VOLUME 17, N° 3 2023
Journal of Automation, Mobile Robotics and Intelligent Systems

Adaptive and Robust Following of 3D Paths by a Holonomic Manipulator

Submitted: 28th March 2023; accepted: 31st July 2023

Alicja Mazur, Mirela Kaczmarek

DOI: 10.14313/JAMRIS/3-2023/23

Abstract:
This paper addresses the problem of the following three-dimensional path by holonomic manipulator with parametric or structural uncertainty in the dynamics. Description of the manipulator relative to a desired three-dimensional path was presented. The path is parameterized orthogonally to the Serret-Frenet frame, which is moving along the curve. The adaptive and robust control laws for a stationary manipulator which ensures realization of the task were specified. Theoretical considerations are supported by the results of computer simulations conducted for an RTR manipulator.

Keywords: *Path following, Serret–Frenet parametrization, Orthogonal projection, Backstepping algorithm, Holonomic manipulator*

1. Introduction

Over the years, the use of robots has been increasing, especially in industry. Manipulators are able to realize different tasks, including welding, painting, assembly and palletizing, among others. During those tasks, high endurance, speed, and precision are required.

Taking into account the control point of view, three types of tasks for industrial manipulators can be defined: point stabilization; trajectory tracking when the robot has to follow a desired curve which is time-parametrized, and path following – the robot has to follow a curve parametrized by a curvilinear distance from a fixed point. During the trajectory tracking task, the robot's particular position at a prespecified time is required. To the contrary, a path following task requires the robot to converge to a geometric curve with any feasible speed profile. In the paper, only the following desired path, i.e. a curve parameterized by curvilinear distance, has been considered.

Recently, path-following tasks have been discussed many times for different robots: usually for mobile platform [9–11], but also for fixed base manipulators and mobile manipulators [2, 3, 6, 8]. Most of the papers deal only with two-dimensional paths. Moreover, algorithms presented in the literature are devoted to robots with fully known dynamic models.

When the full model of manipulator dynamics is unknown, there are two possibilities – structural uncertainty, when mathematical expressions of some forces in a dynamical model are unknown, and parametric uncertainty, when we do not know certain number of model parameters. In this article, both types are being considered.

Developing research presented in the article [7] will be continued in this paper. In the work, a general solution to the tracking of three-dimensional curves for manipulators with different level of dynamics knowledge has been proposed. In Section 2, models of dynamics with different types of uncertainties have been presented. General equations of the robot's motion describing its position relative to the path have been established in Section 3. Control problem formulated in the paper has been presented in Section 4. Since the system equations have a cascade structure, the control will consist of a kinematic (Section 5) and a dynamic controller (Sections 6–8). The main result is a class of dynamic controllers dedicated to different levels of fixed-base manipulator dynamics knowledge. In Section 6, dynamic control algorithm for fully known manipulator has been presented, Section 7 contains a dynamic control law for the case of parametric uncertainty in the model, and Section 8 presents a robust version of the dynamic algorithm for parametric and structural uncertainty in the dynamics. All considerations were illustrated with simulations for an RTR manipulator, presented in Section 9. Section 10 contains a brief summary of the results.

This paper is an extension of the conference paper [5]. Other methods of the time-dependent description of curvilinear distance measured along the path have been investigated. Moreover, the problem of parametric and structural uncertainties occurring in the dynamics has been considered. Finally, an adaptive and robust version of the dynamic controller has been introduced. The proofs of asymptotic stability of the proposed algorithms have been shown.

2. Dynamics of a Holonomic Manipulator

Typically, it is assumed that the dynamics of the manipulator are fully known. This approach is the starting point for designing subsequent control algorithms that require less and less knowledge about the dynamics of the object. In practice, such a situation is rare, because it requires fully identified dynamics of the object (identification process conducted before regulation gives us all parameters of the dynamics) and moreover, that the robot does not carry the payload during operation (mass and moment of inertia of the unknown payload are added to the parameters of the last link, which contradicts the full knowledge of dynamics).

A typical situation that we face during control is uncertainty about the dynamics model. When the full model of the manipulator is unknown, there are two possibilities – the structural uncertainty, when forms of functions describing some elements of the dynamical model are unknown (or some "impacts" – forces or torques – are not included in the model), and the parametric uncertainty, when we do not know certain number of model parameters.

The control for each of these cases requires a different dynamic algorithm. For this reason, three different control algorithms will be presented: for full knowledge of the model (a non-adaptive algorithm), for parametric uncertainty (an adaptive algorithm with full or partial parameterization of the model), and for structural uncertainty (a robust algorithm using a sliding mode approach [13] simultaneously solving the problem of parametric and structural uncertainty).

For each of the mentioned types of algorithms, the dynamics model should be presented in a slightly different way. This will be presented in the section.

2.1. Model of the Fully Known Manipulator

Suppose that the model of the holonomic manipulator with n degrees of freedom is fully known. Then, the robot dynamics are described in joint coordinates $\theta = (\theta_1, ..., \theta_n)$ in the following form

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + D(\theta) = u, \tag{1}$$

where the left-hand side of the expression describes the dynamics of the manipulator with the elements:

- $\theta, \dot{\theta}, \ddot{\theta} \in R^n$ – vectors of joint positions, velocities and accelerations,
- $M(\theta)$ – symmetrical, positive definite inertia matrix,
- $C(\theta,\dot{\theta})$ – matrix of Coriolis and centripetal forces,
- $D(\theta)$ – vector of gravitational forces.

The right-hand side of (1) includes control vector u.

It is clear that the above model describes only serial chain manipulators with neglected dissipative interactions, such as friction forces. This approach is not very restrictive because friction forces can be included in the model in a form linearly dependent on unknown parameters, and in the article, they have been omitted only for the sake of simplicity.

2.2. Model of the Manipulator with Parametric Uncertainty

When the full model of the manipulator is unknown, there are two possibilities – structural uncertainty, when some forces acting on the manipulator are not included in the mathematical model of dynamics, and parametric uncertainty, when we know functions describing real forces but a certain number of model parameters standing before functions expressing forces are unknown. At the beginning, parametric uncertainty is being considered.

From the control point of view, it is crucial to present the dynamics model as linearly dependent on unknown parameters as follows

$$\begin{aligned} M(\theta,a)\ddot{\theta} + C(\theta,\dot{\theta},a)\dot{\theta} + D(\theta,a) &= \\ Y(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})a &= u, \end{aligned} \tag{2}$$

where $a \in R^p$ is a vector of unknown parameters and matrix $Y(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})$ is a so-called *regression matrix*.

The first argument of the regression matrix Y defines trajectory, along which the model is described; the second component defines velocity occuring in Coriolis matrix; the third component gives the vector by which the Coriolis matrix is multiplied; and the last component gives the vector by which the inertia matrix is multiplied.

When all parameters a are unknown, then the model (2) is called **fully parametrized**. However, only some parameters a are unknown, and parts of the dynamic model can be represented as

$$\begin{aligned} M(\theta,a) &= M_0(\theta) + M_1(\theta,a), \\ C(\theta,\dot{\theta},a) &= C_0(\theta,\dot{\theta}) + C_1(\theta,\dot{\theta},a), \\ D(\theta,a) &= D_0(\theta) + D_1(\theta,a), \end{aligned}$$

where $M_0(\theta)$, $C_0(\theta,\dot{\theta})$, and $D_0(\theta)$ represent known parts of model, while $M_1(\theta,a)$, $C_1(\theta,\dot{\theta},a)$, and $D_1(\theta,a)$ include unknown parameters in vector a. Then, for partial knowledge of the model, the dynamic equation has the following form

$$M_0(\theta)\ddot{\theta} + C_0(\dot{\theta},\theta)\dot{\theta} + D_0(\theta) + Y_1(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})a = u, \tag{3}$$

where

$$Y_1(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})a = M_1(\theta,a)\ddot{\theta} + C_1(\theta,\dot{\theta},a)\dot{\theta} + D_1(\theta,a).$$

In further considerations, we will mention the model (3) as dynamics with **partial parametrization**.

2.3. Model of the Manipulator with Structural and Parametric Uncertainty

In the case under consideration, a model with both parametric and structural uncertainty will be presented. The idea is to get the dynamics corresponding to all possible situations that can happen in practice.

Suppose that the dynamics have parametric and structural uncertainties, i.e., the model has a form

$$\begin{aligned} M(\theta,a)\ddot{\theta} + C(\theta,\dot{\theta},a)\dot{\theta} + D(\theta,a) + \delta &= \\ M_0(\theta)\ddot{\theta} + C_0(\dot{\theta},\theta)\dot{\theta} + D_0(\theta) + Y_2(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})a + \delta &= \\ &u. \end{aligned} \tag{4}$$

The symbol δ represents an unknown part of the model, e.g., forces not included in the dynamics equations. For further considerations, some assumption has to be taken.

Assumption 1 Unknown part of the dynamics δ is bounded by physical reasons. However, an estimate for the structural uncertainty is known, i.e.,

$$\parallel \delta \parallel \leq A \in R_+. \tag{5}$$

3. Equation of Robot Motion Relative to a Path

A path-following task requires describing the robot's motion relative to an object moving along a curve. To obtain such a description, we will use the Serret-Frenet orthogonal parameterization. Using this parameterization, we get the equations that must be met for the system to correctly follow the desired path.

3.1. Serret-Frenet Parametrization for 3D Curve

Let's consider the manipulator's movement along a given curve

$$r(s) = (r_1(s), r_2(s), r_3(s))^T \tag{6}$$

in three-dimensional space, as in Figure 1. Point M describes the position of the manipulator's end-effector and can be defined by Cartesian coordinates $p = (x, y, z)^T$ expressed relative to base body-fixed frame $X_0Y_0Z_0$. In some distance s calculated along the path, the Serret-Frenet frame $Q(s)$ should be located. Parameter s is so-called curvilinear distance which may be interpreted as the length of a string laying perfectly on the path. The Serret-Frenet frame is an orthonormal basis of 3-vectors: $T(s)$ – the unit tangent, $N(s)$ – the unit normal, and $B(s)$ – the unit binormal, defined as follows

$$T = \frac{dr}{ds}, \quad N = \frac{\frac{dT}{ds}}{\|\frac{dT}{ds}\|}, \quad B = T \times N. \tag{7}$$

Vectors $T(s)$, $N(s)$, $B(s)$ are completely determined by the curvature $c(s)$ and torsion $\tau(s)$ of the three-dimensional curve as a function of s. The curvature of a plain curve in some point is equal to the inversion of the radius of such a circle, which is tangent to the curve in the same point, and can be calculated from the definition as follows

$$c(s) = \left\|\frac{d}{ds}\frac{dr}{ds}\right\| = \left\|\frac{d^2r}{ds^2}\right\| = \sqrt{\left(\frac{dr_1^2}{ds^2}\right)^2 + \left(\frac{dr_2^2}{ds^2}\right)^2}. \tag{8}$$

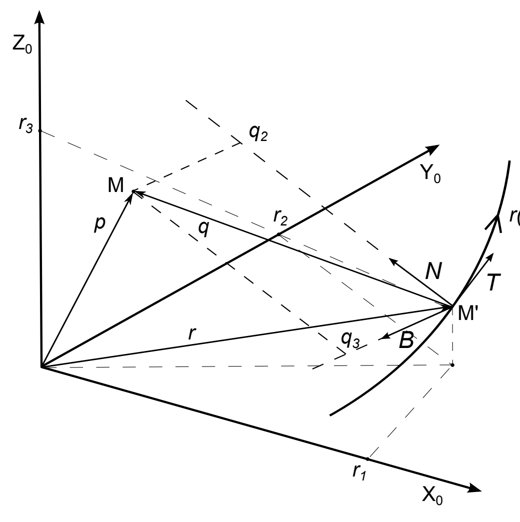

Figure 1. Illustration of path tracking problem using three-dimensional Serret-Frenet frame with orthogonal projection on a path

In turn, torsion defines how much the curve swerves from the plane and is defined in the following way

$$\tau(s) = \left\|\frac{dB}{ds}\right\| = \frac{1}{c^2(s)}\left(\frac{dr}{ds} \times \frac{d^2}{ds^2}, \frac{d^3r}{ds^3}\right), \tag{9}$$

where $(\cdot\,,\,\cdot)$ denotes dot product of two vectors.

According to the definitions of curvature and torsion, motion of the the Serret-Frenet frame $Q(s) = [T(s), N(s), B(s)]$ defined along a given path can be expressed by Serret-Frenet matrix $K(s)$ equations (using curvilinear distance s) as follows

$$\frac{dQ}{ds} = Q(s)K(s), \tag{10}$$

which can be rewritten in matrix form

$$\begin{pmatrix} \frac{dT}{ds} \\ \frac{dN}{ds} \\ \frac{dB}{ds} \end{pmatrix} = \begin{bmatrix} 0 & c(s) & 0 \\ -c(s) & 0 & \tau(s) \\ 0 & -\tau(s) & 0 \end{bmatrix} \cdot \begin{pmatrix} T \\ N \\ B \end{pmatrix}$$

$$= K(s) \cdot \begin{pmatrix} T \\ N \\ B \end{pmatrix}. \tag{11}$$

3.2. Equation of Robot Motion Relative to Path

Considering the manipulator's movement along a given curve, as it has been presented in Figure 1, we can observe that the coordinates of point M relative to the Serret-Frenet frame are equal to $q = (q_1, q_2, q_3)^T$. Whereas in the normal plane (spanned by N and B unit vectors) the position of the same point is defined by coordinates $(q_2, q_3)^T$. To describe a robot's motion relative to the moving Serret-Frenet frame it is necessary to obtain $(s, q_2, q_3)^T$ coordinates. According to the above assumption, to locate point M in the normal plane of the path, the following condition has to be satisfied

$$p - r \perp T \quad \Rightarrow \quad (T, p - r) = 0. \tag{12}$$

After making some more transformations, which are presented with details in [7], the following equations describing robot position relative to the moving Serret-Frenet frame were obtained

$$\dot{s} = -\frac{(T, \dot{p} - \dot{r})}{c(N, p - r)}, \tag{13}$$

$$\dot{q}_2 = \left(N - \frac{\tau}{c}\frac{(B, p - r)}{(N, p - r)}T, \dot{p} - \dot{r}\right), \tag{14}$$

$$\dot{q}_3 = \left(B + \frac{\tau}{c}T, \dot{p} - \dot{r}\right) \tag{15}$$

$$\dot{T} = -\frac{(T, \dot{p} - \dot{r})}{(N, p - r)}N, \tag{16}$$

$$\dot{N} = \frac{(T, \dot{p} - \dot{r})}{(N, p - r)}\left(T - \frac{\tau}{c}B\right), \tag{17}$$

$$\dot{B} = \frac{\tau}{c}\frac{(T, \dot{p} - \dot{r})}{(N, p - r)}N. \tag{18}$$

The above expressions are the point of departure to design control algorithms for three-dimensional path tracking. The crucial point to note here is that vector $p = (x, y, z)^T$ describes the robot's Cartesian position relative to base body-fixed frame $X_0Y_0Z_0$, vector $r = (r_1, r_2, r_3)^T$ describes the given path in R^3 relative to the same base body-fixed frame and $(s, q_2, q_3)^T$ are coordinates of the robot relative to the path.

3.3. Description of Manipulator Moving Along the Curve

The Equations (13,18) can be rewritten as follows

$$\begin{aligned} \dot{s} &= -\frac{(T, \dot{p} - \dot{r})}{c(N, p - r)} = P_1\dot{p} + R_1, & (19) \\ \dot{q}_2 &= \left(N - \frac{\tau}{c}\frac{(B, p - r)}{(N, p - r)}T, \dot{p} - \dot{r}\right) \\ &= P_2\dot{p} + R_2, & (20) \\ \dot{q}_3 &= \left(B + \frac{\tau}{c}T, \dot{p} - \dot{r}\right) = P_3\dot{p} + R_3, & (21) \end{aligned}$$

with the elements equal to

$$\begin{aligned} P_1 &= -\frac{T^T}{c(N, p - r)}, \\ P_2 &= \left(N - \frac{\tau}{c}\frac{(B, p - r)}{(N, p - r)}T\right)^T, \\ P_3 &= \left(B + \frac{\tau}{c}T\right)^T, \\ R_1 &= \frac{T, \dot{r}}{c(N, p - r)}, \\ R_2 &= -\left(N - \frac{\tau}{c}\frac{(B, p - r)}{(N, p - r)}T, \dot{r}\right), \\ R_3 &= -\left(B + \frac{\tau}{c}T, \dot{r}\right). \end{aligned}$$

Let's introduce the following notation

$$\xi = \begin{pmatrix} s \\ q_2 \\ q_3 \end{pmatrix}, \quad P = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}, \quad R = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix}.$$

Then, the equations (19,21) can be expressed in the matrix form as below

$$\dot{\xi} = P\dot{p} + R. \tag{22}$$

It is easy to see that Cartesian coordinates p of the end-effector are functions of joint variables, given by manipulator's kinematics

$$p = k(\theta), \tag{23}$$

so $\dot{p}$ depends on joint velocities in the following manner

$$\dot{p} = \frac{\partial k}{\partial \theta}\dot{\theta} = J(\theta)\dot{\theta}, \tag{24}$$

where $J(\theta)$ is the Jacobi matrix for position coordinates. Substituting (24) into (22) we obtain the expression

$$\dot{\xi} = PJ\dot{\theta} + R, \tag{25}$$

where signal $\dot{\theta}$ plays a role of a control input.

4. Control Problem Statement

As it was mentioned in the introduction, this paper addresses the following control problem:

A fixed-based manipulator with parametric or structural uncertainty in the dynamics should follow the desired smooth path defined in R^3 space.

The essential issue to note here is that the complete mathematical equations describing the manipulator relative to desired curve in R^3 has a **cascaded structure** consisting of two groups of equations:
- kinematics (25) – description of robot motion relative to the path (plays a role of constraints) and
- dynamics (1).

Because the model structure is a cascade, it is necessary to use the control method intended for cascaded systems, i.e., backstepping integrator method [4]. Therefore structure of the path-following controller is divided into two parts due to a backstepping-like procedure, as in Figure 2:

- Kinematic controller θ_{ref} – represents a vector of embedded control inputs, which ensure the realization of the task for the geometric path tracking problem if the dynamics were not present. 'Velocity profile', which can be executed in practice, to follow the desired curve in R^3 is generated.
- Dynamic controller – as a consequence of the cascaded structure of the system model, the system's velocities cannot be commanded directly, as it is assumed in the design of the kinematic controller, and instead, they must be realized as the output of the dynamics driven by u.

Using those controllers working simultaneously, it is possible to solve the presented control problem for the manipulators.

5. Kinematic Control Algorithm

To ensure that the Jacobi matrix $J(\theta)$ is invertible, we can assume that the manipulator is non-redundant and the desired path does not pass through singular configurations of the manipulator.

Matrix P is invertible if the following condition is satisfied

$$\det\ P = \frac{-1}{c(s)(N, p - r)} \neq 0. \tag{26}$$

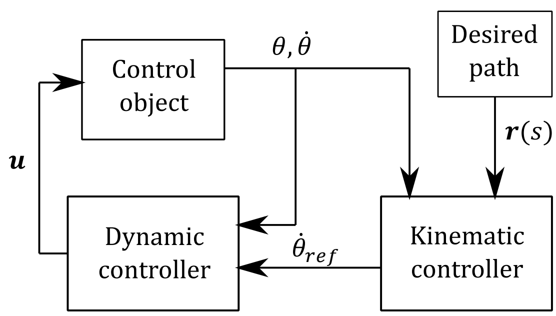

Figure 2. Structure of the proposed control algorithm

Cases where $\det P = 0$ are called **singularities of Serret-Frenet orthogonal parameterization**, e.g.,

- $c(s) = 0$ – curvature of the curve is equal 0, i.e., the curve is infact the straight line,
- $p - r = 0$ – at the end of regulation process gripper is located strictly on the path.

The weak side of the Serret-Frenet orthogonal parameterization is the fact that it does not allow straight lines to be traced. This problem can be solved in another way, namely by moving the singularity to another place, not necessarily lying on the path. The singularity shifting procedure in the orthogonal parameterizations has been presented in the paper [1].

According to the above remark, the following kinematic control algorithm can be proposed

$$\dot{\theta}_{ref} = J^{-1}P^{-1}\left(\dot{\xi}_d - K_p e_\xi - R\right), \quad e_\xi = \xi - \xi_d, \quad (27)$$

with a positive definite regulation matrix $K_p > 0$. Vector $\xi_d = (s_d(t), q_{2d}, q_{3d})$ where usually $q_{2d} = 0$, $q_{3d} = 0$ and $s_d(t)$ – desired path parametrization (dependency on time) can be an arbitrary function, depending on the designer's choice. Signal $\dot{\theta}_{ref}$ is proposed velocity of the robot's joints, i.e. 'velocity profile' coming from the kinematic controller – motion planning subsystem. Such velocity has to be realized on a dynamic level.

5.1. Proof of the Convergence

After substituting kinematic control law (27) to the constraints equations (25), we get a kinematic closed-loop system in the form

$$\begin{aligned} \dot{\xi} &= PJJ^{-1}P^{-1}\left(\dot{\xi}_d - K_p e_\xi - R\right) + R \\ \dot{\xi} &= \dot{\xi}_d - K_p e_\xi, \end{aligned} \quad (28)$$

or equivalently, after moving to one side,

$$\begin{aligned} \dot{\xi} - \dot{\xi}_d + K_p e_\xi &= 0, \\ \dot{e}_\xi + K_p e_\xi &= 0. \end{aligned} \quad (29)$$

Let's propose the following Lyapunov-like function

$$V_1(e_\xi) = \frac{1}{2} e_\xi^T e_\xi. \quad (30)$$

The time derivative of this function calculated along trajectories of the closed-loop system (29) is equal to

$$\dot{V}_1 = e_\xi^T \dot{e}_\xi = e_\xi^T(-K_p e_\xi) = -e_\xi^T K_p e_\xi \leq 0. \quad (31)$$

From LaSalle's invariance principle [4], we know that the equilibrium point of (31) is equal to

$$e_\xi = 0.$$

This ends the proof of asymptotic stability of the kinematic controller.

6. Dynamic Controller – Full Knowledge of the Manipulator's Dynamics

When the manipulator's dynamics are fully known, then non-adaptive version of the dynamic control law has been proposed:

$$u = M(\theta)\ddot{\theta}_{ref} + C(\theta, \dot{\theta})\dot{\theta}_{ref} + D(\theta) - K_d \dot{e}_\theta. \quad (32)$$

In the above equation $\dot{\theta}_{ref}$ is a control signal (27) coming from the kinematic controller, K_d is the positive definite matrix of regulation parameters and $\dot{e}_\theta = \dot{\theta} - \dot{\theta}_{ref}$.

6.1. Proof of the Convergence

After substituting non-adaptive control law (32) into the model of fully known dynamics (1), we obtain the closed-loop system

$$M(\theta)\ddot{e}_\theta + C(\theta, \dot{\theta})\dot{e}_\theta + K_d \dot{e}_\theta = 0. \quad (33)$$

Let's propose the following Lyapunov-like function

$$V_2(e_\xi, \dot{e}_\theta) = V_1(e_\xi) + \frac{1}{2}\dot{e}_\theta^T M(\theta)\dot{e}_\theta. \quad (34)$$

The time derivative of this function calculated along trajectories of the closed-loop system (33) and (29) is equal to

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + \dot{e}_\theta^T M(\theta)\ddot{e}_\theta + \frac{1}{2}\dot{e}_\theta^T \dot{M}(\theta)\dot{e}_\theta \\ &= \dot{V}_1 + \dot{e}_\theta^T(-C(\theta, \dot{\theta})\dot{e}_\theta - K_d \dot{e}_\theta) + \frac{1}{2}\dot{e}_\theta^T \dot{M}(\theta)\dot{e}_\theta. \end{aligned}$$

Using skew-symmetry between inertia and Coriolis matrices, we get

$$\dot{V}_2 = -e_\xi^T K_p e_\xi - \dot{e}_\theta^T K_d \dot{e}_\theta \leq 0. \quad (35)$$

Again, from LaSalle's invariance principle, it can be concluded that the equilibrium point of the control algorithm is

$$(e_\xi, \dot{e}_\theta) = (0, 0).$$

This ends the proof of asymptotic stability of the non-adaptive control algorithm for the cascaded system (25)-(1).

7. Adaptive Dynamic Controller – Parametric Uncertainty of the Manipulator's Dynamics

The parametric uncertainty applies only to the dynamics model, and the unknown parameters are most often related to the payload being carried. In order to design an adaptive control algorithm that performs the path-following task, it is necessary to [12]:

- assume that an estimate of the unknown parameters $\hat{a}(t)$ is available at any time,
- find the u control algorithm that uses the current estimates of the unknown parameters,
- find an algorithm for estimating unknown parameters and
- prove the asymptotic stability of the proposed adaptive control system consisting of a control subsystem and a subsystem for estimating unknown parameters.

Since the unacquaintance of the dynamics parameters of the object does not affect the kinematic control, the considerations presented in Section 6 regarding the control structure and the kinematic controller are still valid.

The kinematic controller for obtaining the velocity profile is therefore given by the equation (27). Due to the presence of unknown parameters a, it is necessary to use the adaptive version of the algorithm (32), preferably in version (3) for partial parameterization of the model.

When the full model of the manipulator is unknown, it is crucial to design a proper dynamic controller. As mentioned before, in this article, the parametric uncertainty, when the certain number of model parameters is unknown, is considered. Parametric uncertainty applies only to the dynamics model and does not affect the kinematic controller.

For the realization of motion $\xi_d(t)$ (path tracking with time regime), the dynamic adaptive control algorithm can be proposed

$$\begin{aligned} u \;&=\; M_0(\theta)\ddot{\theta}_{ref} + C_0(\theta,\dot{\theta})\dot{\theta}_{ref} + D_0(\theta) + \\ &+Y_1(\theta,\dot{\theta},\dot{\theta}_{ref},\ddot{\theta}_{ref})\hat{a}(t) - K_d\dot{e}_\theta, \end{aligned} \quad (36)$$

where M_0, C_0, D_0 represent known parts of the model, $Y_1 \cdot \hat{a}(t)$ is unknown one, and K_d is a positive definite regulation matrix. Y_1 is the so-called regression matrix, and $\hat{a}(t)$ is a vector of time estimates of unknown coefficients of the robot model generated by the so-called adaptation law. These estimates are calculated as follows

$$\dot{\hat{a}}(t) = \dot{\tilde{a}}(t) = -\Gamma Y_1^T(\theta,\dot{\theta},\dot{\theta}_{ref},\ddot{\theta}_{ref})\dot{e}_\theta, \quad (37)$$

where $\Gamma = \Gamma^T > 0$ is positive definite matrix of adaptation gains and $\tilde{a}(t) = \hat{a}(t) - a$ is a vector of parameter errors. Vector a represents unknown but constant real parameters of the dynamics.

7.1. Proof of the Convergence

In the case of partial parametrization of the model, the dynamics equation of the manipulator takes the form

$$M_0(\theta)\ddot{\theta} + C_0(\dot{\theta},\theta)\dot{\theta} + D_0(\theta) + Y_1(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})a = u, \quad (38)$$

where M_0, C_0, D_0 represent known parts of the model and $Y_1(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})a = M_1(\theta,a)\ddot{\theta} + C_1(\theta,\dot{\theta},a)\dot{\theta} + D_1(\theta,a)$ represents unknown parts.

Equations of the system (38) with a closed-loop of the feedback (36) can be expressed as

$$\begin{aligned} &M_0\left(\ddot{\theta} - \ddot{\theta}_{ref}\right) + C_0\left(\dot{\theta} - \dot{\theta}_{ref}\right) + D_0 \;+ \\ &+\underbrace{Y_1(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})}_{Y_1} a - \underbrace{Y_1(\theta,\dot{\theta},\dot{\theta}_{ref},\ddot{\theta}_{ref})}_{Y_{1r}}\hat{a} \;+ \\ &+K_d\dot{e}_\theta \;=\; 0, \end{aligned} \quad (39)$$

which after transformation gives the following expression

$$M_0\ddot{e}_\theta + C_0\dot{e}_\theta + (Y_1 a - Y_{1r}a) + (Y_{1r}a - Y_{1r}\hat{a}) + K_d\dot{e}_\theta = 0. \quad (40)$$

Then, considering that

$$Y_1 a = M_1(\theta,a)\ddot{\theta} + C_1(\theta,\dot{\theta},a)\dot{\theta} + D_1(\theta,a)$$

and

$$Y_{1r}a = M_1(\theta,a)\ddot{\theta}_{ref} + C_1(\theta,\dot{\theta},a)\dot{\theta}_{ref} + D_1(\theta,a)$$

the equation (40) can be converted to the form

$$\begin{aligned} &M_0\ddot{e}_\theta + C_0\dot{e}_\theta + M_1(\theta,a)\ddot{\theta} + C_1(\theta,\dot{\theta},a)\dot{\theta} + D_1(\theta,a) \\ &-\left(M_1(\theta,a)\ddot{\theta}_{ref} + C_1(\theta,\dot{\theta},a)\dot{\theta}_{ref} + D_1(\theta,a)\right) \\ &\qquad +Y_{1r}\,(a - \hat{a}) + K_d\dot{e}_\theta = 0. \end{aligned} \quad (41)$$

Assuming that $\tilde{a} = \hat{a} - a$ is an error of parameter estimation, and making the necessary transformations, the equation (41) can be written as

$$(M_0 + M_1)\,\ddot{e}_\theta + (C_0 + C_1)\,\dot{e}_\theta + K_d\dot{e}_\theta = Y_{1r}\tilde{a}, \quad (42)$$

which, taking into account the dependence $M_0 + M_1 = M(\theta,a) = M(\theta)$ and $C_0 + C_1 = C(\theta,\dot{\theta},a) = C(\theta,\dot{\theta})$, leads to the equation

$$M(\theta)\ddot{e}_\theta + C(\theta,\dot{\theta})\dot{e}_\theta + K_d\dot{e}_\theta = Y_{1r}\tilde{a}. \quad (43)$$

For a system with a closed feedback loop (29), (43), the Lyapunov-like function of the form was proposed

$$V_3(e_\xi,\dot{e}_\theta,\tilde{a}) = \frac{1}{2}\dot{e}_\theta^T M(\theta)\dot{e}_\theta + \frac{1}{2}\tilde{a}^T\Gamma^{-1}\tilde{a} + V_1(e_\xi), \quad (44)$$

where the first term is the same as in the case of the non-adaptive version of the algorithm (34), and $V_1(e_\xi)$ given by the equation (30) is the Lyapunov function for the kinematic controller (1st stage of the cascade), performing path following task.

The time derivative of the proposed Lyapunov function calculated along the trajectory of the system (43) is equal to

$$\begin{aligned} \dot{V}_3 \;&=\; \dot{V}_1 + \dot{e}_\theta^T M(\theta)\ddot{e}_\theta + \frac{1}{2}\dot{e}_\theta^T\dot{M}(\theta)\dot{e}_\theta + \tilde{a}^T\Gamma^{-1}\dot{\tilde{a}} \\ &=\; \dot{V}_1 + \dot{e}_\theta^T\left(Y_r\tilde{a} - C(\theta,\dot{\theta})\dot{e}_\theta - K_d\dot{e}_\theta\right) \\ &\quad +\frac{1}{2}\dot{e}_\theta^T\dot{M}(\theta)\dot{e}_\theta + \tilde{a}^T\Gamma^{-1}\dot{\tilde{a}}. \end{aligned} \quad (45)$$

The derivative of the estimation error for $\tilde{a}$ is $\dot{\tilde{a}} = \dot{\hat{a}} - \dot{a}$, which, assuming a is constant in time leads to the equality $\dot{\tilde{a}} = \dot{\hat{a}}$. The estimated parameter $\hat{a}$ is calculated according to the adaptation law given by the equation (37).

The derivative of the Lyapunov function (53) can then be rearranged to the following form

$$\begin{aligned} \dot{V}_3 \;&=\; \dot{V}_1 + \dot{e}_\theta^T\left(Y_{1r}\tilde{a} - C(\theta,\dot{\theta})\dot{e}_\theta - K_d\dot{e}_\theta\right) \\ &\quad +\frac{1}{2}\dot{e}_\theta^T\dot{M}(\theta)\dot{e}_\theta + \tilde{a}^T\Gamma^{-1}\left(-\Gamma Y_{1r}^T\dot{e}_\theta\right) \\ &=\; \dot{V}_1 + \dot{e}_\theta^T Y_{1r}\tilde{a} - \dot{e}_\theta^T K_d\dot{e}_\theta - \tilde{a}^T Y_{1r}^T\dot{e}_\theta. \end{aligned} \quad (46)$$

The fourth term is a scalar, so it can be transposed to the same result:

$$\begin{aligned} \dot{V}_3 \;&=\; \dot{V}_1 + \dot{e}_\theta^T Y_{1r}\tilde{a} - \dot{e}_\theta^T K_d\dot{e}_\theta - \left(\tilde{a}^T Y_{1r}^T\dot{e}_\theta\right)^T \\ &=\; \dot{V}_1 + \dot{e}_\theta^T Y_{1r}\tilde{a} - \dot{e}_\theta^T K_d\dot{e}_\theta - \dot{e}_\theta^T Y_{1r}\tilde{a} \\ &=\; -\dot{e}_\theta K_d\dot{e}_\theta - e_\xi^T K_p e_\xi \le 0. \end{aligned} \quad (47)$$

As you can see, the derivative of the Lyapunov-like function is negative semidefinite. From LaSalle's invariance principle it follows that $(\dot{e}_\theta, e_\xi) = (0,0)$ is an invariant set to which the trajectories of a system with a closed feedback loop converge asymptotically.

This ends the proof of the convergence of the adaptive backstepping integration algorithm for both stages of the cascade.

It is worth mentioning that if all parameters are unknown (full parameterization of the model), it is enough to assume $M_0(\theta) = C_0(\theta,\dot{\theta}) = D_0(\theta) = 0$ in the adaptive dynamic control algorithm (36).

8. Robust Dynamic Controller – Structural and Parametric Uncertainty of the Manipulator's Dynamics

As mentioned earlier in the article, in addition to parametric uncertainty, there may also occur structural uncertainty in the dynamics model. This situation may arise when some interaction has not been taken into account in the equations of dynamics. In practice, this means that the dynamic model does not take into account some interaction that occurs in the real system. Often, structural uncertainty also appears in a system that is not fully identified. This corresponds to the dynamics model (4), which includes not only structural but also parametric uncertainty

$$\begin{aligned} M(\theta,a)\ddot{\theta} + C(\theta,\dot{\theta},a)\dot{\theta} + D(\theta,a) + \delta &= \\ M_0(\theta)\ddot{\theta} + C_0(\dot{\theta},\theta)\dot{\theta} + D_0(\theta) + Y_2(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})a + \delta &= \\ u. \quad (48) \end{aligned}$$

For the above model of dynamics and for fulfilled Assumption 1, let us propose a robust dynamic control law as follows

$$\begin{aligned} u = {} & M_0(\theta)\ddot{\theta}_{ref} + C_0(\theta,\dot{\theta})\dot{\theta}_{ref} + D_0(\theta) + \\ & + Y_2(\theta,\dot{\theta},\dot{\theta}_{ref},\ddot{\theta}_{ref})\hat{a}(t) - K_d\dot{e}_\theta - K_s \mathrm{sign}\dot{e}_\theta, \quad (49) \end{aligned}$$

where $K_s > 0 \in R^+$ is the regulation parameter of the additional switching regulator.

If there is also parametric uncertainty in the model ($Y_2 \neq 0$), then the law of estimating the unknown parameters will be given by the same formula, as in the adaptive case, i.e., (37).

8.1. Proof of the Convergence

In the case of structural and parametric uncertainty (partial parametrization of the model), the dynamics equation of the manipulator takes the form

$$M_0(\theta)\ddot{\theta} + C_0(\dot{\theta},\theta)\dot{\theta} + D_0(\theta) + Y_2(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})a + \delta = u, \quad (50)$$

where M_0, C_0, D_0 represent known parts of model, $Y_2(\theta,\dot{\theta},\dot{\theta},\ddot{\theta})a = M_1(\theta,a)\ddot{\theta} + C_1(\theta,\dot{\theta},a)\dot{\theta} + D_1(\theta,a)$ represents parts of the model with parametric uncertainty and $\delta \in R^n$ represents the forces not included in the model which occur in real conditions.

It is worth it to mention that the regression matrix Y_2 – although it depends on unknown parameters a – differs from matrix Y_1, because Y_1 should include parameters derived from all forces, while Y_2 contains parameters derived only from forces included in the model, without δ.

After transforming the equations, similar to Section 7.1, the equations of the system (50) with a closed feedback loop (49) are obtained as below

$$M(\theta)\ddot{e}_\theta + C(\theta,\dot{\theta})\dot{e}_\theta + K_d\dot{e}_\theta + K_s\mathrm{sign}\dot{e}_\theta + \delta = Y_r\tilde{a}. \quad (51)$$

For a system with a closed feedback loop (29), (51), the Lyapunov-like function of the form was proposed

$$V_4(e_\xi,\dot{e}_\theta,\tilde{a}) = \frac{1}{2}\dot{e}_\theta^T M(\theta)\dot{e}_\theta + \frac{1}{2}\tilde{a}^T\Gamma^{-1}\tilde{a} + V_1(e_\xi), \quad (52)$$

where the first term is the same as in the case of the non-adaptive version of the algorithm (34), and $V_1(e_\xi)$ given by the equation (30) is the Lyapunov-like function for the kinematic controller following the path.

The time derivative of $\dot{V}_4$ calculated along trajectories of the closed-loop system (29,37,51) is equal to

$$\begin{aligned} \dot{V}_4 &= \dot{V}_1 + \dot{e}_\theta^T M(\theta)\ddot{e}_\theta + \frac{1}{2}\dot{e}_\theta^T\dot{M}(\theta)\dot{e}_\theta + \tilde{a}^T\Gamma^{-1}\dot{\tilde{a}} \\ &= \dot{V}_1 + \dot{e}_\theta^T\left(Y_{2r}\tilde{a} - C(\theta,\dot{\theta})\dot{e}_\theta - K_d\dot{e}_\theta - K_s\mathrm{sign}\dot{e}_\theta\right. \\ &\quad \left.-\delta\right) + \frac{1}{2}\dot{e}_\theta^T\dot{M}(\theta)\dot{e}_\theta + \tilde{a}^T\Gamma^{-1}\cdot\left(-\Gamma Y_{2r}^T\dot{e}_\theta\right). \quad (53) \end{aligned}$$

Using skew-symmetry between the inertia matrix and the Coriolis matrix, i.e.,

$$\dot{M}(\theta) = C(\theta,\dot{\theta}) + C^T(\theta,\dot{\theta}),$$

the time derivative of the Lyapunov-like function can be simplified as follows

$$\begin{aligned} \dot{V}_4 = {} & \dot{V}_1 + \dot{e}_\theta^T Y_{2r}\tilde{a} - \dot{e}_\theta^T K_d\dot{e}_\theta - \dot{e}_\theta^T\delta \\ & -\dot{e}_\theta^T K_s\mathrm{sign}\dot{e}_\theta - \tilde{a}^T Y_{2r}^T\dot{e}_\theta. \quad (54) \end{aligned}$$

Second and last term on the right side of the above the equation can be reduced; therefore, we can write

$$\begin{aligned} \dot{V}_4 &= \dot{V}_1 - \dot{e}_\theta^T K_d\dot{e}_\theta - \dot{e}_\theta^T\delta - \dot{e}_\theta^T K_s\mathrm{sign}\dot{e}_\theta \\ &= \dot{V}_1 - \dot{e}_\theta^T K_d\dot{e}_\theta - \dot{e}_\theta^T\mathrm{sign}\dot{e}_\theta\left(K_s + \delta\mathrm{sign}\dot{e}_\theta\right) \\ &= \dot{V}_1 - \dot{e}_\theta^T K_d\dot{e}_\theta - \mid\dot{e}_\theta\mid\left(K_s + \delta\mathrm{sign}\dot{e}_\theta\right). \end{aligned}$$

If

$$K_s \geq \parallel A \parallel + \varepsilon, \qquad \varepsilon \in R^+,$$

where $\parallel A \parallel$ is the limit from above of the norm of unknown force acting on the dynamical object (structural uncertainty of the dynamics), due to Assumption 1, then the following evaluation is true

$$\begin{aligned} \dot{V}_4 &= \dot{V}_1 - \dot{e}_\theta^T K_d\dot{e}_\theta - \varepsilon\mid\dot{e}_\theta\mid \\ &= -e_\xi^T K_p e_\xi - \dot{e}_\theta^T K_d\dot{e}_\theta - \varepsilon\mid\dot{e}_\theta\mid \\ &= -W(e_\xi,\dot{e}_\theta) \leq 0. \quad (55) \end{aligned}$$

By the LaSalle's invariance principle, it can be concluded that

$$W(e_\xi,\dot{e}_\theta) = 0$$

defines an invariant set, to which the trajectories of the closed-loop system converge asymptotically. It means that the invariant set is equal to the asymptotic equilibrium point

$$(e_\xi, \dot{e}_\theta) = (0, 0).$$

This ends the proof.

9. Simulation Study

Simulations were run with the MATLAB package and the SIMULINK toolbox. The object of the simulations was the RTR manipulator with three degrees of freedom, presented in Figure 3.

Links of the RTR manipulator have been modelled as homogeneous sticks with a length equal to $l_2 = 0.9$ m and $l_3 = 1$ m and masses $m_2 = 20$ kg and $m_3 = 20$ kg. The dynamics of the RTR manipulator are given by (1) with elements equal to:

- inertia matrix

$$M(\theta) = \begin{bmatrix} M_{11} & 0 & 0 \\ 0 & M_{22} & M_{23} \\ 0 & M_{23} & M_{33} \end{bmatrix}, \qquad (56)$$

$$\begin{aligned} M_{11} &= \tfrac{1}{3}m_2 l_2^2 + m_3(l_2^2 + \tfrac{1}{3}l_3^2\cos^2\theta_3 + l_2 l_3\cos\theta_3), \\ M_{22} &= m_2 + m_3, \\ M_{23} &= \tfrac{1}{2}m_3 l_2 l_3 \cos\theta_3, \\ M_{33} &= \tfrac{1}{3}m_3 l_3^2, \end{aligned}$$

- matrix of Coriolis and centrifugal forces

$$C(\theta, \dot{\theta}) = \begin{bmatrix} C_{11} & 0 & C_{13} \\ 0 & 0 & C_{23} \\ C_{31} & 0 & 0 \end{bmatrix} \qquad (57)$$

$$\begin{aligned} C_{11} &= \dot{\theta}_3(-\tfrac{1}{2}m_3 l_2 l_3\sin\theta_3 - \tfrac{1}{3}m_3 l_3^2 \sin\theta_3\cos\theta_3), \\ C_{13} &= -\dot{\theta}_1(\tfrac{1}{2}m_3 l_2 l_3\sin\theta_3 + \tfrac{1}{3}m_3 l_3^2\sin\theta_3\cos\theta_3), \end{aligned}$$

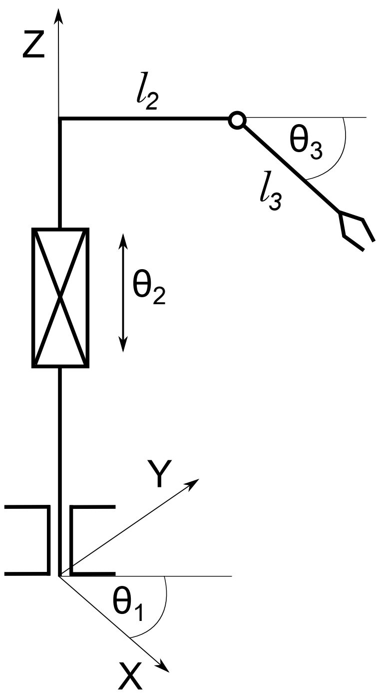

Figure 3. Manipulator RTR – the object of simulation [6]

$$\begin{aligned} C_{23} &= -\tfrac{1}{2}\dot{\theta}_3 m_3 l_2 l_3 \sin\theta_3, \\ C_{31} &= \dot{\theta}_1(\tfrac{1}{2}m_3 l_2 l_3\sin\theta_3 + \tfrac{1}{3}m_3 l_3^2\sin\theta_3\cos\theta_3), \end{aligned}$$

- gravity vector

$$D(\theta) = \begin{pmatrix} 0 \\ (m_2 + m_3)g \\ \tfrac{1}{2}g m_3 l_3 \cos\theta_3 \end{pmatrix}. \qquad (58)$$

The Cartesian position of the end-effector for the given manipulator can be expressed as

$$p = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos\theta_1(l_3\cos\theta_3 + l_2) \\ \sin\theta_1(l_3\cos\theta_3 + l_2) \\ l_3\sin\theta_3 + \theta_2 \end{pmatrix},$$

then the Jacobi matrix has a form

$$J(\theta) = \begin{bmatrix} -\sin\theta_1(l_3\cos\theta_3 + l_2) & 0 & -\cos\theta_1\sin\theta_3 l_3 \\ \cos\theta_1(l_3\cos\theta_3 + l_2) & 0 & -\sin\theta_1\sin\theta_3 l_3 \\ 0 & 1 & \cos\theta_3 l_3 \end{bmatrix}. \qquad (59)$$

The goal of the simulations was to investigate a behavior of this rigid fixed-base manipulator with parametric uncertainty using the controllers (27) and (37) proposed in the paper. The simulation was conducted for linear and square path parameterizations s_d for the assumption that one or two parameters of dynamics are unknown.

A screw curve has been chosen as a desired path:

$$\begin{aligned} r(s) &= (r_1(s), r_2(s), r_3(s))^T \\ &= \left(\cos\frac{s}{\sqrt{2}}, \sin\frac{s}{\sqrt{2}}, \frac{s}{\sqrt{2}}\right)^T. \end{aligned} \qquad (60)$$

Vectors T, N, and B have been selected as below

$$T(s) = \frac{1}{\sqrt{2}}\begin{pmatrix} -\sin\frac{s}{\sqrt{2}} \\ \cos\frac{s}{\sqrt{2}} \\ 1 \end{pmatrix}, \quad N(s) = \begin{pmatrix} -\cos\frac{s}{\sqrt{2}} \\ -\sin\frac{s}{\sqrt{2}} \\ 0 \end{pmatrix},$$

$$B(s) = \frac{1}{\sqrt{2}}\begin{pmatrix} \sin\frac{s}{\sqrt{2}} \\ -\cos\frac{s}{\sqrt{2}} \\ 1 \end{pmatrix},$$

and path parameters as $c(s) = \frac{1}{2}$, $\tau(s) = \frac{1}{2}$.

9.1. Linear Time Parametrization

For the path following linear time dependency $s_d(t)$ was chosen

$$\xi_d(t) = (s_d, q_{2d}, q_{3d})^T(t) = \left(\frac{t}{10}, 0, 0\right)^T. \qquad (61)$$

Due to the fact that, for control law (27), matrix $J(\theta)^{-1}$ is required, we assume that it is possible to avoid all singularities in robotic joint space if the manipulator can realize motion from initial configuration to desired task without the necessity to pass through singular configuration. Such a case is presented in this simulation study.

Simulations have been done for cases where one or two parameters of the manipulator are unknown. As a consequence, it is necessary to rewrite the model of dynamics as it was mentioned in Section 2.2.

One unknown parameter Assuming that among the parameters of the RTR manipulator the unknown parameter is

$$a_5 = \frac{1}{2} g m_3 l_3$$

from gravity vector D, the dynamics of the manipulator can be represented as

$$M_0(\theta)\ddot{\theta} + C_0(\theta,\dot{\theta})\dot{\theta} + D_0(\theta) + D_1(a_5) = u, \quad (62)$$

where the known part $D_0(\theta)$ and vector with unknown parameter $D_1(a_5)$ are

$$D_0(\theta) = \begin{pmatrix} 0 \\ (m_2+m_3)g \\ 0 \end{pmatrix}, \quad D_1(a_1) = \begin{pmatrix} 0 \\ 0 \\ a_5 \cos\theta_3 \end{pmatrix}.$$

It is easy to see that $M_0(\theta)$ and $C_0(\theta,\dot{\theta})$ contain only known parameters, so $M_0(\theta) = M(\theta)$ and $C_0(\theta,\dot{\theta}) = C(\theta,\dot{\theta})$, where $M(\theta)$ and $C(\theta,\dot{\theta})$ are given by (56) and (57). The dynamic model can be rewritten as follows

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + D_0(\theta) + Y_1(\theta)a_5 = u, \quad (63)$$

where

$$Y_1(\theta) = \begin{pmatrix} 0 \\ 0 \\ \cos\theta_3 \end{pmatrix} \quad (64)$$

is the regression matrix.

An adaptive version of the algorithm has the form

$$\begin{aligned} u \;=\;& M_0(\theta)\ddot{\theta}_{ref} + C_0(\theta,\dot{\theta})\dot{\theta}_{ref} + D_0(\theta) + \\ & + Y_1(\theta)\cdot \hat{a}_5(t) - K_d \dot{e}_\theta, \end{aligned} \quad (65)$$

and from (37) the estimated parameter value can be calculated from the adaptation law

$$\dot{\tilde{a}}_5(t) = \dot{\hat{a}}_5(t) = -\Gamma Y_1^T \dot{e}_\theta = -\Gamma \cos\theta_3 \dot{e}_{3\theta}. \quad (66)$$

The definition of the path with linear time dependency is given by (71). During the simulation, the following control parameters have been chosen: $K_p = 0.05$, $K_d = 100$, and adaptation gain $\Gamma = 800$.

Tracking of the desired path for the RTR manipulator by linear time parametrization has been presented in Figure 4a. Error of curvilinear distance $e_s = s - s_d$ has been plotted in Figure 4b. In turn, tracking errors of Cartesian coordinates in a normal plane have been presented in Figures 4c, d.

In turn, in Figure 5 a vector of parameter errors $\tilde{a}(t) = \hat{a}(t) - a$ has been presented.

From plots in Figure 4, it can be concluded that path tracking with linear time parametrization is realized properly and tracking errors converge to zero. Moreover, real curvilinear parametrization $s(t)$ tends to the desired function $s_d(t)$. Furthermore, distance tracking errors e_2 and e_3 have only positive values. It means that distance $(p - r)$ is positive and does not change sign during the regulation process. In other words, the matrix P is non-singular, and path parametrization using orthogonal projection on the curve is valid.

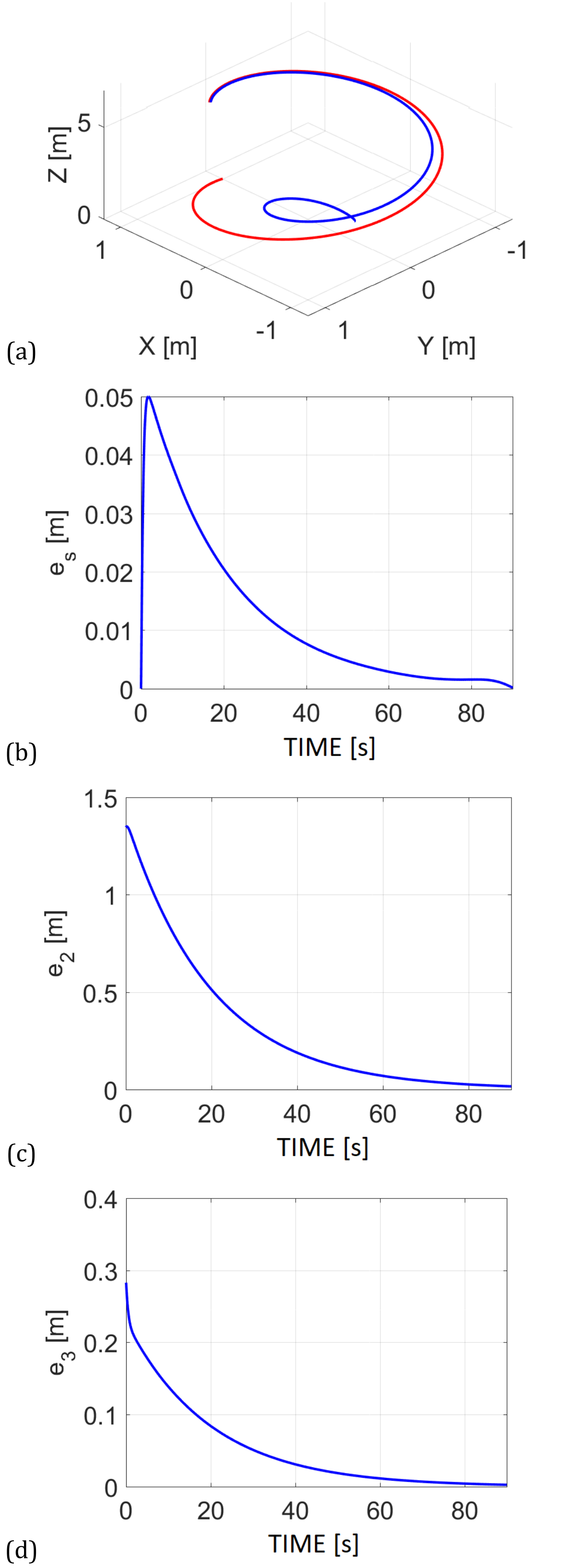

Figure 4. Linear time parametrization for one unknown parameter: (a) – the trajectory of the manipulator, (b) – curvilinear error e_s, (c) – distance error $e_2 = q_2 - q_{2d}$, (d) – distance error $e_3 = q_3 - q_{3d}$

Plot 5 has shown that estimate errors of unknown parameter converge to 0 during the regulation process. However, the convergence of estimation

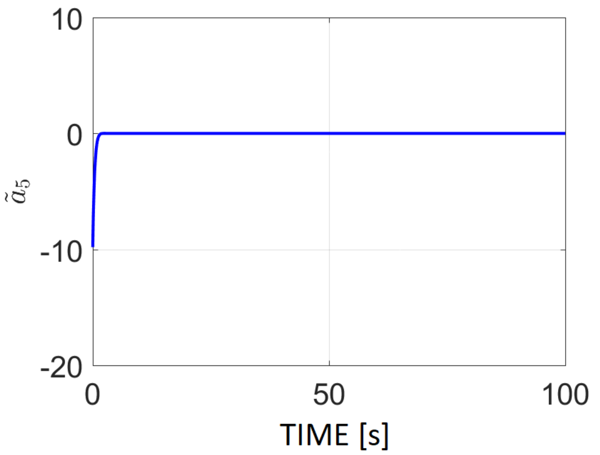

Figure 5. Linear time parametrization – estimate error $\tilde{a}_5 = \hat{a}_5(t) - a_5$

error to zero is not necessary to ensure correct work of the adaptive algorithm.

Two unknown parameters Let's assume that among the parameters of the RTR manipulator the unknown parameters are a_4 and a_5, selected as follows

$$a = \begin{pmatrix} a_4 \\ a_5 \end{pmatrix} = \begin{pmatrix} m_2 + m_3 \\ \frac{1}{2} g m_3 l_3 \end{pmatrix}, \quad a \in R^2. \tag{67}$$

Parameter a_5 occurs only in the gravity vector, while parameter a_4 appears in gravity vector and inertia matrix. Then, the manipulator dynamics model can be presented in the following way

$$(M_0(\theta) + M_1(\theta))\,\ddot{\theta} + C_0(\dot{\theta}, \theta)\dot{\theta} + D_0(\theta) + D_1(\theta, a) = u, \tag{68}$$

where M_1 and D_1 include unknown parameters, while $D_0(\theta) = 0$. The dynamic model of manipulator can be rewritten as follows

$$M_0(\theta)\ddot{\theta} + C_0(\theta, \dot{\theta})\dot{\theta} + Y_1(\theta, \ddot{\theta})a = u, \tag{69}$$

where $Y_1(\theta, \ddot{\theta})$ is regression matrix of unknown parameters a_4 and a_5

$$Y_1 = \begin{bmatrix} 0 & 0 \\ \ddot{\theta}_2 + g & 0 \\ 0 & \cos\theta_3 \end{bmatrix}. \tag{70}$$

The same control parameters as for one parameter have been used. Tracking of the desired path for the RTR manipulator by linear time parametrization given by (71) has been presented in Figure 6a. The error of curvilinear distance $e_s = s - s_d$ has been plotted in Figure 6b. As a result, tracking errors of Cartesian coordinates in the normal plane have been presented in Figure 6c, d. In Figure 7a, b, two vectors of parameter errors $\tilde{a}(t) = \hat{a}(t) - a$ have been presented.

From plots in Figure 6, it can be concluded that path tracking with linear time parametrization for two unknown parameters of the dynamic model is realized properly and tracking errors converge to zero.

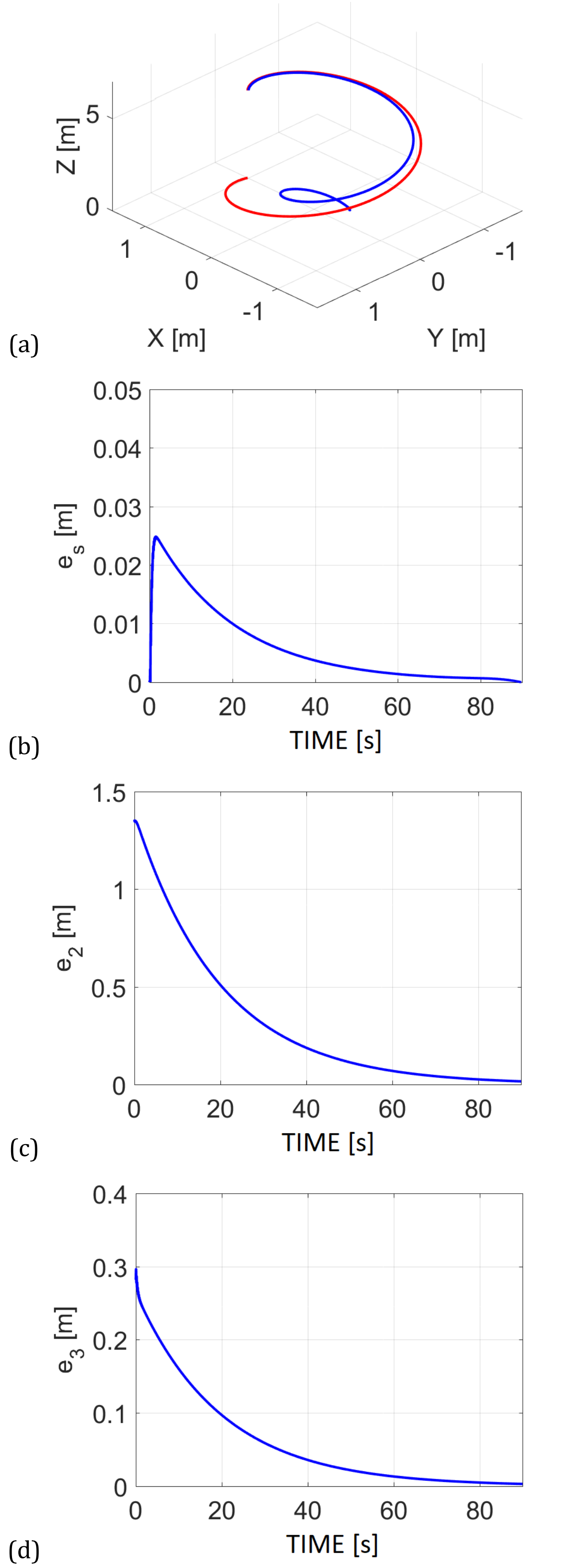

Figure 6. Linear time parametrization for two unknown parameters: (a) – the trajectory of the manipulator, (b) – curvilinear error e_s, (c) – distance error $e_2 = q_2 - q_{2d}$ and (d) – distance error $e_3 = q_3 - q_{3d}$

Moreover, real curvilinear parametrization $s(t)$ tends to the desired function $s_d(t)$. As for one unknown parameter, distance tracking errors e_2 and e_3 have only positive values.

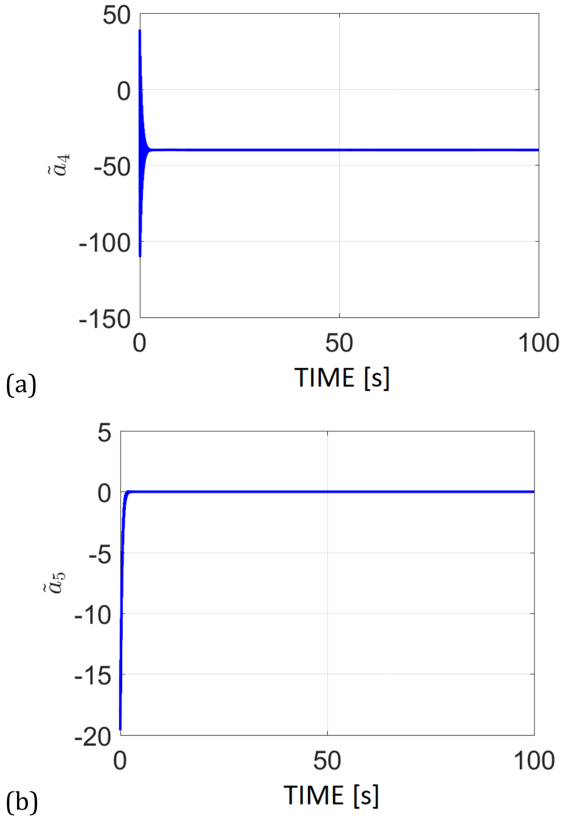

Figure 7. Linear time parametrization: (a) – estimate error $\tilde{a}_4 = \hat{a}_4 - a_4$ and (b) – estimate error $\tilde{a}_5 = \hat{a}_5 - a_5$

Plots 7a, b have shown that estimate errors of unknown parameters converge to zero 0 only for one of the unknown parameters. However, even if estimate value of the parameter is not equal to real value, the adaptive algorithm works correctly.

9.2. Square Time Parametrization

In the case of the two unknown parameters in the dynamics and the square parameterized path

$$\xi_d(t) = (s_d, q_{2d}, q_{3d})^T(t) = \left(0.1t - 0.0001t^2, 0, 0\right)^T, \tag{71}$$

simulation tests were carried out while maintaining the values of the control parameters.

The obtained results are shown in the Figures 8–10.

As can be seen in Figure 10, again the estimation errors of unknown parameters, and in particular $\tilde{a}_4$, did not converge to zero. This means that during the path following process, the realized trajectory did not meet the condition of persistent excitation.

10. Conclusion

In the paper, the general solution to the path tracking problem in three-dimensional space for the manipulator has been presented. To achieve the robot's description relative to the curve, Serret-Frenet parametrization with orthogonal projection on a given path has been used. Given equations are valid only if

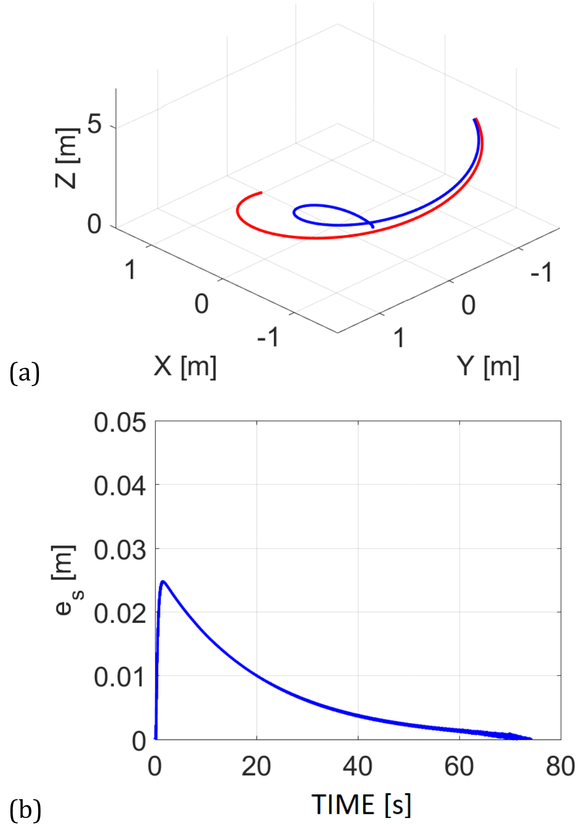

Figure 8. Square time parametrization: (a) – the trajectory of the manipulator, (b) – curvilinear error e_s

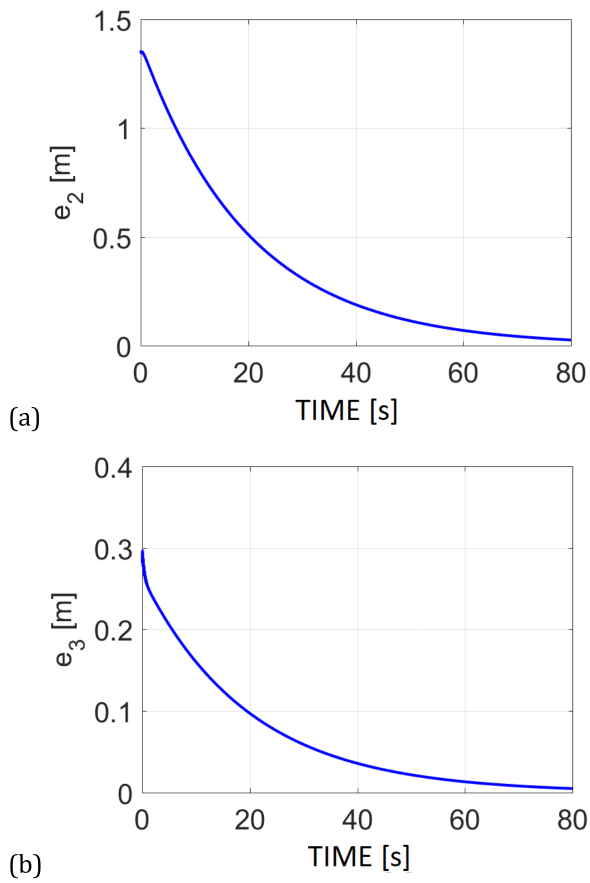

Figure 9. Square time parametrization: (a) – distance error $e_2 = q_2 - q_{2d}$, (b) – distance error $e_3 = q_3 - q_{3d}$

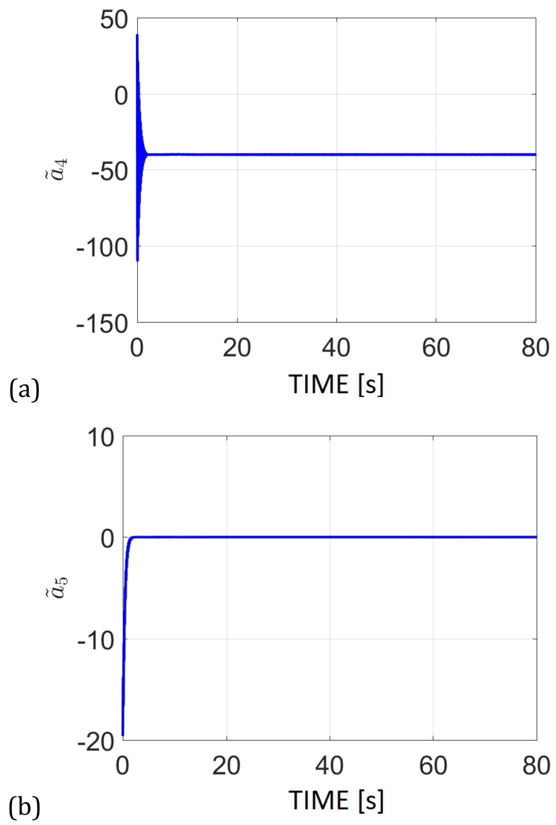

Figure 10. Square time parametrization: (a) – estimate error $\tilde{a}_4 = \hat{a}_4 - a_4$, (b) – estimate error $\tilde{a}_5 = \hat{a}_5 - a_5$

the distance between the object and the path, i.e., $p - r$, does not equal zero.

According to the fact, that the manipulator has been described by two groups of equations: expressions describing manipulator moving along the curve and dynamics equations, a cascaded control scheme has been proposed. The control scheme consists of two stages of cascade working simultaneously: the kinematic controller and the dynamic controller. The first one is responsible for solving the geometric problem of path tracking, and the second one makes it possible to realize velocities designed in the kinematic controller on the dynamic level.

The dynamic controllers, non-adaptive, adaptive, and robust algorithm, suitable for the manipulator with or without parametric uncertainty and structural uncertainty, were proposed. In simulation studies, we presented only the case of adaptive control with partial parameterization of the model. Comparing Figures 4 and 6, it can be observed that the errors of the distance from the path in the normal plane, i.e., e_2 and e_3, practically do not differ, even if only 1 parameter was unknown at first, and then two parameters. What is significantly different for both cases are parameter estimation errors, see Figures 5 and 7. In Figure 5, the parameter estimation error tends to 0, while in Figure 7 only one error tends to 0, and the other is very large. This agrees with the theory of adaptive control, which guarantees convergence to 0 for tracking errors. On the other hand, parameter estimation errors would all converge to 0 only if the condition of persistent excitation was met. Simulations have confirmed proper action of control algorithms introduced in the paper.

Future works will be focused on path following with other parametrisations, which are not limited by the assumption that distance to the path has to stay not equal to zero.

AUTHORS
Alicja Mazur* – Faculty of Electronics, Photonics and Microsystems, Wrocław University of Science and Technology, Janiszewskiego St. 11/17, 50-372 Wrocław, Poland, e-mail: alicja.mazur@pwr.edu.pl.
Mirela Kaczmarek – Department of Physics and Astronomy, Institute of Experimental Physics, University of Wrocław, Max Born Square 9, 50-204 Wrocław, Poland, e-mail: mirela.kaczmarek@uwr.edu.pl.

*Corresponding author

References

[1] F. Dyba, and A. Mazur. "Comparison of curvilinear parametrizations methods and avoidance of orthogonal singularities in the path following task," *Journal of Automation, Mobile Robotics and Intelligent Systems*, 2023, accepted for publication.

[2] M. Galicki. "Adaptive control of kinematically redundant manipulator along a prescribed geometric path," *Robot Motion and Control. Lecture Notes in Control and Information Sciences*, vol. 335, Springer, London, 2006, pp. 129–139.

[3] N. Hung, F. Rego, J. Quintas, J. Cruz, M. Jacinto, D. Souto, A. Potes, L. Sebastiao, and A. Pascoal. "A review of path following control strategies for autonomous robotic vehicles: Theory, simulations, and experiments," *Journal of Field Robotics*, vol. 40(3), 2023, pp. 747–779.

[4] M. Krstić, I. Kanellakopoulos, and P. V. Kokotovic. *Nonlinear and Adaptive Control Design*, John Wiley & Sons, Inc.: New York, USA, 1995.

[5] A. Mazur, and M. Kaczmarek. "Following 3D Paths by a manipulator – adaptive case," A. Mazur and C. Zieliński, editors, *Advances in Robotics (in Polish)*, vol. 1, 2022, pp. 45–58.

[6] A. Mazur. "Hybrid adaptive control laws solving a path following problem for nonholonomic mobile manipulators," *Int. Journal of Control*, vol. 77(15), 2004, pp. 1297–1306.

[7] A. Mazur, J. Płaskonka, and M. Kaczmarek. "Following 3D paths by a manipulator," *Archives of Control Sciences*, vol. 25(1), 2015, pp. 117–133; doi: 10.1515/acsc-2015-0008.

[8] A. Mazur, and D. Szakiel. "On path following control of nonholonomic mobile manipulators," *International Journal of Applied Mathematics and Computer Science*, vol. 19(4), 2009, pp. 561–574.

[9] A. Micaelli, and C. Samson. "Trajectory tracking for unicycle-type and two-steering-wheels mobile robots," *Technical Report No. 2097*, Sophia-Antipolis, 1993.

[10] J. Płaskonka. "Different kinematic path following controllers for a wheeled mobile robot of (2,0) type," *Journal of Intelligent & Robotic Systems*, vol. 77, 2013, pp. 481–498; doi: 10.1007/s10846-013-9879-6.

[11] D. Soetanto, L. Lapierre, and A. Pascoal. "Adaptive, non-singular path-following control of dynamic wheeled robots," *Proceedings of the IEEE Conference on Decision and Control, Maui, Hawaii, USA*, 2003, pp. 1765–1770; doi: 10.1109/CDC.2003.1272868.

[12] K. Tchoń, A. Mazur, I. Dulęba, R. Hossa, and R. Muszyński. *Manipulators and Mobile Robots: Models, Motion Planning, Control (in Polish)*, Academic Publishing House PLJ: Warsaw, 2000.

[13] V. Utkin, J. Guldner, and J. Shi. *Sliding Mode Control in Electro-mechanical Systems*, CRC Press, 1999.