PROBLEMY MECHATRONIKI
UZBROJENIE, LOTNICTWO, INŻYNIERIA BEZPIECZEŃSTWA

ISSN 2081-5891
E-ISSN 2720-5266

13, 4 (50), 2022, 67-84

PROBLEMS OF MECHATRONICS
ARMAMENT, AVIATION, SAFETY ENGINEERING

Trajectory Tracking Control of a Mobile Robot with the ROS System

Michał SIWEK*, Jarosław PANASIUK, Leszek BARANOWSKI, Wojciech KACZMAREK, Szymon BORYS

Military University of Technology,
Faculty of Mechatronics, Armament and Aerospace,
2 Sylwestra Kaliskiego Str., 00-908 Warsaw, Poland
**Corresponding author's e-mail address and ORCID*:
michal.siwek@wat.edu.pl; https://orcid.org/0000-0003-2818-9725

Received: August 15, 2022 / Revised: September 1, 2022 / Accepted: December 6, 2022/ Published: December 30, 2022.

DOI 10.5604/01.3001.0016.1458

Abstract. This article provides a simulation and laboratory study of a control system for a two-wheeled differential-drive mobile robot with ROS system. The authors propose an approach to designing a control system based on a parametric model of the robot's dynamics. The values of unknown parameters of the dynamic model have been determined by means of a Levenberg-Marguardt identification method. By comparing the desired trajectories with those obtained from simulation and laboratory tests, and based on errors analysis, the correctness of the model parameter identification process and the control system operation was then determined.

Keywords: mobile robot, ROS, parametric dynamic model, identification, trajectory tracking

1. INTRODUCTION

When choosing a robot to perform a given task, we usually look at its accuracy, functionality and driving capabilities. Parameters such as functionality and driving ability are characteristics of a given robot and do not change over time. Given the complexity of mobile robot design, positioning accuracy (a measure of how close a robot can get to a designated point in space) is a parameter that is most influenced by several varied factors. Positioning accuracy is not a fixed characteristic, it depends on the nature of the work being performed by the robot, the length and intensity of the operation, and the way in which it is controlled. There are works [1, 2] in the literature concerning research into the influence of external factors on the positioning accuracy of robots. The sources of errors are divided into four groups: kinematic, dynamic, control and design. For each group, the factors that have the greatest impact on positioning accuracy have been identified (Table 1).

The impact of some of these factors on the positioning accuracy of a mobile robot can be reduced at the design stage of the robot and its control system. These are mainly dynamic or kinematic related factors, where the designer can perform simulations at the initial design stage and then, based on these, verify the correct selection of the motors or robot dimensions [17]. Also, for design-related factors, their importance in positioning accuracy can be reduced based on the designer's experience, engineering knowledge and accuracy during assembly. However, there are a number of parameters that change over the robot's lifetime and with any change in purpose. The working environment has a particular impact on any change in operating parameters, especially if it is harsh, such as dusty rooms, industrial and sewage pipelines, or mines [18]. To this end, when designing a control algorithm, it is necessary to take into account the possibility of changing the model parameters based on their identification by numerical methods.

Table 1. Factors affecting the accuracy of trajectory tracking by a mobile robot

Kinematics	**Dynamics**
- running gear - complexity - operating space	- inertia - friction - drives
Control	**Design**
- sensor resolution - speed of the algorithm - correct operation of the algorithm	- operating time - precision - workload

2. MODELLING THE TURTLEBOT 2 ROBOT

The studies presented in this paper included the laboratory TURTLEBOT 2 robot (Figure 1) [3], which was a low-budget exercise kit, for the construction and testing of wheeled mobile robot control systems.

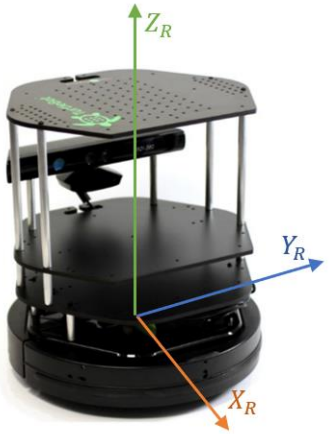

Fig. 1. Isometric view of the TURTLEBOT 2 robot with the base coordinate system.

The platform was equipped with a two-wheeled differential drive and basic sensors, such as a laser scanner, a 640 × 480 resolution RGB vision camera (used for image processing, colour and texture recognition), an IR camera (depth information processing, distance measurement 0.4 - 6.5 m), four directional microphones and an accelerometer. The advantage of this robot is its open control system ROS (*Robot Operating System*), which enables the platform to be quickly started and integrated with sensors [19].

2.1. Robot kinematics and dynamics model

To determine the kinematics model of the TURTLEBOT 2, the diagrams shown in Figure 2 were used.

The coordinate system, shown in Figure 2 $O_0X_0Y_0$, was a global, fixed reference system. Its P and C points were respectively: point P – centre of the arc for planar motion, point C – target point of the robot's progressive movement, with the angle β being the rotation angle of the robot around the point P. The position of the robot in the coordinate system $O_0X_0Y_0$ determined the coordinates x, y while the orientation was determined by the angle Θ.

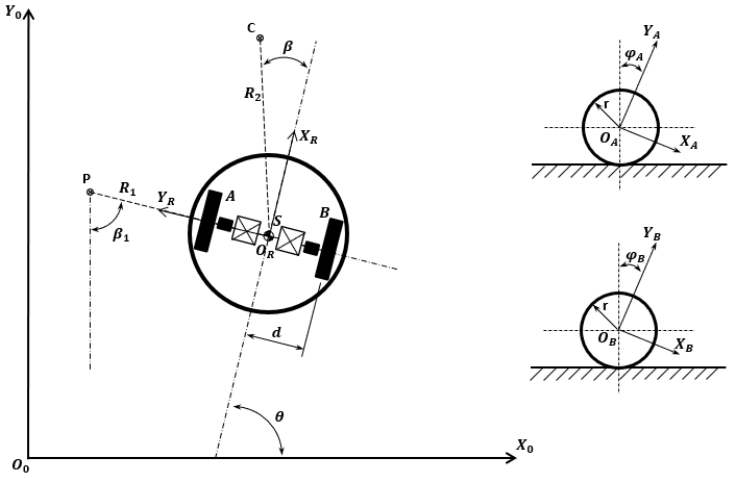

Fig. 2. Kinematic structure of the TURTLEBOT 2

The coordinate system $O_RX_RY_R$ was a system with an origin at the geometric centre of the robot and was linked to the robot frame. The coordinate systems $O_AX_AY_A$ and $O_BX_BY_B$ were connected with the drive wheels: *A* and *B* respectively, with an origin in the wheel axes. The angles φ_A and φ_B were the angles of self-rotation of wheels *A* and *B*.

The kinematics of the TURTLEBOT 2 mobile robot was described by equation (1) in a form in which the angular velocities of the drive wheels were the control signals [4-7]:

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}cos\theta & -sin\theta & 0\\ sin\theta & cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}\frac{r}{2}\begin{bmatrix}\dot{\varphi}_A + \dot{\varphi}_B\\ 0\\ \dfrac{\dot{\varphi}_A + \dot{\varphi}_B}{d}\end{bmatrix} \tag{1}$$

By converting the above equation into a form where the linear υ and angular ω velocities of the robot were control signals, described by the relations:

$$\upsilon = \frac{r(\dot{\varphi}_A + \dot{\varphi}_B)}{2}, \quad \omega = \frac{r(\dot{\varphi}_A - \dot{\varphi}_B)}{2d} \tag{2}$$

the robot kinematics model described by the equation system were obtained:

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}cos\theta & 0\\ sin\theta & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\upsilon\\ \omega\end{bmatrix} \tag{3}$$

When analysing the literature for control systems, it can be noted that most of the proposed controllers performing a trajectory tracking task consider only the robot's kinematics.

This was an acceptable solution, but for tasks requiring high positioning accuracy and high-speed movement, it was necessary to include the robot's dynamics in the control. In the literature, dynamic models of robots that consider torques or motor voltages as control signals were popular [13-16]. A much better solution was to use a dynamic model that takes the linear and angular speed of the robot as control signals [8, 9]. This was how commercially available mobile robots were usually controlled. Figure 3 shows a diagram of the TURTLEBOT 2 robot used to determine the dynamics model.

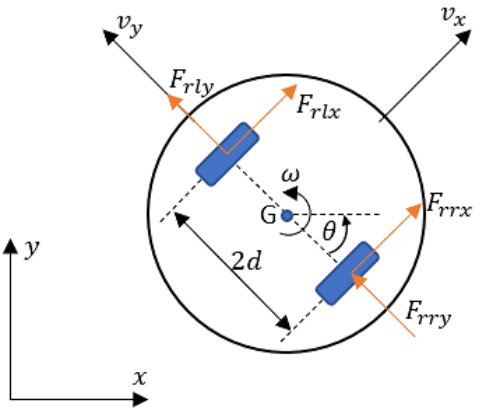

Fig. 3. TURTLEBOT 2 robot diagram for dynamic model development

The *G* point shown in the TURTLEBOT 2 diagram (Fig. 3) was the robot's centre of mass and centre of rotation, and at the same time the tracking point of the desired trajectory. The robot movement velocities were indicated as: υ_x – forward velocity, υ_y – side velocity, and ω – angular velocity. Figure 3 indicates the robot's drive forces 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 on the basis of the forces and moments balance in the system related to the robot, following [10]:

$$\begin{aligned} \sum F_x &= F_{rlx} + F_{rrx} = m\left(\dot{v}_x - v_y\omega\right) \\ \sum F_y &= F_{rly} + F_{rry} = m\left(\dot{v}_y - v_x\omega\right) \\ \sum M_z &= I_z\omega = d(F_{rrx} - F_{rlx}) \end{aligned} \tag{4}$$

where:

m – mass of the robot,

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

The conversion of the equation system (4) was the dynamic model of the TURTLEBOT 2 (see Appendix 1):

$$\begin{bmatrix} \dot{v}_x \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -\frac{\delta_3}{\delta_1} v_x \\ -\frac{\delta_4}{\delta_2} \omega \end{bmatrix} + \begin{bmatrix} \frac{1}{\delta_1} & 0 \\ 0 & \frac{1}{\delta_2} \end{bmatrix} \begin{bmatrix} v_d \\ \omega_d \end{bmatrix} \tag{5}$$

The coefficients of the model $\delta_1 \ldots \delta_4$ included parameters that were difficult for the laboratory robot user to obtain. As these factors were the physical functions parameters of the robot related to drives, transmissions, internal controls and friction, and were usually not even included in the user manuals or equipment technical sheets, their identification had to be carried out.

2.2. Identification of the TURTLEBOT 2 robot model parameters

Before starting to identify the model parameters, it should be checked whether any of the model parameters can be written as a linear combination of two others. In this case, it would be possible to describe the robot dynamics model with fewer parameters. Analysing the developed mathematical model of the robot, the linear independence between the parameters $\delta_1 \ldots \delta_4$ was not clearly visible because some physical parameters affect more than one parameter δ_n, so a detailed analysis was necessary, which was carried out in the work [9] to demonstrate that the model parameters from δ_1 to δ_4 were independent, so they could not be written as a linear combination. Ultimately, it was necessary to identify each of the parameters $\delta_1 \ldots \delta_4$.

The robot model parameters were identified using the Levenberg-Marguardt method (lsqnonline) for the minimization criterion:

$$\min_q \|J(x)q + F(x)\|_2^2 \tag{6}$$

where $J(x)$ was a Jacobian vector of $F(x)$, and q was the direction of the search for the minimum function determined in accordance with the rule:

$$[J(x)^T J(x) + \alpha I]q = -J(x)F(x), \quad dla\ \alpha \geq 0 \tag{7}$$

The identification was performed for random start values of the identified parameter described by a normal distribution with the expected value $\mu = 1$ and variance $\sigma^2 = 1$ in the range $\langle 0, 1 \rangle$. The mean value from all tests of the identified parameter was then determined, along with the variance and standard deviation (Table2).

Table 2. Sample values of model parameters and identification process.

No. of test		σ_1	σ_2	σ_3	σ_4	Identification time [s]
1	Start values	0.16	0.52	0.96	0.37	
	Identified values	4.09	5.46	0.97	0.29	6.65
2	Start values	0.61	0.26	0.76	0.29	
	Identified values	3.14	4.06	0.81	0.24	6.01
3	Start values	0.59	0.55	0.92	0.29	
	Identified values	3.14	4.17	0.83	0.24	6.40
4	Start values	0.65	0.45	0.55	0.30	
	Identified values	3.49	4.63	0.85	0.24	5.92
Average		3.47	4.58	0.87	0.25	6.25
Variance		0.15	0.30	0.00	0.00	
Standard deviation		0.39	0.55	0.06	0.02	

2.3. Kinematic and the dynamic controllers

The task of tracking a reference trajectory by a robot has always been associated with the occurrence of errors - the difference between the reference position and the actual position. In order to minimise the errors, a proportional kinematic controller with a feedback loop, developed on the basis of [11], which was designed to obtain control signals that could direct the robot on a desired trajectory based on minimizing errors of the positions e_x, e_y, e_Θ described by equation (8) as a transformation of the difference between the reference position and orientation, denoted as $x(t)$, $y(t)$, $\Theta(t)$, and the actual position and orientation, denoted as $x_r(t)$, $y_r(t)$, $\Theta_r(t)$:

$$\begin{bmatrix} e_x(t) \\ e_y(t) \\ e_\theta(t) \end{bmatrix} = \begin{bmatrix} cos\theta(t) & sin\theta(t) & 0 \\ -sin\theta(t) & cos\theta(t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(t) - x_r(t) \\ y(t) - y_r(t) \\ \theta(\mathrm{t}) - \theta_r(t) \end{bmatrix} \tag{8}$$

The kinematic controller proposed on the basis of [11] took the form:

$$\begin{bmatrix} u_{kv}(t) \\ u_{k\omega}(t) \end{bmatrix} = \begin{bmatrix} -k_1 & 0 & 0 \\ 0 & -sign(v_r(t))k_2 & -k_3 \end{bmatrix} \begin{bmatrix} e_x(t) \\ e_y(t) \\ e_\theta(t) \end{bmatrix} \tag{9}$$

where: $u_{kv}(t)$ and $u_{k\omega}(t)$ were the control signals of the kinematic controller (linear and angular velocity) and its gains k_1, k_2, k_3 were determined from the following dependencies [11]:

$$\begin{gathered} k_1 = k_3 = 2\varepsilon\omega_n(t), \quad dla\ \varepsilon \in (0,1) \\ k_2 = b * |v_r(t)|, \quad b > 0 \\ \omega_n(t) = \sqrt{{\omega_r}^2(t) + b{v_r}^2(t)} \end{gathered} \tag{10}$$

where: ε – oscillations damping coefficient,
$\omega_n(t)$ – characteristic frequency,
b – additional adjustment coefficient,
$v_r(t)$ – actual linear velocity, and
$\omega_r(t)$ – actual angular velocity.

The assumption for the development of the dynamic controller was to treat the control signals from the kinematics controller (linear and rotational angular velocities) expressed by dependencies (9) as reference signals. The control law for the dynamic controller was developed on the basis of the inverse dynamics task (Appendix 2) using the parametric model of the robot dynamics (5) and the considerations presented in the paper [12]. The control law was described by the equation:

$$\begin{bmatrix} u_{dv}(t) \\ u_{d\omega}(t) \end{bmatrix} = \begin{bmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{bmatrix} \begin{bmatrix} \vartheta_1 \\ \vartheta_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 & v_r(t) & 0 \\ 0 & 0 & 0 & \omega_r(t) \end{bmatrix} [\delta_1 \quad \delta_2 \quad \delta_3 \quad \delta_4]^T$$

$$\begin{gathered} \vartheta_1 = \dot{v}(t) + k_v(u_{kv}(t) - v_r(t)) \\ \vartheta_2 = \dot{\omega}(t) + k_\omega(u_{k\omega}(t) - \omega_r(t)) \end{gathered} \tag{11}$$

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

3. CONTROL ALGORITHM TESTS

The correct operation of the TURTLEBOT 2 robot's trajectory tracking control algorithm and the correctness of the parameter identification for its mathematical model were first verified by simulation and then implemented on the actual robot. Tests were conducted for the circular trajectory described by the equation system:

$$\begin{bmatrix} x_d \\ y_d \\ \theta_d \end{bmatrix} = \begin{bmatrix} x_0 + R\sin(\omega t) \\ y_0 + 2\omega t \\ atan2(\dot{y}_d, \dot{x}_d) + k\pi \end{bmatrix} \tag{12}$$

where: x_0 =0, y_0 = 0, R =1 [m], ω = 0.1 [rad/s], k = 0 if the robot moves forward, k = 1 if the robot moves backward.

3.1. SIMULATION TESTS

The block diagram of the TURTLEBOT 2 follow-up motion simulation (Figure 4) was conducted in the MATLAB/Simulink environment, allowing the input of identified robot model parameters. The control system consisted of a set trajectory generator module determining position (x_d, y_d) and orientation Θ_d, a reference linear v_d and angular ω_d velocity generator, a kinematic controller described by equation (9), a dynamic controller described by equation (11) and a mathematical model of the robot described by equations (3) and (5).

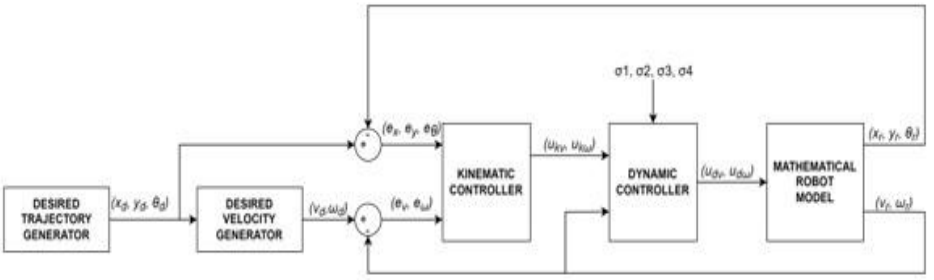

Fig. 4. Block diagram of the TURTLEBOT 2 trajectory tracking simulation

The simulation uses the average values of the model parameters shown in Table 2. A comparison between the set trajectory and the trajectory obtained from the simulation, taking into account the parameters identified by the Levenberg-Marguardt method is shown in Fig. 5, the position and orientation deviations are shown in Fig. 6a, while speed deviations are shown in Fig. 6b.

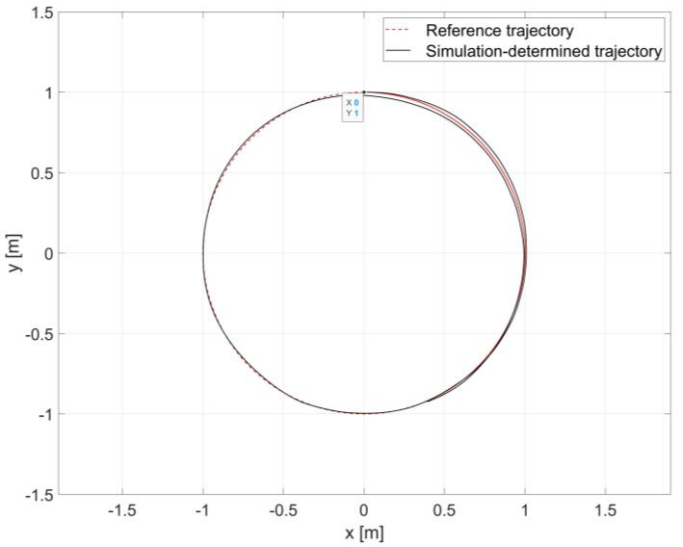

Fig. 5. Comparison of the desired trajectory and the trajectory obtained from the simulation

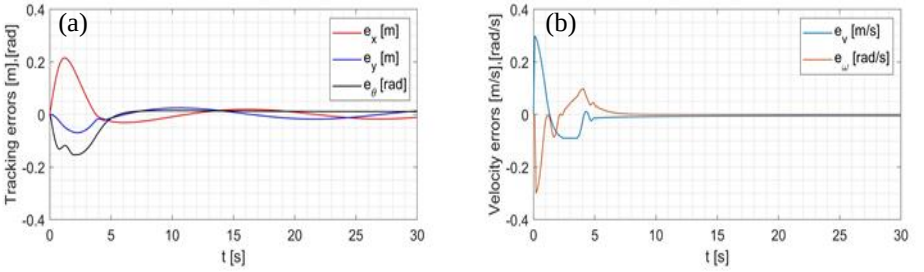

Fig. 6. Errors: a) tracking the reference trajectory, b) velocities

Based on the analysis of the simulation-determined errors shown in Fig. 6, it can be concluded that the controller based on the inverse dynamic task with the identified parameters by the Levenberg-Marguardt method performed trajectory tracking control with position and orientation (Figure 5a), and velocity (Figure 5b) errors not greater than:

- x-axis direction: $e_x = 0.031$ [m];
- x-axis direction: $e_y = 0.025$ [m];
- orientation: $e_\Theta = 0.014$ [rad];
- linear velocity: $e_v = 0.006$ [m/s];
- angular velocity: $e_\omega = 0.0003$ [rad/s].

The regulation time for a circular trajectory tracking task was equal to $t_r \approx 3.5$ [s].

3.2. Laboratory tests

The laboratory tests for trajectory tracking control of the real TURTLEBOT 2 were carried out on the basis of the control system shown in Figure 7. The system used a kinematic and dynamic controller implemented using the basic elements of the Simulink package. The Simulink/ROS library and two tools were used to communicate with the robot's operating system: Publisher - to publish a specific type of message in the declared communication channel, and Subscriber - for receiving messages from the robot system transmitted in the declared communication node. Using the Publisher, control signals were sent to the robot system in the form of linear and angular speed. Using the Subscriber, position and orientation were received from the robot system.

The laboratory tests used the average values of the model parameters shown in Table 2. A comparison between the reference and real trajectories for the TURTLEBOT 2 with the parameters identified by the Levenberg-Marguardt method were shown in Figure 8, the position and orientation errors were shown in Figure 9a, and the velocity errors were shown in Figure 9b.

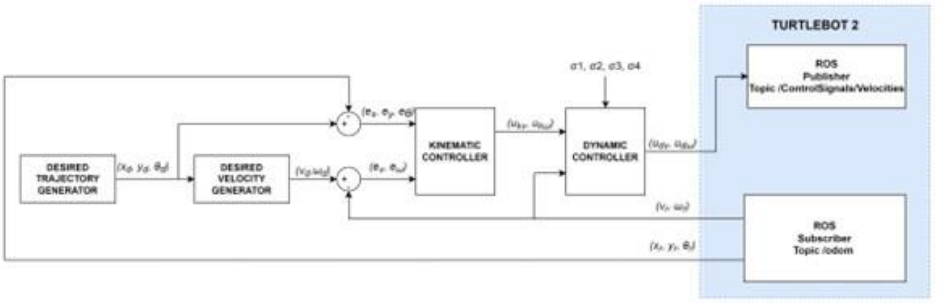

Fig. 7. Block diagram of the system for the laboratory test of the trajectory tracking

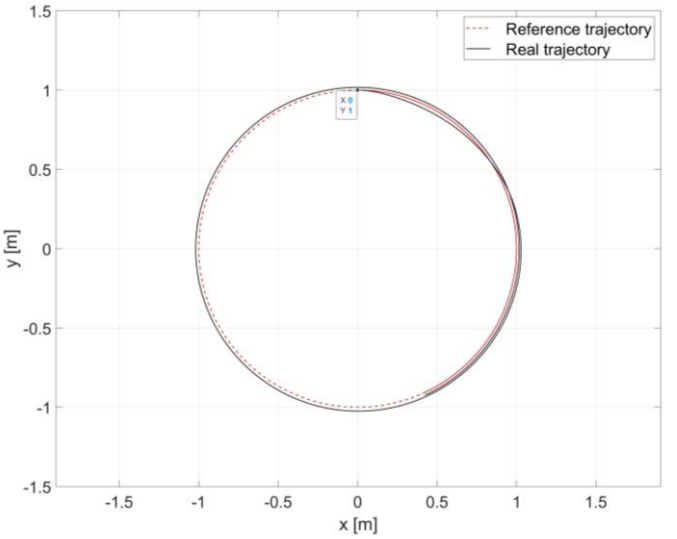

Fig. 8. Comparison between the reference and real trajectories in the laboratory test

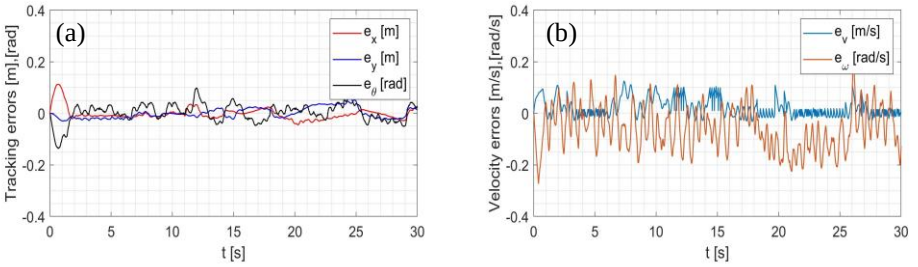

Fig. 9. Errors: a) tracking the reference trajectory, b) velocities

Based on the analysis of the errors determined in the laboratory tests shown in Fig. 9, it could be concluded that a controller based on the inverse dynamic task with the identified parameters by the Levenberg-Marguardt method performed trajectory tracking control with position and orientation (Figure 9a), and velocity (Figure 9b) errors not greater than:

- x-axis direction: $e_x = 0.045$ [m];
- x-axis direction: $e_y = 0.036$ [m];
- orientation: $e_\Theta = 0.1$ [rad];
- linear velocity: $e_v = 0.1$ [m/s];
- angular velocity: $e_\omega = 0.2$ [rad/s].

The regulation time for a circular trajectory tracking task was equal to $t_r \approx 1.7$ [s].

4. CONCLUSIONS

Tracking the trajectory by a mobile robot is a fundamental task of control algorithms, which allows more advanced robot functionality, including transport tasks, exploratory and reconnaissance tasks, or integration of multiple robots into a group. The control system should resist a number of errors from such sources as robot design inaccuracies, drive wear and tear, data transmission delays, and operating environment impact. The degree to which the above-mentioned errors were corrected is responsible for the accuracy of the robot achieving a desired position in the workspace.

In these studies, the use of a parametric model of the robot's dynamics made it possible to easily and quickly design a control system implementing the follow-up motion of the TURTLEBOT 2 robot. By changing the model parameters, the system is resistant to design changes or environmental impacts. Studies of the identified parameters have shown that, in the example indicated, the Levenberg-Marguardt method (lsqnonline) allowed the correct values of the identified parameters to be determined for their random start values in a very short time. This is very important when the user does not have any information about the parameter values of the robot model.

The presented comparative analysis of the reference and actual trajectories and the errors analysis confirmed the accuracy of the identified TURTLEBOT 2 parameters, and by means of the proposed control system the satisfactory trajectory tracking results were achieved with an average deviation in position $\bar{e}_{xy} \approx 30$ mm and orientation $\bar{e}_{\theta} = 0.05$ rad. However, if the robot's design is modified, e.g., by adding a laser scanner, additional batteries or replacing drives, the identification process must be carried out again. The way to solve this problem is to use on-line identification when parameters were determined during movement. The issues raised by the authors will be developed in this direction.

Appendix 1

In the case shown, the robot's pivot point *G* is located on the axis connecting the drive wheels, such that forces acting in the direction of the axis *y* do not produce any torque. Consequently, the system of equations (4) can be converted to:

$$\begin{aligned} \dot{v}_x &= \frac{F_{rlx} + F_{rrx}}{m} + v_y\omega \\ \dot{\omega} &= \frac{d(F_{rrx} - F_{rlx})}{2I_z} \end{aligned} \tag{13}$$

The forward v_x, side v_y and angular ω speed excluding slip were equal to:

$$\begin{aligned} v_x &= \frac{1}{2}r(\omega_r + \omega_l) \\ v_x &= 0 \\ \omega &= \frac{1}{2d}r(\omega_r - \omega_l) \end{aligned} \tag{14}$$

where r - radius of the drive wheels, ω_r, ω_l – angular velocity of the wheels (right and left).

The angular velocity of the wheels can be determined by converting the formulas to the driving moments of the right τ_r and left τ_l motors [8]:

$$\begin{aligned} \tau_r &= \frac{k_a(V_r - k_b\omega_r)}{R_a} \\ \tau_l &= \frac{k_a(V_l - k_b\omega_l)}{R_a} \end{aligned} \tag{15}$$

where V_r, V_l – supply voltages of the motors (right and left), k_b – electrical constant of the motors, k_a – mechanical constant of the motors, R_a – resistance of the motor windings.

The drive wheel dynamics equations [8] take the form of:

$$\begin{aligned} I_e\dot{\omega}_r + B_e\omega_r &= \tau_r - F_{rrx}r \\ I_e\dot{\omega}_l + B_e\omega_l &= \tau_l - F_{rlx}r \end{aligned} \tag{16}$$

where I_e – moment of inertia of the wheel in relation to the axis of rotation and B_e– viscous friction coefficient reduced to the motor shaft.

The equations of the PD controller implemented to regulate the motor supply voltage [8] take the form:

$$\begin{aligned} u_v &= k_{PT}(v_d - v_x) - k_{DT}\dot{v}_x \\ u_\omega &= k_{PR}(\omega_d - \omega) - k_{DR}\dot{\omega} \end{aligned} \tag{17}$$

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} – PD controller gains,
v_d – desired linear velocity, and
ω_d –desired angular velocity.

Substituting equation (15) into equation (16) and transforming it into the form for determining the longitudinal driving forces, the following system of equations was obtained:

$$\begin{aligned} F_{rrx} &= \frac{1}{r}\left(\frac{k_a}{R_a}(V_r - k_b\omega_r) - I_e\dot{\omega}_r + B_e\omega_r\right) \\ F_{rlx} &= \frac{1}{r}\left(\frac{k_a}{R_a}(V_l - k_b\omega_l) - I_e\dot{\omega}_l + B_e\omega_l\right) \end{aligned} \tag{18}$$

Then, substituting equations (18) to the system of equations (13), resulted in:

$$\begin{aligned} \dot{v}_x &= \frac{1}{rm}\left(\frac{k_a}{R_a}(V_l + V_r) + (\omega_l + \omega_r)\left(-\frac{k_a}{R_a} + B_e\right) - I_e(\omega_l \dot{+} \omega_r)\right) + v_y\omega \\ \dot{\omega} &= \frac{d}{{I_z}^2 r}\left(\frac{k_a}{R_a}(V_r - V_l) + (\omega_r + \omega_l)\left(-\frac{k_a}{R_a}k_b + B_e\right) - I_e(\omega_r \dot{+} \omega_l)\right) \end{aligned} \tag{19}$$

Combining the above equations with equations (14) and (17) meant that and when making algebraic transformations the expressions describing the dynamic model of the TURTLEBOT 2 robot were obtained:

$$\begin{aligned} \dot{v}_x\left(\frac{1}{k_{PT}}\left(\frac{R_a}{k_a}\left(\frac{rm}{2} + \frac{I_e}{r}\right) + k_{DT}\right)\right) &= -v_x\left(1 - \frac{1}{rk_{PT}}\left(1 + B_e\frac{R_a}{k_a}\right)\right) + v_d \\ \dot{\omega}_x\left(\frac{1}{k_{PR}}\left(\frac{R_a}{2k_a}\left(\frac{{I_z}^2 r}{d} + \frac{dI_e}{r}\right) + k_{DR}\right)\right) &= -\omega\left(\frac{d}{rk_{PR}}\left(1 - \frac{R_a}{k_a}\right) + 1\right) + \omega_d \end{aligned} \tag{20}$$

or in abbreviated form:

$$\begin{aligned} \dot{v}_x\delta_1 &= -v_x\delta_3 + v_d \\ \dot{\omega}_x\delta_2 &= -\omega\delta_4 + \omega_d \end{aligned} \tag{21}$$

Appendix 2

Multiplying the equation (5) by $\begin{bmatrix}\delta_1 & 0\\ 0 & \delta_2\end{bmatrix}$ on both sides gave the form:

$$\begin{bmatrix}\delta_1 & 0\\ 0 & \delta_2\end{bmatrix}\begin{bmatrix}\dot{v}_x\\ \dot{\omega}\end{bmatrix}=\begin{bmatrix}-\delta_3 v_x\\ -\delta_4\omega\end{bmatrix}+\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}v_d\\ \omega_d\end{bmatrix} \tag{22}$$

When making transformations, the above expression can be written as:

$$\begin{bmatrix}\delta_1 & 0\\ 0 & \delta_2\end{bmatrix}\begin{bmatrix}\dot{v}_x\\ \dot{\omega}\end{bmatrix}+\begin{bmatrix}\delta_3\\ \delta_4\end{bmatrix}\begin{bmatrix}v_x & 0\\ 0 & \omega\end{bmatrix}=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}v_d\\ \omega_d\end{bmatrix} \tag{23}$$

and then adjusted to a parametric form:

$$\begin{bmatrix}v_d\\ \omega_d\end{bmatrix}=\begin{bmatrix}\delta_1 & 0\\ 0 & \delta_2\end{bmatrix}\begin{bmatrix}\dot{v}_x\\ \dot{\omega}\end{bmatrix}+\begin{bmatrix}0 & 0 & v_x & 0\\ 0 & 0 & 0 & \omega\end{bmatrix}[\sigma_1 \quad \sigma_2 \quad \sigma_3 \quad \sigma_4]^T \tag{24}$$

Based on the inverse dynamic task [12], the above expression can be used in the form of a dynamic controller (11).

FUNDING

This work was co-financed by the Military University of Technology (Warsaw, Poland) within the framework of the research project UGB 893/2021 "Study of algorithms for tracking the position of objects in 3D space using inertial and spatial measurement techniques".

REFERENCES

[1] Wiśniewski, Marcin. 2004. ”Propozycja metody pomiaru dokładności i powtarzalności pozycjonowania robotów przemysłowych w warunkach przemysłowych”. *Technologia i Automatyzacja Montażu* 3 : 39-43.

[2] Dudka, Piotr. 2016. “Czynniki wpływające na dokładność i powtarzalność pozycjonowania robota przemysłowego”. *Pomiary, Automatyka, Robotyka* 20 (4) : 59-66.

[3] https://www.roscomponents.com/en/mobile-robots/9-turtlebot-2.html

[4] Giergiel, Mariusz, Zenon Hendzel, Wiesław Żylski. 2002. *Modelowanie i sterowanie mobilnych robotów kołowych*. Warszawa: PWN.

[5] Giergiel, Mariusz, Piotr Małka. 2006. “Modelowanie kinematyki i dynamiki mobilnego minirobota”. *Modelowanie Inżynierskie* 32 : 157-162.

[6] Małka, Piotr. 2007. *Pozycjonowanie i nadążanie minirobota kołowego,* (praca doktorska*).* Kraków, Akademia Górniczo-Hutnicza.

[7] Siegwart, Roland, and Illah R Nourbakhsh. 2004. *Introduction to Autonomous Mobile Robots*. England, London: The MIT Press Cambridge Massachusetts.

[8] De La Cruz, Celso, and Ricardo Carelli. 2008. "Dynamic model based formation control and obstacle voidance of multi-robot systems". *Robotica* 26 (3) : 345-356.

[9] Martins, Nascimento Filipe, Mario Sarcinelli-Filho, and Ricardo Carelli. 2017. "A Velocity-Based Dynamic Model and Its Properties for Differential Drive Mobile Robots". *Journal of Intelligent & Robotic Systems* 85 (2) : 277-292.

[10] Zhang, Yulin, Daehie Hong, Jae H. Chung, and Steven A. Velinski. 1998. Dynamic Model Based Robust Tracking Control of a Differential Steered Wheeled Mobile Robot. In *Proceedings of the 1998 American Control Conference, ACC 1998*, 850-855.

[11] Klancar, Gregor, D. Matko, and Saso Blazic. 2005. Mobile Robot Control on a Reference Path. In *Proceedings of the 2005 IEEE International Symposium, Mediterrean Conference on Control and Automation*, 1343-1348.

[12] Martins, Nascimento Filipe, Wanderley C. Celeste, Ricardo Carelli, Mario Sarcinelli-Filho, and Teodiano F. Bastos-Filho. 2008. "An adaptive dynamic controller for autonomous mobile robot trajectory tracking". *Control Engineering Practice* 16 (11) : 1354-1363.

[13] Martins, Nascimento Filipe, and Alexandre S. Brandao. 2019. Motion control and Velocity based Dynamic Compensation for Mobile Robots. In *Applications of Mobile Robots* 23-41.

[14] Das, Tamoghna, and Indra Narayan Kar. 2006. "Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots". *IEEE Transaction on Control Systems Technology* 3 (14) : 501-510.

[15] Hendzel, Zenon, and Łukasz Rykała. 2017. "Modeling of dynamics of wheeled mobile robot with mecanum wheels with the use of Lagrange Equations of the Second Kind". *International Journal of Applied Mechanics and Engineering* 1 (22) : 81-99.

[16] Siwek, Michał, Leszek Baranowski, Jarosław Panasiuk, and Wojciech Kaczmarek. 2019. "Modeling and Simulation of Movement of Dispersed Group of Mobile Robots Using Simscape Multibody Software". *AIP Conference Proceedings* 2078, 020045-1-5.

[17] Siwek, Michał, and Leszek Baranowski. 2018. "Use of 3D simulation to design theoretical and real pipe inspection mobile robot model". *Acta Mechanica et Automatica* 12 (3) : 232-236.

[18] Panasiuk, Jarosław, Michał Siwek, Wojciech Kaczmarek, Szymon Borys, and Piotr Prusaczyk. 2018. "The concept of using the mobile robot for telemechanical wires installation in pipelines". *AIP Conference Proceedings* 2029, 020054 -1-9.
[19] Besseghieur, L. Khadir, Radosław Trębiński, Wojciech Kaczmarek, and Jarosław Panasiuk. 2019. Leader-follower formation control for a group of ROS-enabled mobile robots. In *Proceedings of 6th International Conference on Control, Decision and Information Technologies* 2078, 020045.

Sterowanie w ruchu nadążnym robota mobilnego z systemem ROS

Michał SIWEK, Jarosław PANASIUK, Leszek BARANOWSKI, Wojciech KACZMAREK, Szymon BORYS

Wojskowa Akademia Techniczna
ul. Sylwestra Kaliskiego 2, 00-908 Warszawa

Streszczenie. W artykule przedstawiono badania symulacyjne i laboratoryjne systemu sterowania dwukołowym robotem mobilnym o napędzie różnicowym, z systemem ROS. Autorzy zaproponowali podejście projektowania systemu sterowania w oparciu o parametryczny model dynamiki robota. Wartości nieznanych parametrów modelu dynamiki wyznaczono przeprowadzając ich identyfikację metodą Levenberga-Marguardta. Następnie porównując trajektorie zadane z otrzymanymi na drodze badań symulacyjnych i laboratoryjnych, a także na podstawie analizy uchybów określono poprawność procesu identyfikacji parametrów modelu i działania systemu sterowania.
Słowa kluczowe: robot mobilny, ROS, model parametryczny, identyfikacja, śledzenie trajektorii