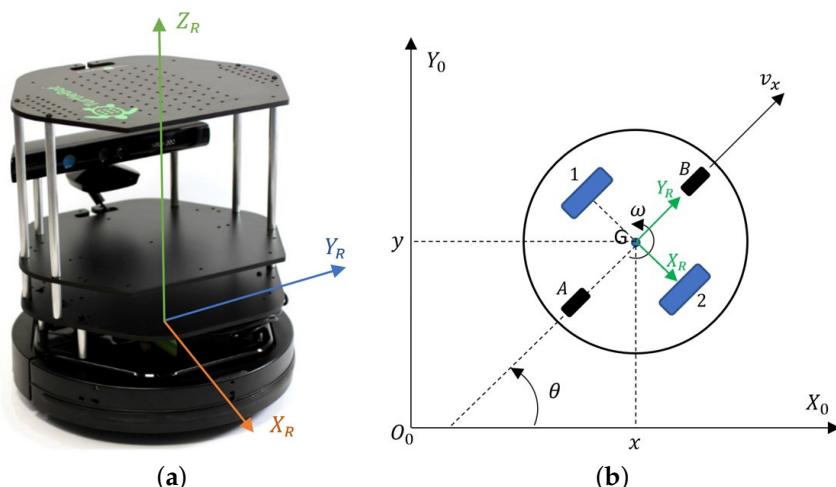


Figure 2. (a) Isometric view of the TURTLEBOT 2 robot with the coordinate system; (b) kinematic structure to determine the kinematic model.

3.1. Kinematic and Dynamic Robot Model

In order to determine the kinematic and dynamic model of the TURTLEBOT 2 robot, the authors used the diagrams shown in Figures 2 and 3.

The $O_0X_0Y_0$ coordinate system shown in Figure 2b is a global, fixed reference system. The x, y coordinates define the positions and the angle θ define the orientation of the robot

in the global coordinate system. A movable coordinate system GX_RY_R with an origin at the geometric center of the robot G , is associated with the base of the robot. The vector v determines the linear velocity, while ω determines the angular velocity of the robot. The numbers 1 and 2 denote the drive wheels, while the letters A and B denote the supporting wheels. The kinematics of the mobile robot TURTLEBOT 2 is described based on [49–52] by Equation (1) in the form in which the control signals are the linear and angular velocities of the robot:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (1)$$

While analyzing the literature in view of control systems, it can be noted that most of the proposed controllers performing the trajectory tracking task consider only the kinematics of the robot. This is an acceptable solution, but in the case of tasks requiring high positioning accuracy and moving at high speed, it is necessary to consider the dynamics of the robot in the control system. In the literature, dynamic models of robots are popular, taking into account torque or motor voltages as steering signals [53–56]. A much better solution is to use a dynamics model that takes the linear and angular velocities of the robot as control signals [57,58]. This is how commercially available mobile robots are usually controlled. Figure 3 shows a schematic of the TURTLEBOT 2 robot used to determine the dynamics model.

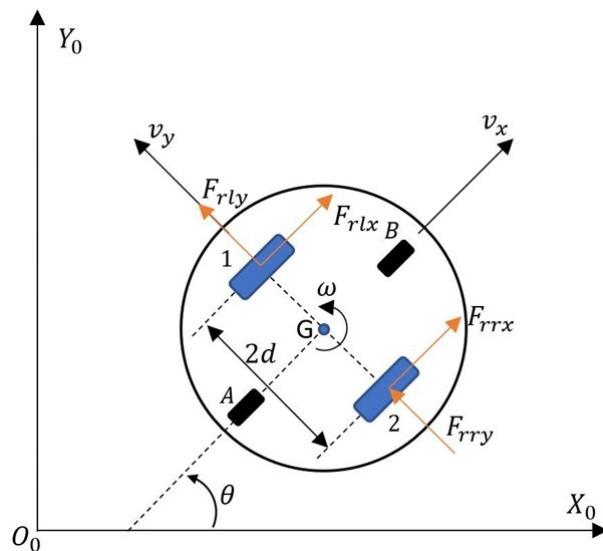


Figure 3. TURTLEBOT 2 robot schematic for developing the dynamics model.

The point G (Figure 3) is not only the robot's center of mass and center of rotation, but also the tracking point for the desired trajectory. The robot's velocities are denoted as: v_x —forward velocity, v_y —lateral velocity, ω —angular velocity. In Figure 3, the driving forces of the robot are denoted as F_{rrx} , F_{rry} —the longitudinal and lateral force of the right wheel and F_{rlx} , F_{rly} —the longitudinal and lateral force of the left wheel. The dynamics model of the robot was formulated by performing a law of conservation of linear and angular momentum, in the coordinate system associated with the robot, following [59]:

$$\begin{aligned} \sum F_x &= F_{rlx} + F_{rrx} = m(\dot{v}_x - v_y \omega) \\ \sum F_y &= F_{rly} + F_{rry} = m(\dot{v}_y - v_x \omega) \\ \sum M_z &= I_z \omega = \frac{d}{2}(F_{rrx} - F_{rlx}) \end{aligned} \quad (2)$$

where:

m —mass of the robot,

I_z —moment of inertia of the robot with respect to the axis passing through the point G .

The transformation of the system of Equation (2) (see Appendix A) is the dynamic model of the TURTLEBOT 2 robot as presented:

$$\begin{bmatrix} \dot{v}_x \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -\frac{\sigma_3}{\sigma_1} v_x \\ -\frac{\sigma_4}{\sigma_2} \omega \end{bmatrix} + \begin{bmatrix} \frac{1}{\sigma_1} & 0 \\ 0 & \frac{1}{\sigma_2} \end{bmatrix} \begin{bmatrix} v_{xd} \\ \omega_d \end{bmatrix} \quad (3)$$

The model coefficients $\sigma_1 \dots \sigma_4$ contain parameters that are very difficult for a laboratory robot user to obtain. These coefficients are functions of the physical parameters of the robot related to drives, gears, internal controls, friction, and are usually not even included in the manuals or technical sheets of the equipment, so their identification was carried out.

3.2. Offline Identification of Robot Dynamic Model Parameters

Before identifying model parameters, it should be checked whether any of the model parameters can be written as a linear combination of two others. In such a case, it would be possible to write the robot dynamics model with the lower number of parameters. Analyzing the developed mathematical model of the robot, the linear independence between the parameters $\sigma_1 \dots \sigma_4$ is not clearly visible, because some physical parameters affect more than one parameter σ_n . A detailed analysis is necessary, which was carried out in the work [58] showing that the model parameters σ_1 to σ_4 are independent, that is, they cannot be written as their linear combination. Ultimately, it is necessary to carry out identification of each of the parameters $\sigma_1 \dots \sigma_4$.

Offline identification of the parameters of the robot's dynamics model was carried out using the Levenberg-Marguardt method and measurements of the real test trajectory made by the robot at known constant values of linear and angular velocity. Identification is an iterative process, so in order to perform the first iteration, initial starting values must be entered at the beginning.

To evaluate the impact of starting parameter error on identification time and accuracy, the process was carried out for random initial values of the identified parameter. Starting parameters were described by a normal distribution with expectation value $\mu = 1$ variance $\sigma^2 = 1$ in the interval $\langle 0, 1 \rangle$. The next step was to determine the mean value, the variance and the standard deviation from all samples of the identified parameter.

For the Levenberg-Marguardt method, the minimization criterion takes the form [60,61]:

$$\min_d \|J(x)d + F(x)\|_2^2 \quad (4)$$

where $J(x)$ is the Jacobian of the vector $F(x)$, and d is the direction of the search for the minimum of the function determined according to the rule:

$$[J(x)^T J(x) + \alpha I]d = -J(x)F(x), \quad \text{for } \alpha \geq 0 \quad (5)$$

In the identification process, 50 tests were carried out. Table 1 contains sample values of the identified parameters of the robot dynamics model.

The low values of the standard deviation of the identified parameters suggest the correctness of the identification process for the Levenberg-Marguardt method, which was verified in simulation and laboratory tests, as presented in Sections 4.2–4.5.

Table 1. Example values of parameters identified by the Levenberg-Marguardt method.

Test No.		σ_1	σ_2	σ_3	σ_4	Identification Time [s]
1	Initial values	0.16	0.52	0.96	0.37	
	Identified values	4.09	5.46	0.97	0.29	6.65
2	Initial values	0.61	0.26	0.76	0.29	
	Identified values	3.14	4.06	0.81	0.24	6.01
3	Initial values	0.59	0.55	0.92	0.29	
	Identified values	3.14	4.17	0.83	0.24	6.40
4	Initial values	0.65	0.45	0.55	0.30	
	Identified values	3.49	4.63	0.85	0.24	5.92
5	Initial values	0.38	0.56	0.08	0.05	
	Identified values	3.37	4.56	0.84	0.26	5.65
	Average	3.47	4.58	0.87	0.25	6.13
	Variance	0.15	0.30	0.00	0.00	
	Standard deviation	0.35	0.49	0.06	0.02	

3.3. Online Identification of Robot Model Parameters

In a dynamic system, there may be uncertainties in the robot parameters determined by offline methods; therefore, in [40,57,58], adaptation of the robot dynamics model parameters during motion was proposed. In the paper [58], two laws of adaptation of robot dynamics model parameters were presented. The system discussed in this paper uses parameter adaptation based on the expression:

$$\dot{\hat{\sigma}} = \gamma^{-1} A^T \Delta - \gamma^{-1} \Gamma \hat{\sigma} \quad (6)$$

where $\gamma \in R^{4 \times 4}$ is the gain matrix for parameter adaptation, and $\hat{\sigma}$ is the matrix of estimated parameters of the dynamics model, A is the matrix of velocities and accelerations (see Appendix B), and Δ is the matrix of lapses in the form:

$$\Delta = \begin{bmatrix} v_{xd} \\ \omega_d \end{bmatrix} - \begin{bmatrix} v_x \\ \omega \end{bmatrix} \quad (7)$$

The adaptation law (6) by introducing a gain matrix $\Gamma \in R^{4 \times 4}$ gives the possibility to disable the updating of the model parameters when the error is smaller than a preset limit.

In order to check the effect of unknown model parameters on the initial tracking error of the desired trajectory, two studies were conducted in which random values were used as initial parameters in the identification process. Then the values were replaced with values determined by the offline method.

In the first study, online identification was carried out for random initial parameters described by a normal distribution with expectation value $\mu = 1$ and variance $\sigma^2 = 1$ in the interval $(0, 1)$. The simulation time was set at 120 s. An example of the distribution of the identified parameters $\sigma_1 \dots \sigma_4$ is shown in Figure 4. The values of the starting parameters and those identified by the RLS method are shown in Table 2.

In the second test of the identification process by the RLS method, the parameters identified by the Levenberg-Marguardt method were used as initial parameters. Figure 5 shows the distribution of the identified parameters $\sigma_1 \dots \sigma_4$. The values of the initial and identified parameters by the Levenberg-Marguardt-assisted RLS method are shown in Table 3.

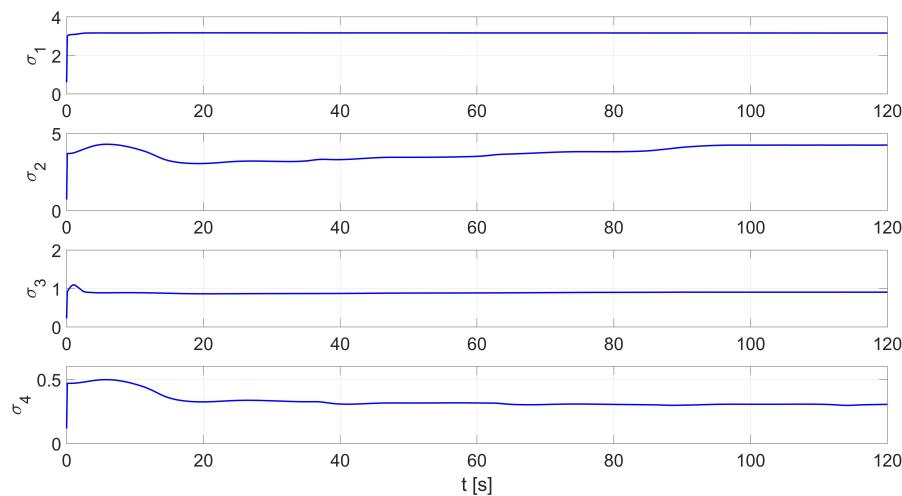


Figure 4. An example of the RLS identification process (test No. 1 from Table 2).

Table 2. Example values of model parameters identified by the RLS method with random values of initial parameters.

Test No.		σ_1	σ_2	σ_3	σ_4
1	Initial values	0.6	0.71	0.22	0.12
	Identified values	3.16	4.15	0.92	0.3
2	Initial values	0.93	0.73	0.49	0.58
	Identified values	3.26	4.51	0.89	0.22
3	Initial values	0.24	0.46	0.96	0.55
	Identified values	3.84	4.34	0.86	0.25
4	Initial values	0.52	0.23	0.49	0.62
	Identified values	3.37	3.87	0.78	0.28
5	Initial values	0.68	0.39	0.37	0.99
	Identified values	3.32	4.16	0.76	0.24
Average		3.39	4.21	0.84	0.26
Variance		0.06	0.05	0.00	0.00
Standard deviation		0.24	0.21	0.06	0.03

Table 3. Values of parameters identified by offline-assisted RLS methods.

Test No.		σ_1	σ_2	σ_3	σ_4
1	Initial values	3.47	4.58	0.87	0.25
	RLS + offline	3.68	4.49	0.90	0.22
2	RLS + offline	3.66	4.50	0.80	0.22
	RLS + offline	3.62	4.51	0.85	0.23
4	RLS + offline	3.67	4.52	0.89	0.20
	RLS + offline	3.62	4.54	0.83	0.24
Average		3.65	4.51	0.85	0.22
Variance		0.00	0.00	0.00	0.00
Standard deviation		0.03	0.02	0.04	0.01

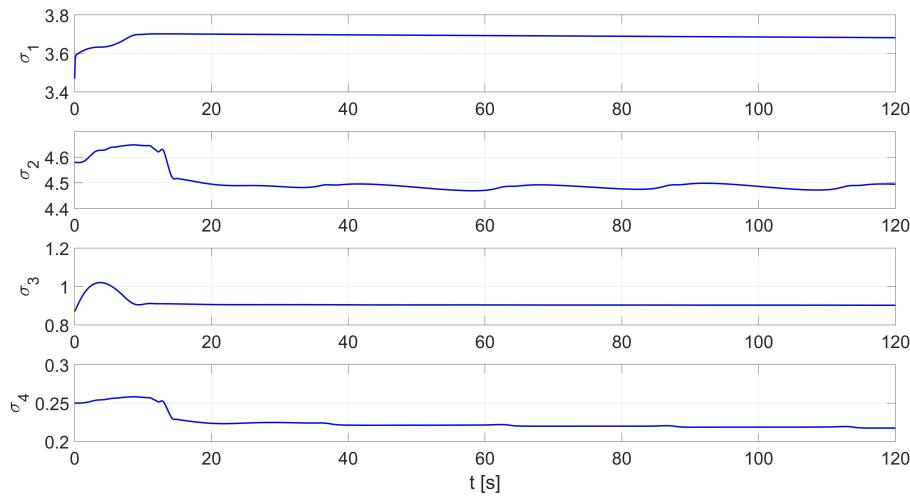


Figure 5. An example of the identification process using the Levenberg-Marguardt-assisted RLS method.

4. Results

The correctness of the TURTLEBOT 2 robot's tracking control algorithm and the correctness of the identification of the parameters of its mathematical dynamic model were checked first by simulation and then by implementation on a real robot. The tests were carried out for an 8-shape trajectory described by a system of equations:

$$\begin{bmatrix} x_d \\ y_d \\ \theta_d \end{bmatrix} = \begin{bmatrix} x_0 + A_x \sin(\omega_x t) \\ y_0 + A_y \cos(\omega_y t) \\ \text{atan2}(\dot{y}_d, \dot{x}_d) + k\pi \end{bmatrix} \quad (8)$$

where: $x_0 = 0$, $y_0 = 0$, $A_x = 2$ [m], $A_y = 2$ [m], $\omega_x = 0.25$ [$\frac{\text{rad}}{\text{s}}$], $\omega_y = 0.125$ [$\frac{\text{rad}}{\text{s}}$], $k = 0$ if the robot moves forward $k = 1$ if the robot moves backwards.

4.1. Kinematic and Dynamic Controller

The implementation of the trajectory tracking task by a robot is always associated with the appearance of errors—the difference between the reference position and the actual position [15,62]. For this purpose, a proportional controller with a feedback loop was proposed, developed based on [63]. Whose task is minimizing the position errors e_x , e_y , e_θ described by Equation (9). It is a transformation of the difference between the desired position and orientation, denoted as $x(t)$, $y(t)$, $\theta(t)$, and the real position and orientation, denoted as $x_r(t)$, $y_r(t)$, $\theta_r(t)$:

$$\begin{bmatrix} e_x(t) \\ e_y(t) \\ e_\theta(t) \end{bmatrix} = \begin{bmatrix} \cos\theta(t) & \sin\theta(t) & 0 \\ -\sin\theta(t) & \cos\theta(t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(t) - x_r(t) \\ y(t) - y_r(t) \\ \theta(t) - \theta_r(t) \end{bmatrix} \quad (9)$$

The kinematic controller proposed based on [63] takes the form:

$$\begin{bmatrix} u_{kv}(t) \\ u_{k\omega}(t) \end{bmatrix} = \begin{bmatrix} -k_1 & 0 & 0 \\ 0 & -\text{sign}(v_r(t))k_2 & -k_3 \end{bmatrix} \begin{bmatrix} e_x(t) \\ e_y(t) \\ e_\theta(t) \end{bmatrix} \quad (10)$$

where $v_r(t)$ —real linear velocity, $u_{kv}(t)$ and $u_{k\omega}(t)$ are the control signals of the kinematic controller (linear and angular velocity) and its gains k_1, k_2, k_3 are determined from the following equations [63]:

$$\begin{aligned} k_1 = k_3 &= 2\varepsilon\omega_n(t), \quad \text{for } \varepsilon \in (0, 1), b > 0 \\ k_2 &= b * |v_r(t)| \\ \omega_n(t) &= \sqrt{\omega_r^2(t) + bv_r^2(t)} \end{aligned} \quad (11)$$

where:

ε —damping coefficient, $\omega_n(t)$ —characteristic frequency, b —additional control coefficient, $\omega_r(t)$ —real angular velocity.

The assumption for the development of the dynamic controller was to treat the control signals from the kinematics controller (linear and angular velocity) expressed by relation (10) as reference signals. The control law for the dynamic controller was developed based on the inverse dynamics (Appendix B), using the parametric model of robot dynamics (3) and the considerations presented in the paper [58]. The control law is described by the equation:

$$\begin{aligned} \begin{bmatrix} u_{dv}(t) \\ u_{d\omega}(t) \end{bmatrix} &= \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} \vartheta_1 \\ \vartheta_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 & v_r(t) & 0 \\ 0 & 0 & 0 & \omega_r(t) \end{bmatrix} \begin{bmatrix} \sigma_1 & \sigma_2 & \sigma_3 & \sigma_4 \end{bmatrix}^T \\ \vartheta_1 &= \dot{\vartheta}(t) + k_v(u_{kv}(t) - v_r(t)) \\ \vartheta_2 &= \dot{\omega}(t) + k_\omega(u_{k\omega}(t) - \omega_r(t)) \end{aligned} \quad (12)$$

where k_v and k_ω are gains, $u_{dv}(t)$ and $u_{d\omega}(t)$ are control signals of the dynamic controller.

4.2. Simulation Tests of the Control System Considering Parameters Identified Offline

Simulation of the tracking motion of the TURTLEBOT 2 robot (Figure 6) was developed in MATLAB/Simulink environment, using the identified parameters of the robot model. The control system is composed of a:

- desired trajectory generator module that determines the position x_d, y_d and orientation θ_d ,
- a generator of the desired linear velocity v_d and angular velocity ω_d ,
- a kinematic controller described by Equation (10),
- a dynamic controller described by Equation (12),
- and a mathematical model of the robot described by Equations (1) and (3).

In the simulation, the average values of the model parameters shown in Table 1 were used. A comparison of the desired trajectory and the simulation-obtained trajectory, using the parameters identified by the Levenberg-Marguardt method is shown in Figure 7. The position and orientation errors are shown in Figure 8a, while the velocity errors are shown in Figure 8b.

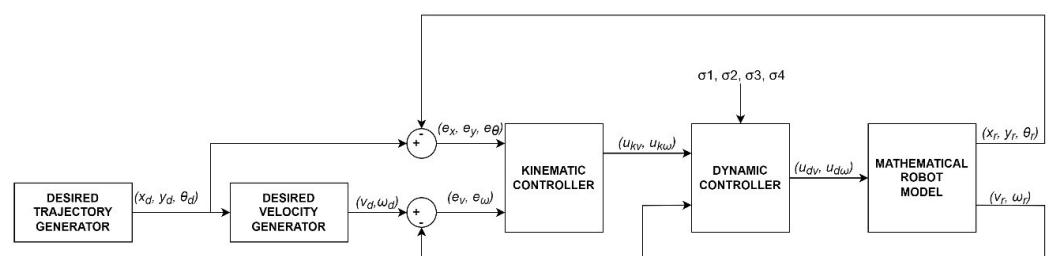


Figure 6. Block diagram of the tracking motion simulation of the TURTLEBOT 2 robot.

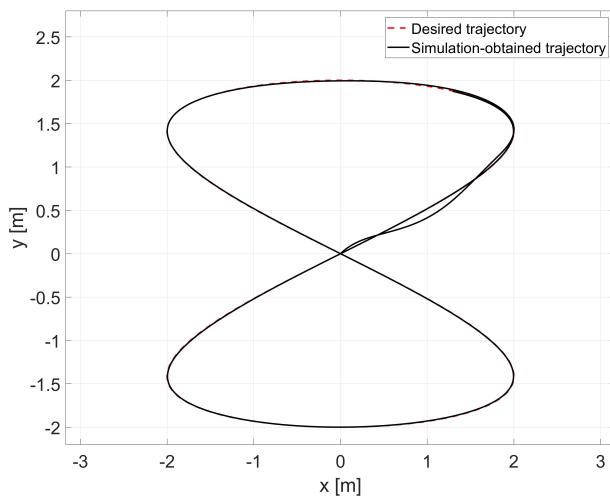


Figure 7. Comparison of the desired trajectory and that obtained by simulation for the Levenberg-Marguardt method.

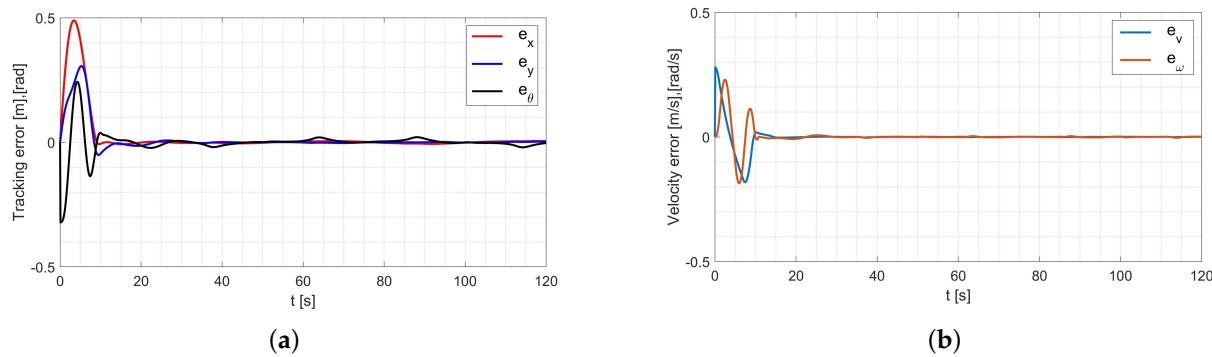


Figure 8. Errors for the Levenberg-Marguardt method: (a) reference trajectory tracking, (b) velocity.

Based on the analysis of the simulation-determined errors for the Levenberg-Marguardt method (Figure 8), it can be concluded that the controller, developed based on the inverse dynamics with the parameters identified by the Levenberg-Marguardt method, realizes control in tracking motion with position, orientation and velocity errors no greater than:

- in the x-axis direction: $e_x = 0.007 \text{ [m]}$,
- in the y-axis direction: $e_y = 0.007 \text{ [m]}$,
- orientation: $e_\theta = 0.021 \text{ [rad]}$,
- linear velocity $e_v = 0.001 \left[\frac{\text{m}}{\text{s}} \right]$,
- angular velocity $e_\omega = 0.007 \left[\frac{\text{rad}}{\text{s}} \right]$.

The regulation time for the case of an 8-shape trajectory tracking task is equal to $t_r \approx 8.2 \text{ [s]}$.

4.3. Simulation Studies of the Control System with Online Identification

Online parameters identification of the robot dynamics model was carried out based on the tracking motion simulation scheme shown in Figure 9.

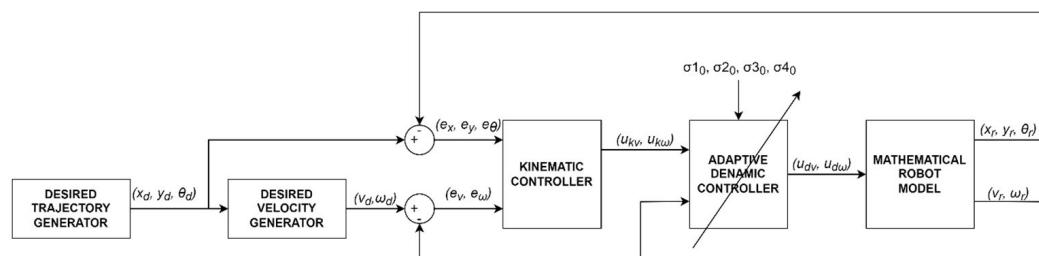


Figure 9. Block diagram of the TURTLEBOT 2 robot’s tracking motion simulation.

In this case, an adaptive dynamic controller using the RLS method described by Equation (6) was used. The simulation was first carried out for parameters with random values equal to $\sigma_1 = 0.52$, $\sigma_2 = 0.23$, $\sigma_3 = 0.49$, $\sigma_4 = 0.62$. A comparison of the desired and resulting trajectories is shown in Figure 10. The position and orientation errors are shown in Figure 11a, while the velocity errors are shown in Figure 11b.

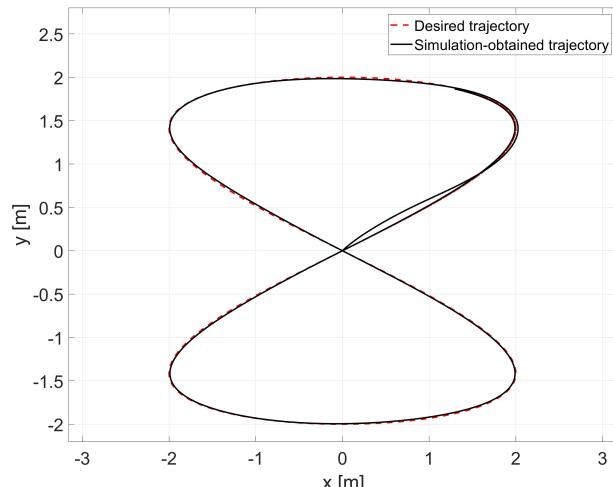


Figure 10. Comparison of reference and actual trajectories in a simulation test for the RLS method with random initial parameters.

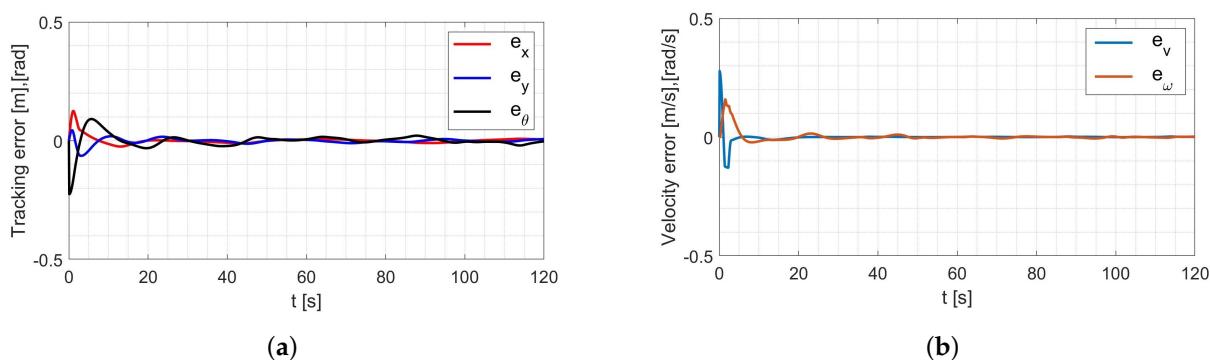


Figure 11. Errors for the RLS method with random initial parameters: (a) tracking the reference trajectory (b) velocity.

Based on the analysis of the simulation-determined errors for the RLS method with random initial parameters (Figure 11), it can be concluded that the adaptive controller, realizes control in tracking motion with position, orientation and velocity errors no greater than:

- in the x-axis direction: $e_x = 0.009$ [m],
- in the y-axis direction: $e_y = 0.016$ [m],

- orientation: $e_\theta = 0.022 \text{ [rad]}$,
- linear velocity $e_v = 0.001 \left[\frac{\text{m}}{\text{s}} \right]$,
- angular velocity $e_\omega = 0.015 \left[\frac{\text{rad}}{\text{s}} \right]$.

The regulation time for the case of an 8-shape trajectory tracking task is equal to $t_r \approx 7.8 \text{ [s]}$.

Next, the random initial parameters were replaced with average values determined using the Levenberg-Marguardt method. A comparison of the desired trajectory and obtained trajectories from this simulation is shown in Figure 12. The position and orientation errors are shown in Figure 13a, while the velocity errors are shown in Figure 13b.

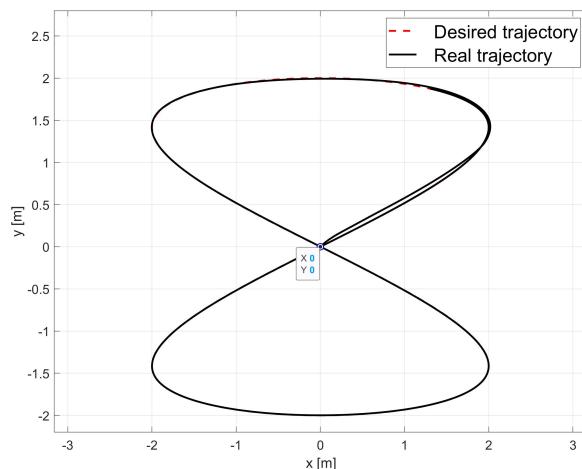


Figure 12. Comparison of the reference trajectory and the one obtained by simulation for the combination of RLS and Levenberg-Marguardt methods.

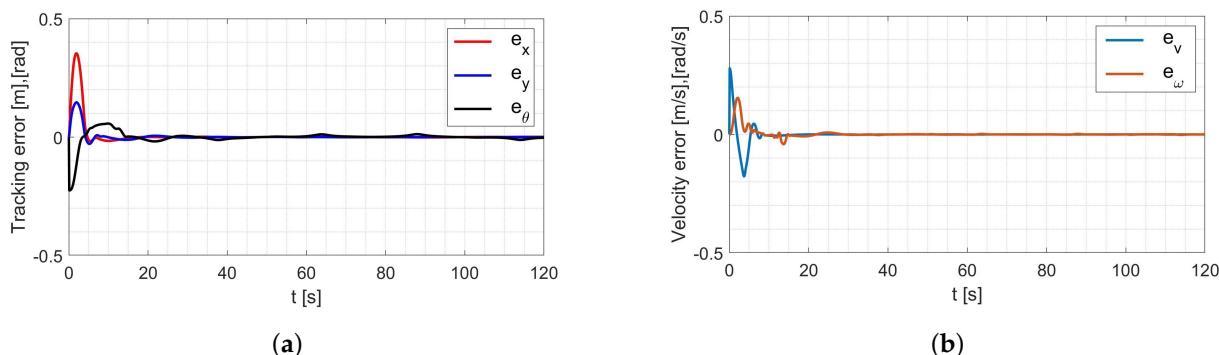


Figure 13. Errors for the RLS method with initial parameters determined by the Levenberg-Marguardt method: (a) tracking the reference trajectory (b) velocity.

Based on the analysis of the simulation-determined errors for the RLS method with the initial parameters determined by the Levenberg-Marguardt method (Figure 13), it can be concluded that the adaptive controller, realizes control in tracking motion with position, orientation and velocity errors no greater than:

- in the x-axis direction: $e_x = 0.005 \text{ [m]}$,
- in the y-axis direction: $e_y = 0.002 \text{ [m]}$,
- orientation: $e_\theta = 0.001 \text{ [rad]}$,
- linear velocity $e_v = 0.001 \left[\frac{\text{m}}{\text{s}} \right]$,
- angular velocity $e_\omega = 0.017 \left[\frac{\text{rad}}{\text{s}} \right]$.

The regulation time for the case of an 8-shape trajectory tracking task is equal to $t_r \approx 4.8 \text{ [s]}$.

4.4. Laboratory Tests of the Control System with Parameters Identified Offline

Verification of the identified parameters was carried out by laboratory tests of the tracking motion using a real robot TURTLEBOT 2. The test used a control system, including a kinematics (10) and dynamics (12) controller, implemented using the basic elements of the Simulink package. The Simulink/ROS library and two tools (Publisher—to publish a specific type of message in a declared communication node, and Subscriber—to receive messages from the robot system transmitted in a declared communication node) were used to communicate with the robot's operating system. Using Publisher, control signals in the form of linear and angular velocity are sent to the robot system. Using Subscriber, position and orientation are received from the robot system. A block diagram of trajectory tracking by the real TURTLEBOT 2 robot, taking into account parameters identified by offline methods, is shown in Figure 14.

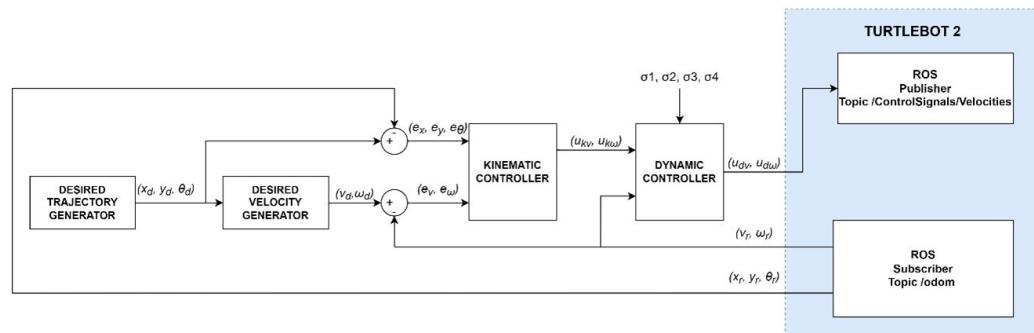


Figure 14. Block diagram of the real TURTLEBOT 2 robot's tracking motion control system.

The laboratory test used the average values of the model parameters shown in Table 1. A comparison of the reference and real trajectory for TURTLEBOT 2 robot with the parameters identified by the Levenberg-Marguardt method is shown in Figure 15, position and orientation errors are shown in Figure 16a, and velocity errors are shown in Figure 16b.

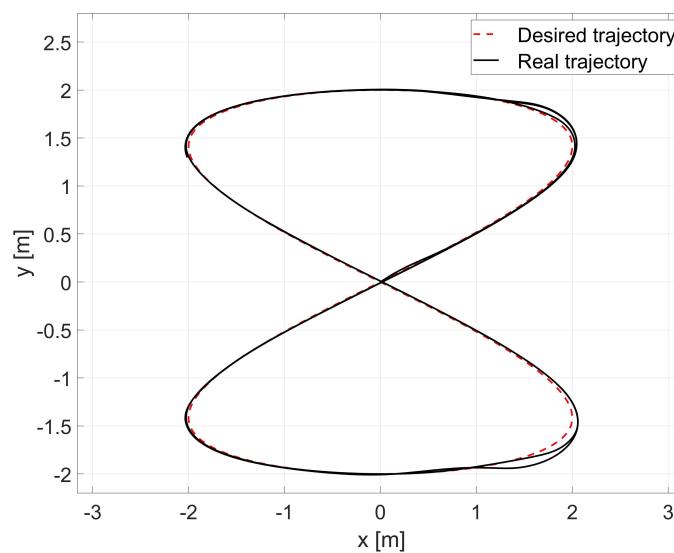


Figure 15. Comparison of reference and real trajectories in a laboratory test.

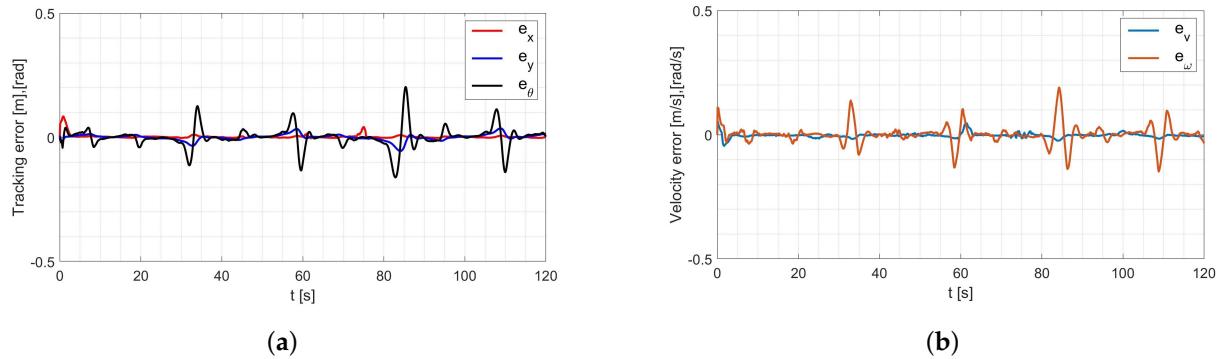


Figure 16. Errors for the Levenberg-Marguardt method: (a) tracking the reference trajectory (b) velocity.

Based on the analysis of the errors determined in laboratory tests (Figure 16), it can be concluded that the controller, developed based on the inverse dynamics with the parameters identified by the Levenberg-Marguardt method, realizes control in tracking motion with position, orientation and velocity errors no greater than:

- in the x-axis direction: $e_x = 0.033$ [m],
- in the y-axis direction: $e_y = 0.046$ [m],
- orientation: $e_\theta = 0.2$ [rad],
- linear velocity $e_v = 0.023$ [$\frac{\text{m}}{\text{s}}$],
- angular velocity $e_\omega = 0.18$ [$\frac{\text{rad}}{\text{s}}$].

The regulation time for the case of 8-shape trajectory tracking task is equal to $t_r \approx 2.9$ [s].

4.5. Laboratory Tests of Control System with Online Identification

The final step was to perform laboratory tests to identify the parameters of the robot's dynamics model using online methods. A block diagram of trajectory tracking by the real TURTLEBOT 2 robot including the adaptive controller is shown in Figure 17.

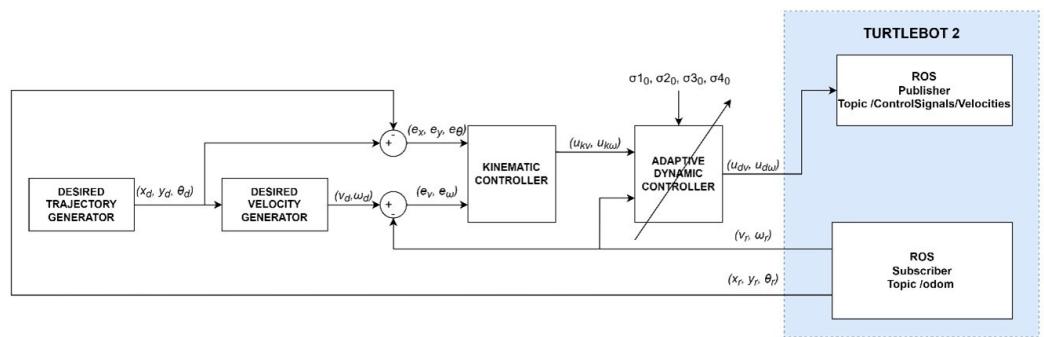


Figure 17. Block diagram of a real robot tracking motion with an adaptive controller.

The simulation was first carried out for random starting parameters with values of $\sigma_1 = 0.52$, $\sigma_2 = 0.23$, $\sigma_3 = 0.49$, $\sigma_4 = 0.62$. A comparison of the desired and resulting trajectories is shown in Figure 18. Position and orientation errors are shown in Figure 19a, while speed errors are shown in Figure 19b. The distribution of parameters identified during movement is shown in Figure 20.

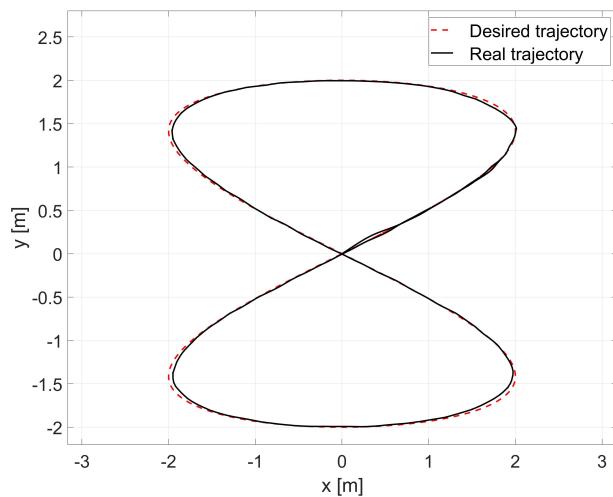


Figure 18. Comparison of reference and real trajectories in a laboratory test of RLS identification with random values of initial parameters.

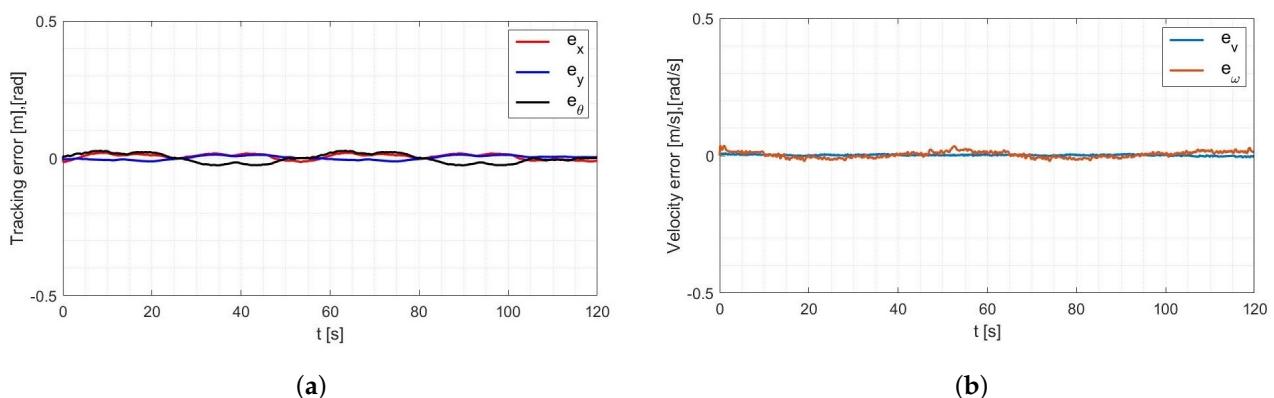


Figure 19. The errors of: (a) tracking the reference trajectory (b) speed.

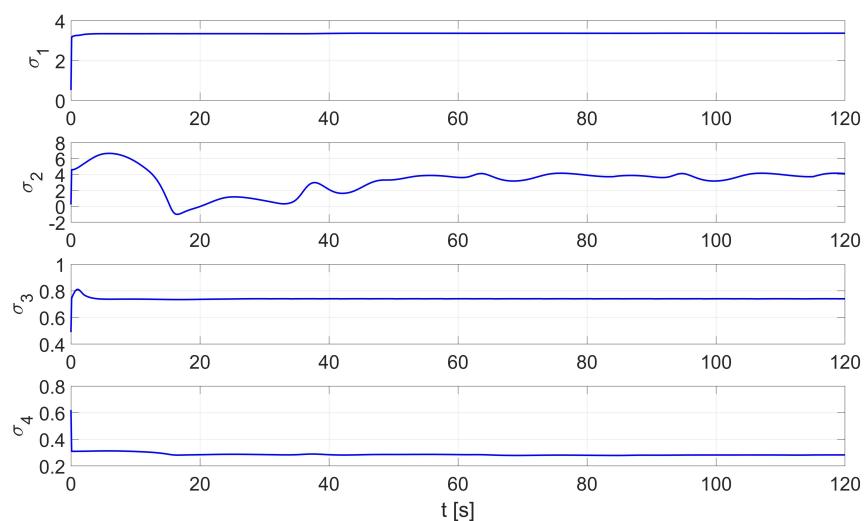


Figure 20. Distribution of parameters identified online by the RLS method with random values of initial parameters in the laboratory test.

Based on the analysis of the errors determined during the laboratory test for the RLS method with random initial parameters (Figure 19), it can be concluded that the adaptive controller, realizes control in tracking motion with position, orientation and velocity errors no greater than:

- in the x-axis direction: $e_x = 0.021 \text{ [m]}$,
- in the y-axis direction: $e_y = 0.013 \text{ [m]}$,
- orientation: $e_\theta = 0.025 \text{ [rad]}$,
- linear velocity $e_v = 0.011 \text{ [\frac{m}{s}]}$,
- angular velocity $e_\omega = 0.033 \text{ [\frac{rad}{s}]}$.

The regulation time for the case of an 8-shape trajectory tracking task is equal to $t_r \approx 7.5 \text{ [s]}$.

The random initial parameters were then replaced with average values determined using the Levenberg-Marguardt method. A comparison of the desired and real trajectory is shown in Figure 21. The position and orientation errors for the combination of the RLS and Levenberg-Marguardt methods are shown in Figure 22a, while the velocity errors are shown in Figure 22b. The distribution of parameters identified during movement is shown in Figure 23.

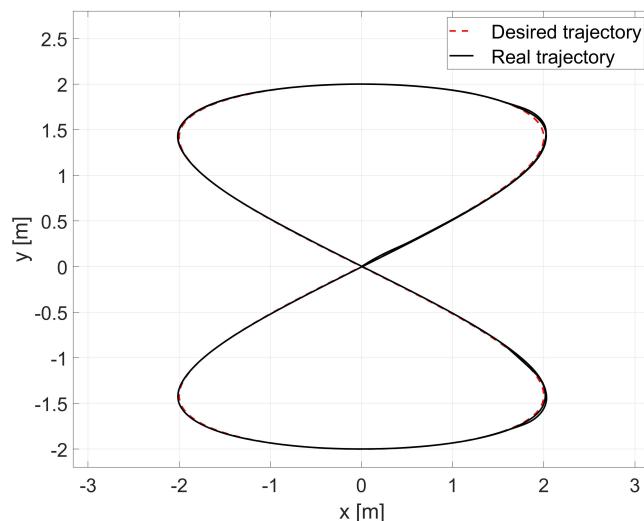


Figure 21. Comparison of the reference and real trajectories in the laboratory test of identification by the RLS method with the values of the initial parameters determined by the Levenberg-Marguardt method.

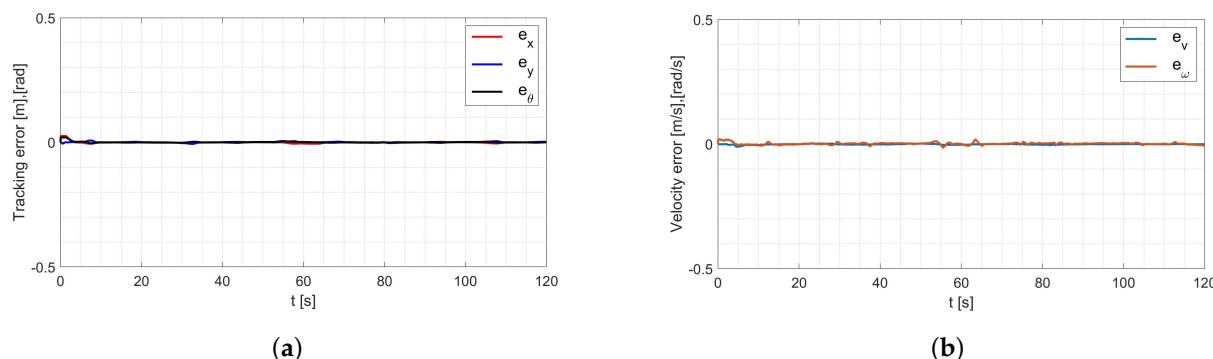


Figure 22. Errors for the RLS method with initial parameters determined by the Levenberg-Marguardt method: (a) tracking of the reference trajectory, (b) velocity.

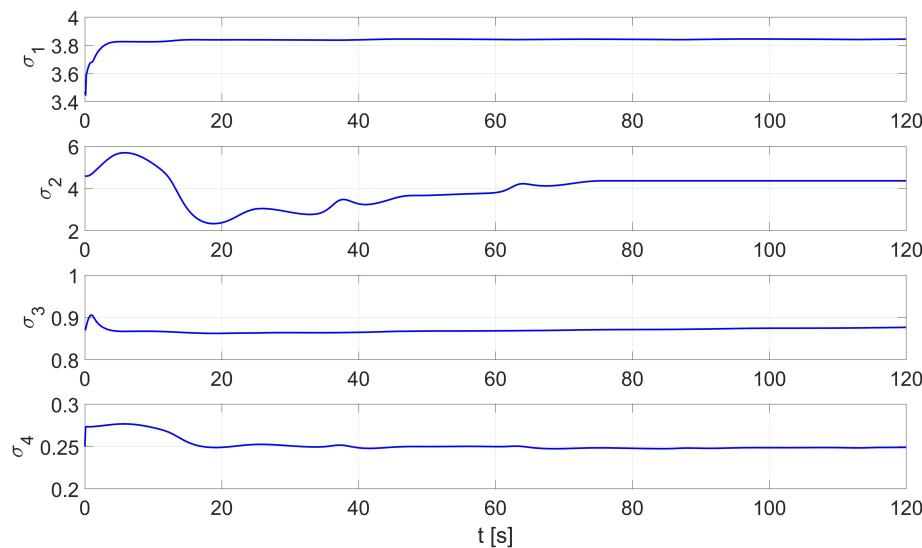


Figure 23. Distribution of parameters identified online by the RLS method with random values of initial parameters in the laboratory test.

Based on the analysis of the errors determined during the laboratory test for the RLS method with the initial parameters determined by the Levenberg-Marguardt method (Figure 22), it can be concluded that the adaptive controller, realizes control in tracking motion with position, orientation and velocity errors no greater than:

- in the x-axis direction: $e_x = 0.009 \text{ [m]}$,
- in the y-axis direction: $e_y = 0.008 \text{ [m]}$,
- orientation: $e_\theta = 0.012 \text{ [rad]}$,
- linear velocity $e_v = 0.001 \left[\frac{\text{m}}{\text{s}} \right]$,
- angular velocity $e_\omega = 0.033 \left[\frac{\text{rad}}{\text{s}} \right]$.

The regulation time for the case of an 8-shape trajectory tracking task is equal to $t_r \approx 1.6 \text{ [s]}$.

To compare the selected identification methods, all determined errors discussed in this article, are summarized in the Table 4 below:

Table 4. A summary of the errors determined in the studies presented.

Errors	Simulation Tests			Laboratory Tests		
	L-M	RLS	RLS + L-M	L-M	RLS	RLS – L-M
$e_x \text{ [m]}$	0.007	0.009	0.005	0.033	0.021	0.009
$e_y \text{ [m]}$	0.007	0.016	0.002	0.046	0.013	0.008
$e_\theta \text{ [rad]}$	0.021	0.022	0.001	0.2	0.025	0.012
$e_v \left[\frac{\text{m}}{\text{s}} \right]$	0.001	0.001	0.001	0.023	0.011	0.001
$e_\omega \left[\frac{\text{rad}}{\text{s}} \right]$	0.007	0.015	0.017	0.18	0.033	0.033
$t_r \text{ [s]}$	8.2	7.8	4.8	2.9	7.5	1.6

5. Conclusions

The article presents the process of identifying the dynamics model parameters of a mobile robot on the example of the laboratory robot TURTLEBOT 2. The identification was carried out by an offline method using a preregistered trajectory and an online method—identifying the parameters during movement. The correctness of the identification process and the values of the identified parameters were checked first by simulating the robot's tracking motion along a desired trajectory, and then by conducting laboratory tests using a

real robot. By making a comparative analysis of the desired and simulation-determined trajectories, and during laboratory testing, it has been concluded that:

- all the values of the parameters identified by the selected methods are correct, and their implementation in the control system gives satisfactory results of tracking the desired trajectory;
- parameters identified by the Levenberg-Marguardt method allow trajectory tracking with a tracking error of about 30–40 [mm] and a regulation time of about 2.9 [s];
- the use of online identification by the RLS method with random initial parameters made it possible to achieve a tracking accuracy of the set trajectory of about 10–20 [mm] and a control time of 7.5 [s]; the use of adaptive technology in the control system made it possible to improve the tracking accuracy compared to the control system with constant values of the parameters of the robot model;
- The best tracking accuracy of the desired trajectory (a tracking error of about 2–5 [mm] and a regulation time of 1.6 [s]) was achieved using the RLS method with the initial parameters determined by the Levenberg-Marguardt method.

The conducted research has shown an improvement in the accuracy of trajectory tracking of AGV robots, by using a novel combination of offline and online methods proposed by authors, for the identification of mathematical model parameters. Since the identification process and control system have been developed for a differentially driven two-wheeled robot, the presented solution can be implemented to other AGV robots, with a drive system based on the one presented in the article. Since the TURTLEBOT 2 robot is not an accurate industrial unit (it is only a low-cost laboratory robot), it introduces errors in the form of vibrations and inaccuracies in measuring the rotation angle in the z-axis and y-axis (Figure 2a). Implementation of the methodology presented in this paper on a professional and more rigid robot could achieve even better accuracy in tracking the desired trajectory. The development trend toward the use of AGV robots in internal transportation and the cooperation of industrial and mobile robots in manufacturing processes in the age of Industry 4.0 confirms the advisability of the presented research and the need to develop methods to improve the accuracy of tracking the desired trajectory.

Author Contributions: Conceptualization, M.S., L.B. and J.P.; methodology, M.S., J.P. and W.K.; software, M.S., P.P. and S.B.; validation, J.P. and W.K.; formal analysis, L.B.; investigation, M.S. and J.P.; resources, M.S., S.B. and W.K.; data curation, M.S. and P.P.; writing—original draft preparation, M.S., S.B. and J.P.; writing—review and editing, M.S. and S.B.; visualization, M.S.; supervision, L.B. and J.P.; project administration, J.P. and W.K.; funding acquisition, W.K. and M.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was cofinanced by the Military University of Technology under research project UGB 776/2022.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

RLS	Recursive Least Squares method
L-M	Levenberg-Marguardt method
x_d, y_d, θ_d	desired robot position and orientation
x, y, θ	desired robot position and orientation
v_d, ω_d	desired robot velocities (linear and angular)
v, ω	real robot velocities (linear and angular)
v_x, v_y	forward and lateral robot velocities
F_{rx}, F_{ry}	longitudinal and lateral force of right wheel
F_{lx}, F_{ly}	longitudinal and lateral force of left wheel
m	mass of the robot
I_z	moment of inertia of the robot with respect to the axis passing through the point G
G	robot center of mass and center of rotation

$\sigma_1, \dots, \sigma_4$	parameters of the robot dynamic model
e_x, e_y, e_θ	position and orientation tracking errors
e_v, e_ω	velocity errors
$u_k v, u_k \omega$	control signals from kinematic controller
k_1, k_2, k_3	gains of the kinematic controllers
ϵ	oscillation damping coefficient
ω_n	characteristic frequency
b	additional control coefficient
$u_d v, u_d \omega$	control signals from dynamic controller
k_v, k_ω	gains of the dynamic controller
t_r	regulation time

Appendix A

In the case shown, the robot's rotation point G is located on the axis connecting the drive wheels so the forces acting in the y -axis direction do not cause any torque. Therefore, the system of Equation (2) can be transformed to the form:

$$\begin{aligned}\dot{v}_x &= \frac{F_{rlx} + F_{rrx}}{m} + v_y \omega \\ \dot{\omega} &= \frac{d(F_{rrx} - F_{rlx})}{2I_z}\end{aligned}\quad (\text{A1})$$

Longitudinal velocity v_x , lateral velocity v_y and angular velocity ω without taking slip into account are equal:

$$\begin{aligned}v_x &= \frac{1}{2}r(\omega_r + \omega_l) \\ v_x &= 0 \\ \omega &= \frac{1}{2d}r(\omega_r - \omega_l)\end{aligned}\quad (\text{A2})$$

where r —radius of driving wheels, ω_r, ω_l —angular velocity of wheels (right and left).

The angular velocity of the wheels can be determined by transforming the expressions for the right motor torques τ_r and left motor torques τ_l [57]:

$$\begin{aligned}\tau_r &= \frac{k_a(V_r - k_b\omega_r)}{R_a} \\ \tau_l &= \frac{k_a(V_l - k_b\omega_l)}{R_a}\end{aligned}\quad (\text{A3})$$

where V_r, V_l —the supply voltages of the motors (right and left), k_b —the electrical constants of motors, k_a —the mechanical constant of the engines, R_a —resistance of motor. The driving wheel dynamics equations [60] take the form:

$$\begin{aligned}I_e \dot{\omega}_r + B_e \omega_r &= \tau_r - F_{rrx}r \\ I_e \dot{\omega}_l + B_e \omega_l &= \tau_l - F_{rlx}r\end{aligned}\quad (\text{A4})$$

where I_e —moment of inertia of the wheel with respect to the axis of rotation, B_e —coefficient of viscous friction reduced to the motor shaft. The equations of the PD controller implemented to regulate the supply voltage of motors [60] take the form:

$$\begin{aligned}u_v &= k_{PT}(v_d - v_x) - k_{DT}\dot{v}_x \\ u_\omega &= k_{PR}(\omega_d - \omega) - k_{DR}\dot{\omega}\end{aligned}\quad (\text{A5})$$

where $u_v = \frac{V_r + V_l}{2}$, $u_\omega = \frac{V_r - V_l}{2}$, $k_{PT}, k_{DT}, k_{PR}, k_{DR}$ —gains of PD controllers, v_d —desired linear velocity, ω_d —desired angular velocity.

By substituting Equation (A3) into Equation (A4) and transforming to form for determining longitudinal driving forces, a system of equations was obtained:

$$\begin{aligned}F_{rrx} &= \frac{1}{r} \left(\frac{k_a}{R_a} (V_r - k_b \omega_r) - I_e \dot{\omega}_r + B_e \omega_r \right) \\ F_{rlx} &= \frac{1}{r} \left(\frac{k_a}{R_a} (V_l - k_b \omega_l) - I_e \dot{\omega}_l + B_e \omega_l \right)\end{aligned}\quad (\text{A6})$$

Then substituting Equation (A6) into the system of Equation (A1) obtained:

$$\begin{aligned}\dot{v}_x &= \frac{1}{rm} \left(\frac{k_a}{R_a} (V_l + V_r) + (\omega_l + \omega_r) \left(-\frac{k_a}{R_a} + B_e \right) - I_e (\omega_l + \omega_r) \right) + v_y \omega \\ \dot{\omega} &= \frac{d}{I_z^2 r} \left(\frac{k_a}{R_a} (V_r - V_l) + (\omega_r + \omega_l) \left(-\frac{k_a}{R_a} k_b + B_e \right) - I_e (\omega_r + \omega_l) \right)\end{aligned}\quad (\text{A7})$$

Combining the above equations with Equations (A2) and (A5) and performing algebraic transformations, expressions describing the TURTLEBOT 2 robot dynamics model were obtained:

$$\begin{aligned}\dot{v}_x \left(\frac{1}{k_{PT}} \left(\frac{R_a}{k_a} \left(\frac{rm}{2} + \frac{I_e}{r} \right) + k_{DT} \right) \right) &= -v_x \left(1 - \frac{1}{rk_{PT}} \left(1 + B_e \frac{R_a}{k_a} \right) \right) + v_d \\ \dot{\omega}_x \left(\frac{1}{k_{PR}} \left(\frac{R_a}{2k_a} \left(\frac{I_z^2 r}{d} + \frac{dI_e}{r} \right) + k_{DR} \right) \right) &= -\omega \left(\frac{d}{rk_{PR}} \left(1 - \frac{R_a}{k_a} \right) + 1 \right) + \omega_d\end{aligned}\quad (\text{A8})$$

or in shortened form:

$$\begin{aligned}\dot{v}_x \sigma_1 &= -v_x \sigma_3 + v_d \\ \dot{\omega}_x \sigma_2 &= -\omega \sigma_4 + \omega_d\end{aligned}\quad (\text{A9})$$

Appendix B

Multiplying Equation (3) on both sides by $\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}$ the form was obtained:

$$\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} \dot{v}_x \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -\sigma_3 v_x \\ -\sigma_4 \omega \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_d \\ \omega_d \end{bmatrix}\quad (\text{A10})$$

By performing transformations, the above expression can be written in the form:

$$\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} \dot{v}_x \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} \sigma_3 \\ \sigma_4 \end{bmatrix} \begin{bmatrix} v_x & 0 \\ 0 & \omega \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_d \\ \omega_d \end{bmatrix}\quad (\text{A11})$$

and then bring it to parametric form:

$$\begin{bmatrix} v_d \\ \omega_d \end{bmatrix} = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} \dot{v}_x \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} 0 & 0 & v_x & 0 \\ 0 & 0 & 0 & \omega \end{bmatrix} \begin{bmatrix} \sigma_1 & \sigma_2 & \sigma_3 & \sigma_4 \end{bmatrix}^T\quad (\text{A12})$$

or in shortened form:

$$\begin{bmatrix} v_d \\ \omega_d \end{bmatrix} = \begin{bmatrix} \dot{v}_x & 0 & v_x & 0 \\ 0 & \dot{\omega} & 0 & \omega \end{bmatrix} \begin{bmatrix} \sigma_1 & \sigma_2 & \sigma_3 & \sigma_4 \end{bmatrix}^T\quad (\text{A13})$$

$$\begin{bmatrix} v_d \\ \omega_d \end{bmatrix} = A\sigma\quad (\text{A14})$$

Based on the inverse dynamic task [61], the above expression (A12) can be reduced to the form of dynamic controller (12).

References

1. Szrek, J.; Jakubiak, J.; Zimroz, R. A Mobile Robot-Based System for Automatic Inspection of Belt Conveyors in Mining Industry. *Energies* **2022**, *15*, 327. [[CrossRef](#)]
2. Piątek, Z. Mobile Industrial Robots Enters Polish Mobile Robot Market (In Polish). 2018. Available online: <https://przemysl-40.pl/index.php/2018/01/11/mobile-industrial-robots-wchodzi-na-polski-rynek/> (accessed on 5 December 2022).
3. AGV, AIV, AMR, SGV—So What’s the Deal with the Nomenclature? (In Polish). Available online: <http://www.mobot.pl/artykul/5095/agv-aiv-amr-sgv-czyl-o-co-chodzi-z-tym-nazewnictwem/> (accessed on 5 December 2022).
4. Jahn, U.; Heß, D.; Stampa, M.; Sutorma, A.; Röhrig, C.; Schulz, P.; Wolff, C. A Taxonomy for Mobile Robots: Types, Applications, Capabilities, Implementations, Requirements, and Challenges. *Robotics* **2020**, *9*, 109. [[CrossRef](#)]
5. SFS-EN ISO 3691-4:2020; Industrial Trucks. Safety Requirements and Verification. Part 4: Driverless Industrial Trucks and Their Systems. SFS: Singapore, 2020.

6. Vasconcelos, J.V.R.; Brandão, A.S.; Sarcinelli-Filho, M. Real-Time Path Planning for Strategic Missions. *Appl. Sci.* **2020**, *10*, 7773. [[CrossRef](#)]
7. Zhou, Z.; Zhu, P.; Zeng, Z.; Xiao, J.; Lu, H.; Zhou, Z. Robot navigation in a crowd by integrating deep reinforcement learning and online planning. *Appl. Intell.* **2022**, *52*, 15600–15616. [[CrossRef](#)]
8. Wróblewski, P. Analysis of Torque Waveforms in Two-Cylinder Engines for Ultralight Aircraft Propulsion Operating on 0W-8 and 0W-16 Oils at High Thermal Loads Using the Diamond-like Carbon Composite Coating. *SAE Int. J. Engines* **2021**, *15*, 2022. [[CrossRef](#)]
9. Chen, Y.; Xie, B.; Long, J.; Kuang, Y.; Chen, X.; Hou, M.; Gao, J.; Zhou, S.; Fan, B.; He, Y.; et al. Interfacial Laser-Induced Graphene Enabling High-Performance Liquid-Solid Triboelectric Nanogenerator. *Adv. Mater.* **2021**, *33*, 2104290. [[CrossRef](#)]
10. Siwek, M.; Bessegheur, K.; Baranowski, L. The effects of the swarm configuration and the obstacles placement on control signals transmission delays in decentralized ROS-embedded group of mobile robots. *AIP Conf. Proc.* **2018**, *2029*, 20069. [[CrossRef](#)]
11. Timofeev, A.; Daef, F. Control of a transport robot in extreme weather based on computer vision systems. *J. Phys. Conf. Ser.* **2021**, *2131*, 32027. [[CrossRef](#)]
12. Javed, M.A.; Muram, F.U.; Punnekkat, S.; Hansson, H. Safe and secure platooning of Automated Guided Vehicles in Industry 4.0. *J. Syst. Archit.* **2021**, *121*, 102309. [[CrossRef](#)]
13. Falcó, A.; Hilario, L.; Montés, N.; Mora, M.C.; Nadal, E. A Path Planning Algorithm for a Dynamic Environment Based on Proper Generalized Decomposition. *Mathematics* **2020**, *8*, 2245. [[CrossRef](#)]
14. Sánchez-Ibáñez, J.R.; Pérez-Del-pulgar, C.J.; García-Cerez, A. Path planning for autonomous mobile robots: A review. *Sensors* **2021**, *21*, 7898. [[CrossRef](#)] [[PubMed](#)]
15. Zhang, H.Y.; Lin, W.M.; Chen, A.X. Path planning for the mobile robot: A review. *Symmetry* **2018**, *10*, 450. [[CrossRef](#)]
16. Kaczmarek, W.; Borys, S.; Panasiuk, J.; Siwek, M.; Prusaczyk, P. Experimental Study of the Vibrations of a Roller Shutter Gripper. *Appl. Sci.* **2022**, *12*, 9996. [[CrossRef](#)]
17. Borys, S.; Kaczmarek, W.; Laskowski, D.; Polak, R. Experimental Study of the Vibration of the Spot Welding Gun at a Robotic Station. *Appl. Sci.* **2022**, *12*, 12209. [[CrossRef](#)]
18. Ravankar, A.; Ravankar, A.A.; Kobayashi, Y.; Hoshino, Y.; Peng, C.C. Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges. *Sensors* **2018**, *18*, 3170. [[CrossRef](#)]
19. Wu, M.; Dai, S.L.; Yang, C. Mixed reality enhanced user interactive path planning for omnidirectional mobile robot. *Appl. Sci.* **2020**, *10*, 1135. [[CrossRef](#)]
20. Andreasson, H.; Saarinen, J.; Cirillo, M.; Stoyanov, T.; Lilenthal, A.J. Drive the drive: From discrete motion plans to smooth drivable trajectories. *Robotics* **2014**, *3*, 400–416. [[CrossRef](#)]
21. Siwek, M.; Baranowski, L.; Panasiuk, J.; Kaczmarek, W. Modeling and simulation of movement of dispersed group of mobile robots using Simscape multibody software. *AIP Conf. Proc.* **2019**, *2078*, 020045. [[CrossRef](#)]
22. Baranowski, L.; Siwek, M. Use of 3D Simulation to Design Theoretical and Real Pipe Inspection Mobile Robot Model. *Acta Mech. Autom.* **2018**, *12*, 232–236. [[CrossRef](#)]
23. Noh, S.; Park, J.; Park, J. Autonomous Mobile Robot Navigation in Indoor Environments: Mapping, Localization, and Planning. In Proceedings of the International Conference on Information and Communication Technology Convergence, Jeju, Republic of Korea, 21–23 October 2020; pp. 908–913. [[CrossRef](#)]
24. Bessegheur, K.L.; Trebinski, R.; Kaczmarek, W.; Panasiuk, J. Trajectory tracking control for a nonholonomic mobile robot under ROS. *J. Phys. Conf. Ser.* **2018**, *1016*, 12008. [[CrossRef](#)]
25. Gonçalves, F.; Ribeiro, T.; Ribeiro, A.F.; Lopes, G.; Flores, P. A Recursive Algorithm for the Forward Kinematic Analysis of Robotic Systems Using Euler Angles. *Robotics* **2022**, *11*, 15. [[CrossRef](#)]
26. Groves, K.; Hernandez, E.; West, A.; Wright, T.; Lennox, B. Robotic exploration of an unknown nuclear environment using radiation informed autonomous navigation. *Robotics* **2021**, *10*, 78. [[CrossRef](#)]
27. Wróblewski, P. Investigation of energy losses of the internal combustion engine taking into account the correlation of the hydrophobic and hydrophilic. *Energy* **2023**, *264*, 126002. [[CrossRef](#)]
28. Panasiuk, J.; Siwek, M.; Kaczmarek, W.; Borys, S.; Prusaczyk, P. The concept of using the mobile robot for telemechanical wires installation in pipelines. *AIP Conf. Proc.* **2018**, *2029*, 020054. [[CrossRef](#)]
29. Khosla, P.K.; Kanade, T. Parameter Identification of Robot Dynamics. In Proceedings of the 24th IEEE Conference on Decision and Control, Fort Lauderdale, FL, USA, 11–13 December 1985; pp. 1754–1760. [[CrossRef](#)]
30. Gautier, M.; Janot, A.; Vandanon, P.O. A new closed-loop output error method for parameter identification of robot dynamics. *IEEE Trans. Control Syst. Technol.* **2013**, *21*, 428–444. [[CrossRef](#)]
31. Briot, S.; Krut, S.; Gautier, M. Dynamic Parameter Identification of Overactuated Parallel Robots. *J. Dyn. Syst. Meas. Control. Trans. ASME* **2015**, *137*, 111002. [[CrossRef](#)]
32. Lomakin, A.; Deutscher, J. Identification of dynamic parameters for rigid robots based on polynomial approximation. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, NV, USA, 24 October 2020–24 January 2021; pp. 7271–7278. [[CrossRef](#)]
33. Uddin, N. Adaptive control system design for two-wheeled robot stabilization. In Proceedings of the 12th South East Asian Technical University Consortium, Yogyakarta, Indonesia, 12–13 March 2018; pp. 1–5. [[CrossRef](#)]

34. Al-Jlailaty, H.; Jomaa, H.; Daher, N.; Asmar, D. Balancing a Two-Wheeled Mobile Robot using Adaptive Control. In Proceedings of the 2018 IEEE International Multidisciplinary Conference on Engineering Technology, Beirut, Lebanon, 14–16 November 2018; pp. 1–7. [[CrossRef](#)]
35. Zhai, J.; Song, Z. Adaptive sliding mode trajectory tracking control for wheeled mobile robots. *Int. J. Control.* **2019**, *92*, 2255–2262. [[CrossRef](#)]
36. Chen, Y.; Li, Z.; Kong, H.; Ke, F. Model predictive tracking control of nonholonomic mobile robots with coupled input constraints and unknown dynamics. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3196–3205. [[CrossRef](#)]
37. Al-Mayyahi, A.; Wang, W.; Birch, P. Adaptive neuro-fuzzy technique for autonomous ground vehicle navigation. *Robotics* **2014**, *3*, 349–370. [[CrossRef](#)]
38. Bouzoualegh, S.; Guechi, E.H.; Kelaiaia, R. Model Predictive Control of a Differential-Drive Mobile Robot. *Acta Univ. Sapientiae Electr. Mech. Eng.* **2018**, *10*, 20–41. [[CrossRef](#)]
39. Zhang, K.; Sun, Q.; Shi, Y. Trajectory Tracking Control of Autonomous Ground Vehicles Using Adaptive Learning MPC. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 5554–5564. [[CrossRef](#)]
40. Wu, W.; You, J.; Zhang, Y.; Li, M.; Su, K. Parameter Identification of Spatial-Temporal Varying Processes by a Multi-Robot System in Realistic Diffusion Fields. *Robotica* **2021**, *39*, 842–861. [[CrossRef](#)]
41. Xu, X.; Liu, X.; Zhao, B.; Yang, B. An extensible positioning system for locating mobile robots in unfamiliar environments. *Sensors* **2019**, *19*, 4025. [[CrossRef](#)]
42. Li, L.; Liu, Y.H.; Jiang, T.; Wang, K.; Fang, M. Adaptive Trajectory Tracking of Nonholonomic Mobile Robots Using Vision-Based Position and Velocity Estimation. *IEEE Trans. Cybern.* **2018**, *48*, 571–582. [[CrossRef](#)] [[PubMed](#)]
43. Alatise, M.; Hancke, G. Pose Estimation of a Mobile Robot Based on Fusion of IMU Data and Vision Data Using an Extended Kalman Filter. *Sensors* **2017**, *17*, 2164. [[CrossRef](#)]
44. Liu, Y.; Zhao, C.; Ren, M. An Enhanced Hybrid Visual–Inertial Odometry System for Indoor Mobile Robot. *Sensors* **2022**, *22*, 2930. [[CrossRef](#)]
45. Robot Operating System. Available online: <https://www.roscomponents.com/en/mobile-robots/9-turtlebot-2.html> (accessed on 5 December 2022).
46. Rivera, Z.B.; De Simone, M.C.; Guida, D. Unmanned Ground Vehicle Modelling in Gazebo/ROS-Based Environments. *Machines* **2019**, *7*, 42. [[CrossRef](#)]
47. 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](#)]
48. Chen, C.S.; Lin, C.J.; Lai, C.C.; Lin, S.Y. Velocity Estimation and Cost Map Generation for Dynamic Obstacle Avoidance of ROS Based AMR. *Machines* **2022**, *10*, 501. [[CrossRef](#)]
49. Giergiel, M.; Malka, P. Modelowanie Kinematyki I Dynamiki Mobilnego Minirobota. *Model. Inżynierskie* **2006**, 157–162.
50. Giergiel, M.J.; Hendzel, Z.; Żylski, W. *Modelowanie i Sterowanie Mobilnych Robotów Kołowych*; Wydawnictwo Naukowe PWN: Warszawa, Poland, 2013; p. 257.
51. Malka, P. *Pozycjonowanie i Nadążanie Minirobota Kołowego*; AGH: Kraków, Poland, 2007.
52. Siegwart, R.; Nourbakhsh, I.R.; Scaramuzza, D. *Introduction to Autonomous Mobile Robots*; The MIT Press: Cambridge, MA, USA, 2011; Volume 49, pp. 49–1492. [[CrossRef](#)]
53. Nascimento Martins, F.; Santos Brandão, A. Motion Control and Velocity-Based Dynamic Compensation for Mobile Robots. In *Applications of Mobile Robots*; IntechOpen: London, UK, 2019. [[CrossRef](#)]
54. Das, T.; Kar, I.N. Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots. *IEEE Trans. Control Syst. Technol.* **2006**, *14*, 501–510. [[CrossRef](#)]
55. Hendzel, Z.; Rykała, Ł. Modelling of dynamics of a wheeled mobile robot with mecanum wheels with the use of lagrange equations of the second kind. *Int. J. Appl. Mech. Eng.* **2017**, *22*, 81–99. [[CrossRef](#)]
56. Ahmad Abu Hatab, R.D. Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework. *Adv. Robot. Autom.* **2013**, *2*, 1–7. [[CrossRef](#)]
57. De La Cruz, C.; Carelli, R. Dynamic model based formation control and obstacle avoidance of multi-robot systems. *Robotica* **2008**, *26*, 345–356. [[CrossRef](#)]
58. Martins, F.N.; Sarcinelli-Filho, M.; Carelli, R. A Velocity-Based Dynamic Model and Its Properties for Differential Drive Mobile Robots. *J. Intell. Robot. Syst. Theory Appl.* **2017**, *85*, 277–292. [[CrossRef](#)]
59. Zhang, Y.; Hong, D.; Chung, J.H.; Velinsky, S.A. Dynamic model based robust tracking control of a differentially steered wheeled mobile robot. In Proceedings of the American Control Conference, Philadelphia, PA, USA, 26–28 June 1998; Volume 2, pp. 850–855. [[CrossRef](#)]
60. Gratton, S.; Lawless, A.S.; Nichols, N.K. Approximate Gauss-newton methods for nonlinear least squares problems. *SIAM J. Optim.* **2007**, *18*, 106–132. [[CrossRef](#)]
61. Moré, J.J. The Levenberg–Marquardt algorithm: Implementation and theory. In *Numerical Analysis*; Springer: Berlin/Heidelberg, Germany, 1978; pp. 105–116. [[CrossRef](#)]

-
- 62. Palacín, J.; Rubies, E.; Clotet, E. Systematic Odometry Error Evaluation and Correction in a Human-Sized Three-Wheeled Omnidirectional Mobile Robot Using Flower-Shaped Calibration Trajectories. *Appl. Sci.* **2022**, *12*, 2606. [[CrossRef](#)]
 - 63. Klančar, G.; Matko, D.; Blažič, S. Mobile robot control on a reference path. In Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control, Limassol, Cyprus, 27–29 June 2005; Volume 2005, pp. 1343–1348. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.