

# COMPARISON OF CURVILINEAR PARAMETRIZATION METHODS AND AVOIDANCE OF ORTHOGONAL SINGULARITIES IN THE PATH FOLLOWING TASK

Submitted: 8<sup>th</sup> January 2023; accepted: 6<sup>th</sup> June 2023

Filip Dyba, Alicja Mazur

DOI: 10.14313/JAMRIS/3-2023/22

## Abstract:

*In this paper applications of curvilinear parametrizations (Serret–Frenet, Bishop) in the path following task have been considered. The parametrizations allow one to derive manipulator’s equations with respect to a path. The full mathematical model of the path following task involves two groups of equations, i.e., the dynamics of the manipulator and the equations obtained from the parametrization method, connected in the cascaded system.*

*Based on those relations two path following algorithms have been designed according to the backstepping integrator method (dedicated to the cascaded systems). Depending on the chosen parametrization method the algorithms differ in requirements and performance. In the paper an in-depth analysis comparing features of both considered methods has been presented.*

*The parametric description of a path requires projection of a robot on the path. In this article the orthogonal projection has been taken into account. It introduces a singularity in the robot description. We have proposed a new form of the orthogonal projection constraint which allows a robot to not only approach the path, but also move along it. This novelty design is an important enhancement of the algorithms used so far.*

*The problem of partially known dynamic parameters of a robot has also been addressed. In this paper, we have shown how to apply an adaptive controller to the path following task.*

*Theoretical considerations have been verified with a simulation study conducted for a holonomic stationary manipulator. Achieved results emphasized why it is strongly recommended to use the algorithm version with the orthogonal singularity outside the path. Moreover, the comparative analysis results may be used to select the best curvilinear parametrization method according to the considered task requirements.*

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

## 1. Introduction

The path following task is one of the basic robot control tasks [19]. This control problem has attracted a great deal of attention as it plays a crucial role in many important technological challenges nowadays, e.g., the usage of autonomous vehicles [29]. In the literature, different solutions for the path following control problem have been discussed many times considering different robots and path definitions. The approach using parametrizations in the path description was harnessed in various papers, for instance, for mobile robots [14, 22, 33], for holonomic fixed-base manipulators [9, 20], and mobile manipulators [18, 21]. The similar control problem was defined also for more complex robotic objects, such as autonomous underwater vehicles [7] and flying robots [17]. However, the vast majority of the mentioned papers consider only the two-dimensional case, which is not easily scalable to the three-dimensional space.

Apart from the approach using parametrizations, also a non-parametric description of a path has been taken into account. One of such methods is e.g. the level set method, which is a tool for numerical analysis of surfaces and shapes. The advantage of the level set model is the possibility to perform numerical computations involving curves and surfaces without having to parametrize those objects. Exemplary solutions were provided in [5, 13, 24–26].

In the presented paper only the parametric description of a path is taken into consideration. Let us define a path as a geometric curve which does not depend on time but is a purely geometrical description of a motion [32]. The curve is usually parametrized with the so-called curvilinear distance which may be interpreted as the length of a string laying perfectly on the path [23]. It has some important implications. Firstly, geometrical planning of the given path is simpler than planning a time-dependant trajectory. Secondly, no time regimes are imposed on a controlled object due to the lack of time dependency [28]. This feature is particularly vital in applications taking into account constraints on control torques or forces.

In the approach presented in the paper it is assumed that the real robot is controlled with respect to a certain reference object, which is also called

a virtual one [33]. The virtual object motion is modelled with the curve geometry by harnessing curvilinear parametrization formalism. The definition of this motion allows one to additionally reduce values of control generalized forces by proper choice of the manoeuvre time. It may be arbitrarily extended due to the fact that paths are independent of time.

Various methods of curvilinear parametrizations may be applied to solving this problem. The most frequently chosen method is the Serret–Frenet parametrization [8,31]. It has already been considered in many applications, e.g., for mobile platforms [22], mobile manipulators [18], and even for stationary holonomic manipulators to some extent [9]. However, it is not the only method which may be used for the considered problem. The Bishop parametrization [1] is an alternative possibility. Its application to the robot control has not been extensively analyzed so far, although the relatively parallel transport frames, which are based on the Bishop idea, have been considered in some modifications of the known control algorithms [11]. Furthermore, the robot description with respect to a path is dependent on the projection method. Each method has its own advantages and disadvantages, but all of them allow deriving description of a robot with respect to a moving reference frame.

In this article the orthogonal projection method is chosen as it minimizes the dimensionality of the control problem. It is combined with different methods of the curvilinear parametrization which are compared to each other. It is noteworthy that the problem in the three-dimensional space is taken into account as many solutions have been reduced to the planar case so far. The possibility of their application to the path following task for a holonomic manipulator is verified with a simulation study.

The article is organized as follows. In Section 2 equations of the considered parametrization methods are defined. Also, dependencies between them are presented. Equations describing the holonomic manipulator in general, including its description with respect to the given path, are provided in Section 3. Different curvilinear parametrizations are used in order to achieve the description. In addition, the analysis of the singularity resulting from the orthogonal projection is presented. In Section 4, the control task is defined. Furthermore, the deployed control algorithms are also presented. Results of numerical simulations are shown in Section 5. The considerations are summarized in Section 6.

This paper is an extension of the conference paper [6] and the following aspects are novel. Another method of the Bishop frame initialization has been investigated. Also, the analysis of local frames' behavior along the curve has been extended. In particular, their orientation changes have been taken into account. It is an essential part of the presented comparative analysis. Moreover, the problem of orthogonal parametrization singularities has been addressed. In the article a modification of the kinematic controller has been proposed. It allows avoiding the singularities

and lets the robot move along the path. Finally, application of an adaptive version of the dynamic controller to the designed control cascade has been verified. The achieved results have been compared with the non-adaptive controller, which was mainly considered in the previous work.

## 2. Curvilinear Parametrizations

The curvilinear parametrizations allow defining a local frame. Its motion along a curve fully describes the curve geometry [27]. In the following sections equations of different parametrization methods are presented.

### 2.1. Serret–Frenet Parametrization

The most common curvilinear parametrization method is the Serret–Frenet parametrization [8, 31]. The local frame consists of three vectors: tangential to the curve  $\mathbf{T}$ , normal to the curve  $\mathbf{N}$ , and binormal to the curve  $\mathbf{B}$ . They create an orthonormal basis in the  $\mathbb{R}^3$  space and span the Frenet trihedron, which is presented in Figure 1 [34]. The given vectors are defined with the following equations [34]

$$\mathbf{T}(s) = \frac{d\mathbf{r}(s)}{ds}, \quad (1a)$$

$$\mathbf{N}(s) = \frac{\frac{d\mathbf{T}(s)}{ds}}{\left\| \frac{d\mathbf{T}(s)}{ds} \right\|}, \quad (1b)$$

$$\mathbf{B}(s) = \mathbf{T}(s) \times \mathbf{N}(s), \quad (1c)$$

where  $\mathbf{r}(s)$  denotes coordinates of a point on a curve expressed in the inertial reference frame. In other words, it is an analytical expression describing a curve. The evolution along a curve of the defined frame is expressed with the following equation

$$\begin{bmatrix} \frac{d\mathbf{T}(s)}{ds} & \frac{d\mathbf{N}(s)}{ds} & \frac{d\mathbf{B}(s)}{ds} \end{bmatrix} = \begin{bmatrix} \mathbf{T}^T(s) \\ \mathbf{N}^T(s) \\ \mathbf{B}^T(s) \end{bmatrix}^T \begin{bmatrix} 0 & -\kappa(s) & 0 \\ \kappa(s) & 0 & -\tau(s) \\ 0 & \tau(s) & 0 \end{bmatrix}, \quad (2)$$

where the curvature  $\kappa(s)$  defines the swerve of a curve from a straight line, whereas the torsion  $\tau(s)$  expresses the swerve of a curve from a plane.

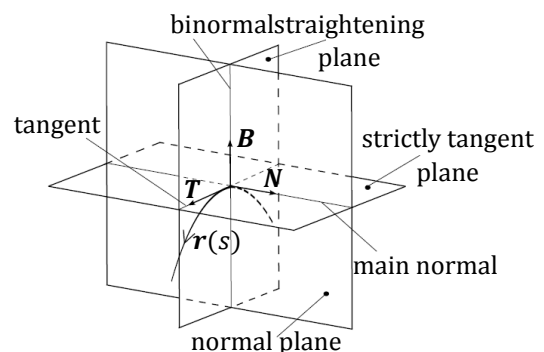


Figure 1. Frenet trihedron

Geometrical invariants of a curve may be formulated with equations [20]

$$\kappa(s) = \left\| \frac{d\mathbf{T}(s)}{ds} \right\|, \quad (3a)$$

$$\tau(s) = \frac{1}{\kappa^2(s)} \left\langle \frac{d\mathbf{r}(s)}{ds} \times \frac{d^2\mathbf{r}(s)}{ds^2}, \frac{d^3\mathbf{r}(s)}{ds^3} \right\rangle, \quad (3b)$$

where  $\langle \cdot, \cdot \rangle$  denotes a scalar product of vectors.

## 2.2. Bishop Parametrization

A different parametrization method was proposed by Bishop in [1]. In this approach the local reference frame is also created with three orthogonal vectors. Again, one of them is the tangential vector  $\mathbf{T}$ . However, the other two vectors,  $\mathbf{N}_1$  and  $\mathbf{N}_2$ , do not have as intuitive of a geometrical interpretation as vectors creating the Serret–Frenet frame described in Section 2.1, although they are also orthogonal to the tangential vector  $\mathbf{T}$ . They create relatively parallel vector fields. The Bishop frame defined in the initial state is transported along a curve preserving its uniqueness [16]. The frame evolution along a curve may be expressed with equations [1]

$$\begin{aligned} & \left[ \begin{array}{ccc} \frac{d\mathbf{T}(s)}{ds} & \frac{d\mathbf{N}_1(s)}{ds} & \frac{d\mathbf{N}_2(s)}{ds} \end{array} \right] = \\ & = \left[ \begin{array}{c} \mathbf{T}^T(s) \\ \mathbf{N}_1^T(s) \\ \mathbf{N}_2^T(s) \end{array} \right]^T \left[ \begin{array}{ccc} 0 & -k_1(s) & -k_2(s) \\ k_1(s) & 0 & 0 \\ k_2(s) & 0 & 0 \end{array} \right], \quad (4) \end{aligned}$$

where  $k_1(s)$  and  $k_2(s)$  are certain functions analogical to the curvature and the torsion of a curve, which fully determines the curve geometry. Their values in the following points of a curve may be calculated with the usage of the relations [16]

$$k_1(s) = \left\langle \frac{d\mathbf{T}(s)}{ds}, \mathbf{N}_1(s) \right\rangle, \quad (5a)$$

$$k_2(s) = \left\langle \frac{d\mathbf{T}(s)}{ds}, \mathbf{N}_2(s) \right\rangle. \quad (5b)$$

## 2.3. Relationships Between Parametrizations

Due to some differences in the definitions of the base vectors of the local frames associated with a curve, the presented parametrization methods have different properties. It is crucial that the Serret–Frenet frame is undefined in all points where the curvature is equal to zero. In particular, it concerns all straight lines, which may be a part of many common path following tasks. It results from the singularity which appears in the normal vector definition (1b). The key advantage of the Bishop frame is the fact that there are no constraints resulting from its definition. However, evolution of this frame is described with a more complex operation of vector relatively parallel transport along a curve [10]. Thus, the geometrical interpretation of the Bishop base vectors is less intuitive. It is also worth noticing that the Serret–Frenet frame requires that the parametrized curve be of at least  $C^3$  class. It results from the equation defining

the curve torsion (3b). The Bishop equations are less restrictive as they allow parametrizing curves of  $C^2$  class.

Despite all the differences, there is a strong connection between the geometrical invariants of the Bishop parametrization  $k_1(s)$ ,  $k_2(s)$  and the geometrical parameters – curvature  $\kappa(s)$  and torsion  $\tau(s)$ . For a curve described with a pair  $(k_1, k_2)$  polar coordinates may be defined. They are equal to  $(\kappa, \int \tau(s) ds)$  [1]. Hence, the direct relation which links the invariants of both parametrizations is defined with the following equation [16]

$$k_1(s) = \kappa(s) \cos \theta(s), \quad (6a)$$

$$k_2(s) = \kappa(s) \sin \theta(s), \quad (6b)$$

where  $\theta(s) = \int \tau(s) ds$ . Due to the integration constant, the Bishop frame may be defined in many ways in the initial state. However, once defined the local frame always evolves along a curve identically in the precisely described manner [30]. Furthermore, a transformation connecting the Serret–Frenet frame and the Bishop frame is defined. The tangential vector  $\mathbf{T}$  is the common part of both methods. The other vectors may be determined based on the following relations [1]

$$\mathbf{N}(s) = \cos \theta(s) \mathbf{N}_1(s) + \sin \theta(s) \mathbf{N}_2(s), \quad (7a)$$

$$\mathbf{B}(s) = -\sin \theta(s) \mathbf{N}_1(s) + \cos \theta(s) \mathbf{N}_2(s). \quad (7b)$$

The aforementioned equations allow one to combine both parametrization methods in order to benefit from their advantages simultaneously. An example of such a solution is the *beta* frame presented in [4].

## 3. Mathematical Model of a Manipulator

### 3.1. Holonomic Manipulator

The considered controlled object is a non-redundant stationary holonomic manipulator. The forward kinematics task, which defines the end-effector position  $\mathbf{p}$  in the base frame, which will be identified with the inertial frame, is defined as

$$\mathbf{p} = k(\mathbf{q}). \quad (8)$$

Hence, velocities in the Cartesian space depend on the velocities in the joint space according to the equation with the Jacobi matrix  $\mathbf{J}$  [34]

$$\dot{\mathbf{p}} = \frac{\partial k(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} = \mathbf{J}(\mathbf{q}) \dot{\mathbf{q}}. \quad (9)$$

The manipulator dynamics is derived with the usage of the Euler–Lagrange formalism and the stationary-action principle. Without loss of generality we may omit the dissipative effects in the dynamics equations as they are not crucial for the considerations presented in the paper. Moreover, direct-drive actuators are considered.

Firstly, let us assume that the dynamics model is fully known. The structure of the manipulator dynamics is described with the equation [34]

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) = \mathbf{u}, \quad (10)$$

where:

- $\mathbf{q} \in \mathbb{R}^n$  is the manipulator configuration which consists of the joint positions;
- $\mathbf{M} \in \mathbb{R}^{n \times n}$  is the inertia matrix;
- $\mathbf{C} \in \mathbb{R}^{n \times n}$  is the matrix of Coriolis and centrifugal forces;
- $\mathbf{D} \in \mathbb{R}^n$  is the vector of the generalized forces resulting from the gravity effects; and
- $\mathbf{u} \in \mathbb{R}^n$  is the vector of generalized control forces.

Secondly, let us consider the case when the dynamics are defined with parametric uncertainty. In other words, not all parameters standing before functions occurring in the dynamical equations are known. According to the theory of adaptive systems it has to be assumed that the unknown parameters are constant and the model is linearly dependent on them [12]. If there are some unknown parameters, the model (10) may be expressed with the usage of regression matrix  $\mathbf{Y}$

$$\begin{aligned} \mathbf{u} &= \mathbf{M}(\mathbf{q}, \mathbf{a})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}, \mathbf{a}) = \\ &= [\mathbf{M}_0(\mathbf{q}) + \mathbf{M}_a(\mathbf{q})]\ddot{\mathbf{q}} + [\mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{C}_a(\mathbf{q}, \dot{\mathbf{q}})]\dot{\mathbf{q}} + \\ &\quad + \mathbf{D}_0(\mathbf{q}) + \mathbf{D}_a(\mathbf{q}) = \\ &= \mathbf{M}_0(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}_0(\mathbf{q}) + \mathbf{Y}(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q})\mathbf{a} \end{aligned} \quad (11)$$

where  $\mathbf{a}$  is the vector of the unknown constant parameters, the elements with the subscript 0 denote the known parts of the model and the elements with  $a$  in the subscript correspond to the part of the model dependent on the unknown parameters.

In turn, the first argument of the regression matrix  $\mathbf{Y}$  gives the vector by which the inertia matrix is multiplied, the second component gives the vector by which the Coriolis matrix is multiplied, the third component defines the velocity occurring in the Coriolis matrix and the last component defines the trajectory along which the model is described.

### 3.2. Robot Equations with Respect to a Path

Let us define a rotation matrix  $\mathbf{S}(s)$  which defines the orientation of the local frame in a certain point of a curve defined by the curvilinear distance  $s$ . The base vectors resulting from the definitions of the curvilinear parametrizations are columns of such a matrix. Hence, the following relation holds for the Serret-Frenet frame

$$\mathbf{S}(s) = [\mathbf{T} \quad \mathbf{N} \quad \mathbf{B}]. \quad (12)$$

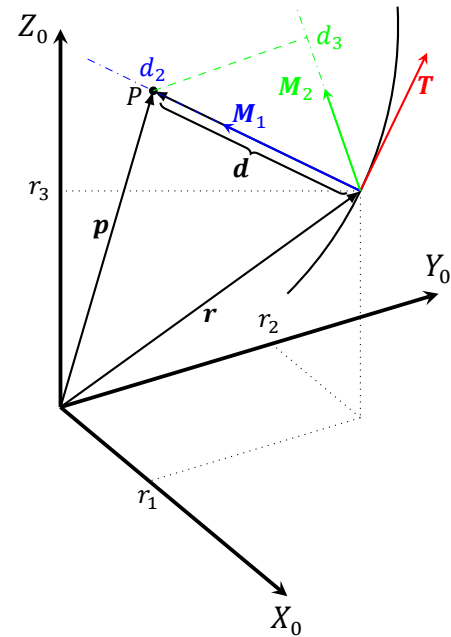
For the Bishop frame equation (12) takes the form

$$\mathbf{S}(s) = [\mathbf{T} \quad \mathbf{N}_1 \quad \mathbf{N}_2]. \quad (13)$$

The position of the end-effector may be then defined in the local frame associated with the given curve

$$\mathbf{d} = (d_1 \quad d_2 \quad d_3)^T = \mathbf{S}^T(\mathbf{p} - \mathbf{r}), \quad (14)$$

where  $\mathbf{r}$  is a point on a curve expressed in the inertial frame and  $\mathbf{p}$  is the end-effector position in the inertial



**Figure 2.** Visualization of the orthogonal projection of a robot guidance point on a curve

frame, which will be referred to as the robot guidance point. In Figure 2, a schematic view of the projection of a robot guidance point on a curve is visualized. This scheme is independent of the parametrization method as the normal vectors are denoted as  $\mathbf{M}_1$  and  $\mathbf{M}_2$ . The pair  $\{\mathbf{M}_1, \mathbf{M}_2\}$  should be understood as  $\{\mathbf{N}, \mathbf{B}\}$  for the Serret-Frenet parametrization and  $\{\mathbf{N}_1, \mathbf{N}_2\}$  for the Bishop parametrization.

The dynamics of the end-effector position with respect to the moving local frame may be derived by differentiating equation (14)

$$\dot{\mathbf{d}} = \mathbf{S}^T(\dot{\mathbf{p}} - \dot{\mathbf{r}}) + \dot{\mathbf{S}}^T(\mathbf{p} - \mathbf{r}). \quad (15)$$

Taking into account relations describing evolution of the local frame, i.e., equation (2) for the Serret-Frenet frame or equation (4) for the Bishop frame, the following form may be derived

$$\dot{\mathbf{d}} = \mathbf{S}^T(\dot{\mathbf{p}} - \dot{\mathbf{r}}) - \mathbf{s}\mathbf{W}\mathbf{d}, \quad (16)$$

where  $\mathbf{W}$  is a skew-symmetric matrix. Its elements are proper parameters describing the curve geometry. Hence, its definition results directly from equation (2) or equation (4).

The aspect of the utmost importance, which significantly influences the form of the constraints enforcing motion along the given path, is the method of projection of a robot guidance point on a curve. There are two projection methods which should be distinguished: the orthogonal projection and the non-orthogonal projection. The latter one does not impose any additional constraints and the reference object associated with the local frame may move freely along the curve. However, position errors in every dimension of the local frame need to be followed. Such an approach was considered in some papers, i.e., in [2, 33]. In turn, for the orthogonal projection it is assumed

that the manipulator end-effector is always located in the minimal distance from the path. Hence, the position of the robot in the tangent vector direction is always equal to zero. Vectors  $\mathbf{T}$  and  $(\mathbf{p} - \mathbf{r})$  must be orthogonal, which leads to the relation

$$\langle \mathbf{T}, \mathbf{p} - \mathbf{r} \rangle = 0. \quad (17)$$

Based on equation (17) the expression describing the curvilinear velocity of the local frame may be derived [20]

$$\dot{s} = -\frac{\langle \mathbf{T}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle}{\left\langle \frac{d\mathbf{T}}{ds}, \mathbf{p} - \mathbf{r} \right\rangle}. \quad (18)$$

Preservation of this condition allows one to decrease the dimensionality of the problem. However, as a consequence the virtual object associated with the local frame cannot move freely along the given path, but according to a strictly defined function. Moreover, equation (18) is a source of potential singularities due to the form of the denominator. It also causes the object description to be valid only locally.

**Serret–Frenet parametrization** Considering the Serret–Frenet definition of the local frame (2), equation (18) is transformed to the form

$$\dot{s} = -\frac{\langle \mathbf{T}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle}{\kappa \langle \mathbf{N}, \mathbf{p} - \mathbf{r} \rangle} = -\frac{\mathbf{T}^T}{\kappa \langle \mathbf{N}, \mathbf{p} - \mathbf{r} \rangle} \dot{\mathbf{p}} + \frac{\langle \mathbf{T}, \dot{\mathbf{r}} \rangle}{\kappa \langle \mathbf{N}, \mathbf{p} - \mathbf{r} \rangle} = \mathbf{P}_1 \dot{\mathbf{p}} + R_1. \quad (19)$$

Furthermore, velocities of a robot guidance point in the Serret–Frenet frame are equal to

$$\dot{d}_1 = \langle \mathbf{T}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle + \dot{s} \kappa \langle \mathbf{N}, \mathbf{p} - \mathbf{r} \rangle, \quad (20a)$$

$$\dot{d}_2 = \langle \mathbf{N}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle - \dot{s} \kappa \langle \mathbf{T}, \mathbf{p} - \mathbf{r} \rangle + \dot{\tau} \langle \mathbf{B}, \mathbf{p} - \mathbf{r} \rangle, \quad (20b)$$

$$\dot{d}_3 = \langle \mathbf{B}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle - \dot{s} \tau \langle \mathbf{N}, \mathbf{p} - \mathbf{r} \rangle. \quad (20c)$$

Taking into account equation (19), equation (20) has the form

$$\dot{d}_1 = 0, \quad (21a)$$

$$\dot{d}_2 = \left( \mathbf{N} - \frac{\tau \langle \mathbf{B}, \mathbf{p} - \mathbf{r} \rangle}{\kappa \langle \mathbf{N}, \mathbf{p} - \mathbf{r} \rangle} \mathbf{T} \right)^T \dot{\mathbf{p}} + \left( \mathbf{N} - \frac{\tau \langle \mathbf{B}, \mathbf{p} - \mathbf{r} \rangle}{\kappa \langle \mathbf{N}, \mathbf{p} - \mathbf{r} \rangle} \mathbf{T}, \dot{\mathbf{r}} \right) = \mathbf{P}_2 \dot{\mathbf{p}} + R_2, \quad (21b)$$

$$\dot{d}_3 = \left( \mathbf{B} + \frac{\tau}{\kappa} \mathbf{T} \right)^T \dot{\mathbf{p}} - \left( \mathbf{B} + \frac{\tau}{\kappa} \mathbf{T}, \dot{\mathbf{r}} \right) = \mathbf{P}_3 \dot{\mathbf{p}} + R_3. \quad (21c)$$

Thus, it may be concluded from equation (21a) that if the constraint (19) is preserved, the end-effector tip is always located in the normal plane spanned by the Frenet trihedron. As a result, the state of the manipulator with respect to the Serret–Frenet frame is defined as

$$\xi = (s \quad d_2 \quad d_3)^T. \quad (22)$$

Evolution of the state (22) is defined by equations (19), (21b) and (21c). It takes the following form

$$\dot{\xi} = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix} \dot{\mathbf{p}} + \begin{pmatrix} R_1 \\ R_2 \\ R_3 \end{pmatrix} = \mathbf{P} \dot{\mathbf{p}} + \mathbf{R}. \quad (23)$$

**Bishop parametrization** A similar procedure leads to the relations defining the robot guidance point velocities with respect to the Bishop frame. Taking into account equation (4), equation (18) may be rewritten as

$$\dot{s} = -\frac{\langle \mathbf{T}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle}{k_1 \langle \mathbf{N}_1, \mathbf{p} - \mathbf{r} \rangle + k_2 \langle \mathbf{N}_2, \mathbf{p} - \mathbf{r} \rangle}, \quad (24)$$

which may be extended to the form

$$\dot{s} = \frac{-\mathbf{T}^T}{k_1 \langle \mathbf{N}_1, \mathbf{p} - \mathbf{r} \rangle + k_2 \langle \mathbf{N}_2, \mathbf{p} - \mathbf{r} \rangle} \dot{\mathbf{p}} + \frac{\langle \mathbf{T}, \dot{\mathbf{r}} \rangle}{k_1 \langle \mathbf{N}_1, \mathbf{p} - \mathbf{r} \rangle + k_2 \langle \mathbf{N}_2, \mathbf{p} - \mathbf{r} \rangle} = \mathbf{P}_1 \dot{\mathbf{p}} + R_1. \quad (25)$$

Furthermore, equations defining the end-effector velocities in the Bishop frame are expressed with the following relations:

$$\dot{d}_1 = \langle \mathbf{T}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle + \dot{s} k_1 \langle \mathbf{N}_1, \mathbf{p} - \mathbf{r} \rangle + \dot{s} k_2 \langle \mathbf{N}_2, \mathbf{p} - \mathbf{r} \rangle, \quad (26a)$$

$$\dot{d}_2 = \langle \mathbf{N}_1, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle - \dot{s} k_1 \langle \mathbf{T}, \mathbf{p} - \mathbf{r} \rangle, \quad (26b)$$

$$\dot{d}_3 = \langle \mathbf{N}_2, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle - \dot{s} k_2 \langle \mathbf{T}, \mathbf{p} - \mathbf{r} \rangle. \quad (26c)$$

Considering equation (25) results in

$$\dot{d}_1 = 0, \quad (27a)$$

$$\dot{d}_2 = \mathbf{N}_1^T \dot{\mathbf{p}} - \langle \mathbf{N}_1, \dot{\mathbf{r}} \rangle = \mathbf{P}_2 \dot{\mathbf{p}} + R_2, \quad (27b)$$

$$\dot{d}_3 = \mathbf{N}_2^T \dot{\mathbf{p}} - \langle \mathbf{N}_2, \dot{\mathbf{r}} \rangle = \mathbf{P}_3 \dot{\mathbf{p}} + R_3. \quad (27c)$$

Similarly, as for the Serret–Frenet parametrization approach, the end-effector tip is located in the normal plane, spanned by the vectors  $\{\mathbf{N}_1, \mathbf{N}_2\}$  for the Bishop frame, if the constraint (25) is satisfied. It may be concluded directly from equation (27a). It is noteworthy that also the state in the Bishop frame may be described with equation (22). Hence, equations (25), (27b), and (27c) express the evolution of the robot in the Bishop frame

$$\dot{\xi} = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix} \dot{\mathbf{p}} + \begin{pmatrix} R_1 \\ R_2 \\ R_3 \end{pmatrix} = \mathbf{P} \dot{\mathbf{p}} + \mathbf{R}. \quad (28)$$

Application of the Bishop parametrization undoubtedly allows achieving the same structure of the equations, whereas expressions defining dynamics of values of  $d_2$  – (27b) – and  $d_3$  – (27c) – are much simpler. However, it needs to be emphasized that the denominator form in equation (24) is more complex than in equation (19). Nonetheless, for both cases it is impossible to move precisely

on the path as then  $\mathbf{p} - \mathbf{r} = 0$  holds. As a result, the orthogonal parametrization in the presented form allows only asymptotic path following—the given curve cannot be reached. Furthermore, the parametrized curve cannot be straight, because for straight lines  $\kappa(s) = k_1(s) = k_2(s) = 0$ . For both curvilinear parametrization methods those cases lead to a singularity resulting from the orthogonal projection assumptions.

### 3.3. Orthogonal Singularity Avoidance

As it has been shown in the previous subsection, the singularity of the orthogonal projection of a robot on the given path is located directly on the path. Due to increasing uncertainty of the problem, the previous solution may be considered only for approaching the path. Thus, a reformulation of equation (18) may be necessary. Based on the definition of the tangential vector of the local frame associated with the curve, given by equation (1a), the following relation may be observed

$$\langle \mathbf{T}, \dot{\mathbf{r}} \rangle = \left\langle \mathbf{T}, \frac{d\mathbf{r}}{ds} \dot{s} \right\rangle = \langle \mathbf{T}, \mathbf{T} \rangle \dot{s} = \dot{s}. \quad (29)$$

Introducing equation (29) into (18) leads to the equation

$$\dot{s} = -\frac{\langle \mathbf{T}, \dot{\mathbf{p}} \rangle}{\left\langle \frac{d\mathbf{T}}{ds}, \mathbf{p} - \mathbf{r} \right\rangle - 1}. \quad (30)$$

It needs to be emphasized that this reformulation of the equation allows us to reach the path using the orthogonal parametrization as now it is required that  $\left\langle \frac{d\mathbf{T}}{ds}, \mathbf{p} - \mathbf{r} \right\rangle \neq 1$ . In particular, this assumption is met when  $\left\langle \frac{d\mathbf{T}}{ds}, \mathbf{p} - \mathbf{r} \right\rangle = 0$ . It means that the controlled robot may move along the desired path. However, the problem is still local due to the form of the denominator of expression (30). It is noteworthy, though, that it might be assured that

$$\left\langle \frac{d\mathbf{T}}{ds}, \mathbf{p} - \mathbf{r} \right\rangle < 1, \quad (31)$$

i.e., the manipulator is close enough to the path during the whole motion, so the orthogonal singularity is always avoided. Although equation (31) is quite conservative and may be violated by some large oscillations, especially in the transient state, it significantly broadens the usability of the orthogonal projection in comparison with the previous approach presented in Section 3.2.

Hence, the equations describing robot evolution in the local frame, defined by equations (23) and (28) for the Serret–Frenet parametrization and the Bishop parametrization, respectively, may be properly rewritten.

**Serret–Frenet parametrization** Let us consider equation (30) in the definition of the time-derivatives of the vector  $\xi$  elements. Taking into account definition of the robot position with respect to the Serret–Frenet local

frame, given by equation (14), leads to the following equations

$$\dot{s} = -\frac{\langle \mathbf{T}, \dot{\mathbf{p}} \rangle}{\kappa \langle \mathbf{N}, \mathbf{p} - \mathbf{r} \rangle - 1} = -\frac{\mathbf{T}^T}{\kappa d_2 - 1} \dot{\mathbf{p}} = \mathbf{P}_1 \dot{\mathbf{p}}, \quad (32)$$

$$\begin{aligned} \dot{d}_2 &= \langle \mathbf{N}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle - \dot{s} \kappa \langle \mathbf{T}, \mathbf{p} - \mathbf{r} \rangle + \dot{s} \tau \langle \mathbf{B}, \mathbf{p} - \mathbf{r} \rangle = \\ &= \langle \mathbf{N}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle - \underbrace{\dot{s} \kappa d_1}_{=0} - \frac{\langle \mathbf{T}, \dot{\mathbf{p}} \rangle}{\kappa d_2 - 1} \tau d_3 = \\ &= \left\langle \mathbf{N} - \tau \frac{\mathbf{T} d_3}{\kappa d_2 - 1}, \dot{\mathbf{p}} \right\rangle - \langle \mathbf{N}, \dot{\mathbf{r}} \rangle = \\ &= \left( \mathbf{N} - \tau \frac{\mathbf{T} d_3}{\kappa d_2 - 1} \right)^T \dot{\mathbf{p}} - \underbrace{\langle \mathbf{N}, \mathbf{T} \rangle}_{=0} \dot{s} = \mathbf{P}_2 \dot{\mathbf{p}}, \quad (33) \end{aligned}$$

$$\begin{aligned} \dot{d}_3 &= \langle \mathbf{B}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle - \dot{s} \tau \langle \mathbf{N}, \mathbf{p} - \mathbf{r} \rangle = \\ &= \langle \mathbf{B}, \dot{\mathbf{p}} - \dot{\mathbf{r}} \rangle + \frac{\langle \mathbf{T}, \dot{\mathbf{p}} \rangle}{\kappa d_2 - 1} \tau d_2 = \\ &= \left\langle \mathbf{B} + \tau \frac{\mathbf{T} d_2}{\kappa d_2 - 1}, \dot{\mathbf{p}} \right\rangle - \langle \mathbf{B}, \dot{\mathbf{r}} \rangle = \\ &= \left( \mathbf{B} + \tau \frac{\mathbf{T} d_2}{\kappa d_2 - 1} \right)^T \dot{\mathbf{p}} - \underbrace{\langle \mathbf{B}, \mathbf{T} \rangle}_{=0} \dot{s} = \mathbf{P}_3 \dot{\mathbf{p}}. \quad (34) \end{aligned}$$

Equations (32), (33), and (34) correspond to equations (19), (21b), and (21c), respectively. It may be observed that after the translation of the orthogonal singularity the aforementioned equations can also be written in the concise form

$$\dot{\xi} = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \end{bmatrix} \dot{\mathbf{p}} + \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \mathbf{P} \dot{\mathbf{p}} + \mathbf{R}, \quad (35)$$

where the vector  $\mathbf{R}$  is equal to zero. It is worth noticing that for the Serret–Frenet parametrization the constraint (31) takes the form

$$d_2 < \frac{1}{\kappa}, \quad (36)$$

which indicates that the robot should always be located close enough to the curve in the normal vector direction. In particular, it may be conservatively assumed that the robot guidance point is close enough to the path along the whole curve by introducing the maximal value of the curvature in equation (36). As a result, the singularity avoidance is always guaranteed.

**Bishop parametrization** A similar approach may be applied for the Bishop parametrization. The analogous procedure leads us to the reformulation of equation (28)

$$\dot{s} = -\frac{\mathbf{T}^T}{k_1 d_2 + k_2 d_3 - 1} \dot{\mathbf{p}} = \mathbf{P}_1 \dot{\mathbf{p}}, \quad (37a)$$

$$\dot{d}_2 = \mathbf{N}_1^T \dot{\mathbf{p}} = \mathbf{P}_2 \dot{\mathbf{p}}, \quad (37b)$$

$$\dot{d}_3 = \mathbf{N}_2^T \dot{\mathbf{p}} = \mathbf{P}_3 \dot{\mathbf{p}}, \quad (37c)$$

which in the concise form is expressed as

$$\dot{\xi} = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix} \dot{p} + \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = P\dot{p} + R. \quad (38)$$

It may be observed that equations (37) correspond to equations (25), (27b), and (27c). Furthermore, it is worth noticing that again the interpretation of the constraint (31) for the Bishop parametrization,

$$k_1 d_2 + k_2 d_3 < 1, \quad (39)$$

is much more complicated as it is a linear combination of the robot positions in both normal vectors' directions. However, the new form of the denominator in expression (30) allows not only moving precisely along the given path, but also following zero-curvature paths. Thus, using this approach one may benefit from all Bishop parametrization advantages.

It is noteworthy that for both parametrization methods the vector  $R$  is reduced to the zero vector. However, it is involved in equations (35) and (38) in order to use the same general form of the kinematic controller as earlier. Moreover, it may be observed that the matrix  $P$  tends to the transposed matrix  $S$  while the robot guidance point tends to the local frame associated with the curve. It results from the fact that the positions in both normal vectors directions tend to zero. Hence, it may be formulated that

$$P \xrightarrow{d_2, d_3 \rightarrow 0} S^T. \quad (40)$$

Based on relation (40), it may be concluded that the matrix  $P$  tends to be a rotation matrix transforming the end-effector velocities expressed in the inertial frame to the reference frame associated with the path.

#### 4. Control Problem Statement

Let us consider a control task which is following the path in the three-dimensional space by the stationary holonomic manipulator.

The following assumptions have to be fulfilled:

- 1) The desired path is smooth enough, i.e., of  $C^2$  class (for the Bishop parametrization) or of  $C^3$  class (for the Serret-Frenet parametrization) and is located in the workspace of the manipulator,
- 2) The mathematical model of the manipulator following the path is defined by two groups of equations, namely by the dynamics (10) and the kinematics (35) or (38) (i.e., description of the robot relative to the path; the form depends on the chosen parametrization method),
- 3) The kinematics is known precisely and parameters of the manipulator dynamics may be fully (non-adaptive case) or partially known (adaptive case). Only position control is taken into account.

According to [11], the path following task is defined as a problem of enforcing a robot to converge to and follow the spatial curve while a desired velocity profile along the path is asymptotically tracked. It

means that the path is defined only geometrically and the desired motion of a local frame associated with the path may be chosen arbitrarily. Although the evolution of the local frame is time-dependent, the choice of its velocity profile is the secondary sub-problem of the path following task. In general, the path can be still understood as a pure geometrical object.

The equations describing robot motion following the path have a structure which is similar to non-holonomic constraints. Thus, the considered system has a cascade structure and the control algorithms may be designed based on the backstepping integrator method [12]. It means that the control law has to consist of two controllers working in parallel:

- (i) kinematic controller  $\dot{q}_{ref}$ —represents a vector of embedded control inputs, which ensure realization of the task for the geometric path tracking problem if the dynamics were not present. Such a controller generates a “velocity profile” which can be executed in practice to follow the desired curve in  $\mathbb{R}^3$ .
- (ii) dynamic controller  $u$ —as a consequence of the cascaded structure of the robot model, the system's velocities cannot be commanded directly, as it is assumed in the design of kinematic controller, and instead they must be realized as the output of the dynamics driven by  $u$ .

The schematic view of the control system cascade is presented in Figure 3.

##### 4.1. Control Law – Non-adaptive Case

The first stage of the cascade is the kinematic controller. It should generate velocity profiles  $\dot{q}_{ref}$  in order to follow the given curve. Taking into account the structure of equations describing the controlled object motion along the curve, (35) or (38), and also relation (9), the following kinematic controller is proposed based on [20]

$$\dot{q}_{ref} = J^{-1}P^{-1}(\dot{\xi}_d - K_k e_\xi - R), \quad (41)$$

where  $e_\xi = \xi - \xi_d$  is the path following error,  $\xi_d$  is the vector of the desired state of the real object with respect to the path, and  $K_k$  is the positive-definite matrix of control coefficients. In equation (41), it is assumed that only non-singular configurations of a non-redundant manipulator are taken into account. Thus, the inverse of the Jacobi matrix  $J$  is considered.

The second stage of the cascade is the dynamic controller which guarantees following the velocity profile  $\dot{q}_{ref}$  by generating proper control generalized forces  $u$ . The control algorithm is designed based on the fully known model of the manipulator dynamics (10). The following dynamic controller is proposed

$$u = M(q)\ddot{q}_{ref} + C(q, \dot{q})\dot{q}_{ref} + D(q) - K_d \dot{e}_q, \quad (42)$$

where  $\dot{e}_q = \dot{q} - \dot{q}_{ref}$  is the vector of the velocity profiles following errors, and  $K_d$  is the positive-definite coefficient matrix. Such a designed system guarantees asymptotic convergence of the defined errors ( $e_\xi, \dot{e}_q$ ) to zero.

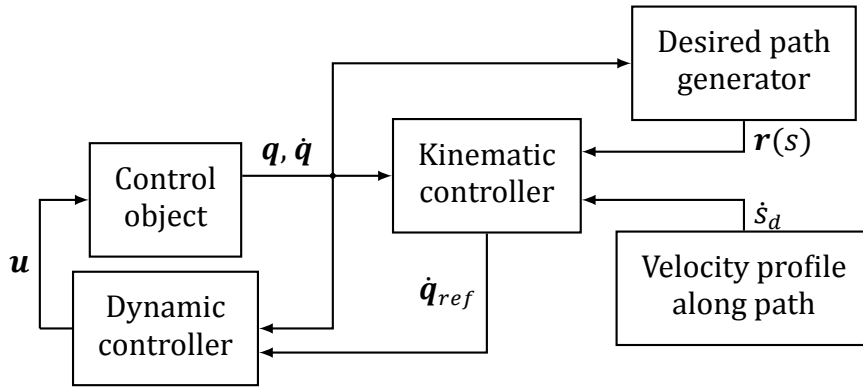


Figure 3. Control system structure

*Proof of the non-adaptive case.* Based on the backstepping integrator approach [12], the stability of the kinematic and the dynamic control laws may be considered separately with the usage of the Lyapunov theory. Thus, the kinematic level of the cascade is taken into account first. The system (35) or (38), depending on the chosen parametrization method, in the closed-feedback loop with the control law (41) may be expressed with the equation

$$\dot{\mathbf{e}}_{\xi} + \mathbf{K}_k \mathbf{e}_{\xi} = 0. \quad (43)$$

Let us define the Lyapunov-like function

$$V_1(\mathbf{e}_{\xi}) = \frac{1}{2} \mathbf{e}_{\xi}^T \mathbf{e}_{\xi}. \quad (44)$$

Thus, the time derivative of  $V_1$  is equal to

$$\begin{aligned} \dot{V}_1(\mathbf{e}_{\xi}) &= \mathbf{e}_{\xi}^T \dot{\mathbf{e}}_{\xi} = \mathbf{e}_{\xi}^T (-\mathbf{K}_k \mathbf{e}_{\xi}) = -\mathbf{e}_{\xi}^T \mathbf{K}_k \mathbf{e}_{\xi} = \\ &= -W_1(\mathbf{e}_{\xi}) \leq 0. \end{aligned} \quad (45)$$

It is clear that the path following errors  $\mathbf{e}_{\xi}$  converge asymptotically to zero based on the LaSalle invariance principle [3]. It confirms that the subsystem on the kinematic level of the cascade is asymptotically stable with zero equilibrium point for the positive-definite matrix  $\mathbf{K}_k$ .

Secondly, the dynamic level needs to be considered. It is noteworthy that the results from the previous cascade stage have to be taken into account. Thus, the function form of the dynamic level errors is strictly defined and depends on the reference velocity profiles given by equation (41). Let us define the Lyapunov-like function for the second level of the control system cascade. It depends on both the path following errors  $\mathbf{e}_{\xi}$  and the velocity profile following errors  $\dot{\mathbf{e}}_q$

$$V_2(\mathbf{e}_{\xi}, \dot{\mathbf{e}}_q) = V_1(\mathbf{e}_{\xi}) + \frac{1}{2} \dot{\mathbf{e}}_q^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{e}}_q. \quad (46)$$

The function  $V_2$  is non-negative due to positive definiteness of the inertia matrix  $\mathbf{M}$  and the definition of the function  $V_1$  given by equation (44). The time derivative of  $V_2$  is equal to

$$\dot{V}_2(\mathbf{e}_{\xi}, \dot{\mathbf{e}}_q) = \dot{V}_1(\mathbf{e}_{\xi}) + \dot{\mathbf{e}}_q^T \mathbf{M} \dot{\mathbf{e}}_q + \frac{1}{2} \dot{\mathbf{e}}_q^T \dot{\mathbf{M}} \dot{\mathbf{e}}_q. \quad (47)$$

Considering the system (10) in the closed-feedback loop with the control law (42)

$$\mathbf{M} \ddot{\mathbf{e}}_q + \mathbf{C} \dot{\mathbf{e}}_q + \mathbf{K}_d \dot{\mathbf{e}}_q = 0, \quad (48)$$

and the skew-symmetry between the inertia matrix  $\mathbf{M}$  and the Coriolis and centrifugal forces matrix  $\mathbf{C}$  [34]

$$\dot{\mathbf{M}} = \mathbf{C} + \mathbf{C}^T, \quad (49)$$

equation (47) may be rewritten as

$$\begin{aligned} \dot{V}_2(\mathbf{e}_{\xi}, \dot{\mathbf{e}}_q) &= \dot{V}_1(\mathbf{e}_{\xi}) - \dot{\mathbf{e}}_q^T \mathbf{K}_d \dot{\mathbf{e}}_q = \\ &= -W_2(\mathbf{e}_{\xi}, \dot{\mathbf{e}}_q) \leq 0. \end{aligned} \quad (50)$$

Once again, the LaSalle invariance principle [3] may be harnessed. Based on that theorem it is proven that the errors  $(\mathbf{e}_{\xi}, \dot{\mathbf{e}}_q)$  asymptotically converge to zero for the positive-definite matrix  $\mathbf{K}_d$ . Thus, the system (48) has an asymptotically stable zero equilibrium point. It is noteworthy that the stability analysis for the dynamic level includes the trajectories of the errors achieved on the previous level. Hence, the proposed Lyapunov-like functions confirm that the designed control law guarantees the correct path following and the control system cascade is stable on all levels.  $\square$

#### 4.2. Control Law – Adaptive Case

In the previous subsection, the full knowledge about the manipulator's dynamics has been assumed. Now the case with the partial knowledge about the dynamics will be considered.

For the partially known model of the manipulator's dynamics (11), the following dynamic adaptive control law is proposed

$$\begin{aligned} \mathbf{u} &= \mathbf{M}_0(\mathbf{q}) \ddot{\mathbf{q}}_{ref} + \mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}_{ref} + \mathbf{D}_0(\mathbf{q}) + \\ &+ \mathbf{Y}(\ddot{\mathbf{q}}_{ref}, \dot{\mathbf{q}}_{ref}, \dot{\mathbf{q}}, \mathbf{q}) \hat{\mathbf{a}}(t) - \mathbf{K}_d \dot{\mathbf{e}}_q, \end{aligned} \quad (51)$$

where  $\hat{\mathbf{a}}(t)$  is the current estimate of the vector of unknown parameters generated due to the adaptation law defined below

$$\dot{\hat{\mathbf{a}}}(t) = \dot{\hat{\mathbf{a}}}(t) = -\mathbf{\Gamma} \mathbf{Y}^T(\ddot{\mathbf{q}}_{ref}, \dot{\mathbf{q}}_{ref}, \dot{\mathbf{q}}, \mathbf{q}) \dot{\mathbf{e}}_q. \quad (52)$$

The symbol  $\tilde{\mathbf{a}}(t) = \hat{\mathbf{a}}(t) - \mathbf{a}$  denotes the difference between the estimated values and the unknown constant real values of the parameters, and  $\mathbf{\Gamma} = \text{diag}\{\gamma\}$  is a positive-definite matrix of adaptation gains.



*Proof of the adaptive case.* The adaptive case applies only to the equations of the dynamics, i.e., the second stage of the cascade. Therefore, the form of the kinematic controller and the proof of convergence for the first stage of the cascade remain unchanged.

The equations of the system (11) with the closed loop of the control law (51) and the adaptation law (52) can be calculated as follows (for the transparency of the notation the matrix arguments are neglected):

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{D} &= \mathbf{M}_0\ddot{\mathbf{q}}_{ref} + \mathbf{C}_0\dot{\mathbf{q}}_{ref} + \mathbf{D}_0 + \mathbf{Y}_r\hat{\mathbf{a}} + \\ &\quad - \mathbf{K}_d\dot{\mathbf{e}}_q = \\ &= \mathbf{M}_0\ddot{\mathbf{q}}_{ref} + \mathbf{C}_0\dot{\mathbf{q}}_{ref} + \mathbf{D}_0 + \mathbf{Y}_r\hat{\mathbf{a}} + \\ &\quad + \mathbf{Y}_r\mathbf{a} - \mathbf{Y}_r\mathbf{a} - \mathbf{K}_d\dot{\mathbf{e}}_q = \\ &= \mathbf{M}\ddot{\mathbf{q}}_{ref} + \mathbf{C}\dot{\mathbf{q}}_{ref} + \mathbf{D} + \mathbf{Y}_r\tilde{\mathbf{a}} + \\ &\quad - \mathbf{K}_d\dot{\mathbf{e}}_q \end{aligned}$$

After rewriting the above equation, one gets the form of the closed-loop system

$$\begin{aligned} \mathbf{M}(\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_{ref}) + \mathbf{C}(\dot{\mathbf{q}} - \dot{\mathbf{q}}_{ref}) - \mathbf{Y}_r\tilde{\mathbf{a}} + \mathbf{K}_d\dot{\mathbf{e}}_q &= \\ = \mathbf{M}\ddot{\mathbf{e}}_q + \mathbf{C}\dot{\mathbf{e}}_q - \mathbf{Y}_r\tilde{\mathbf{a}} + \mathbf{K}_d\dot{\mathbf{e}}_q &= 0. \end{aligned} \quad (53)$$

For the second stage—the dynamics level—the Lyapunov-like function for the adaptive case has the form

$$\begin{aligned} V_{2a}(e_\xi, \dot{e}_q, \tilde{\mathbf{a}}(t)) &= V_1(e_\xi) + \frac{1}{2}\dot{e}_q^T \mathbf{M}(q)\dot{e}_q + \\ &\quad + \frac{1}{2}\tilde{\mathbf{a}}^T(t)\Gamma^{-1}\tilde{\mathbf{a}}(t). \end{aligned} \quad (54)$$

Time derivative of the  $V_{2a}$  function is equal to

$$\begin{aligned} \dot{V}_{2a} &= \dot{V}_1 + \dot{e}_q^T \mathbf{M}(q)\ddot{e}_q + \frac{1}{2}\dot{e}_q^T \dot{\mathbf{M}}(q)\dot{e}_q + \\ &\quad + \tilde{\mathbf{a}}^T \Gamma^{-1}\dot{\tilde{\mathbf{a}}}. \end{aligned} \quad (55)$$

Putting the closed-loop system's dynamics (53), the time derivative of the  $V_1$  (45) and the adaptation law (52) into the equation (55) we obtain

$$\begin{aligned} \dot{V}_{2a}(e_\xi, \dot{e}_q) &= \dot{V}_1(e_\xi) - \dot{e}_q^T \mathbf{K}_d \dot{e}_q = \\ &= -W_{2a}(e_\xi, \dot{e}_q) \leq 0. \end{aligned} \quad (56)$$

From the LaSalle theorem we can conclude that the invariant set is defined by  $W_{2a}(e_\xi, \dot{e}_q) = 0$ . As a consequence, the errors  $(e_\xi, \dot{e}_q)$  asymptotically converge to zero and the point  $(e_\xi, \dot{e}_q) = (0, 0)$  is the asymptotically stable equilibrium point of the closed-loop system.

It is worth to be mentioned that from the proof dedicated to the adaptive case of the dynamic control it has been only shown that the estimation errors  $\tilde{\mathbf{a}}$  remain limited and do not necessarily tend to zero.

This ends the proof.  $\square$

## 5. Simulation Study

Numerical simulations were conducted in order to verify the performance of the presented algorithms. It was assumed that the robot was a non-redundant

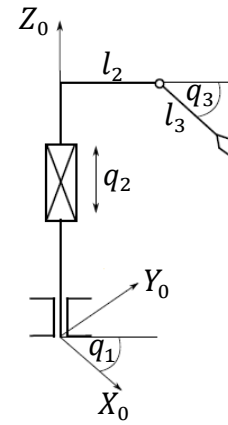
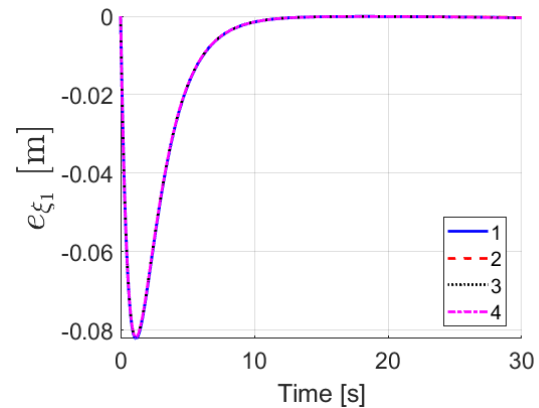
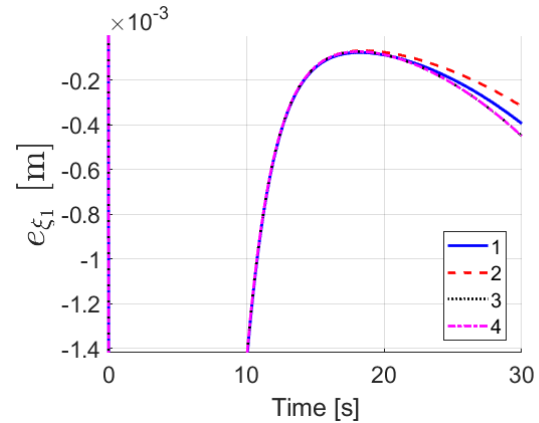


Figure 4. Structure of the RTR manipulator



(a) Comparison of different parametrization methods

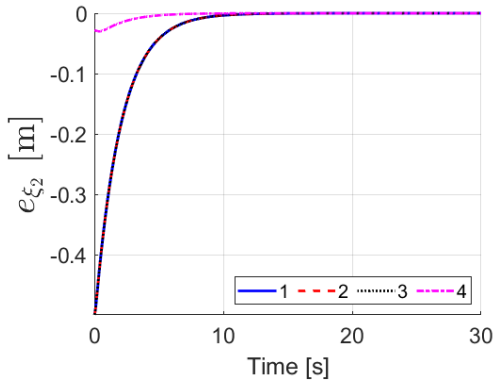


(b) Zoomed view of the graph

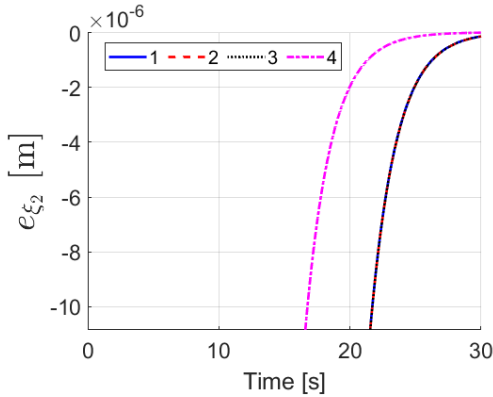
Figure 5. Plots of path following error  $e_{\xi_1} = s - s_d$

- 1 – Serret–Frenet parametrization,
- 2 – Bishop parametrization based on the Serret–Frenet frame,
- 3 – Bishop parametrization initialized as the Serret–Frenet frame,
- 4 – Bishop parametrization initialized arbitrarily

holonomic stationary manipulator of three degrees of freedom (rotational, translational and rotational). Its schematic view is presented in Figure 4 [20]. The manipulator parameters were chosen as follows. The manipulator links weighed 20kg and their lengths were equal to  $l_2 = 1.5\text{m}$  and  $l_3 = 1\text{m}$ .



(a) Comparison of different parametrization methods



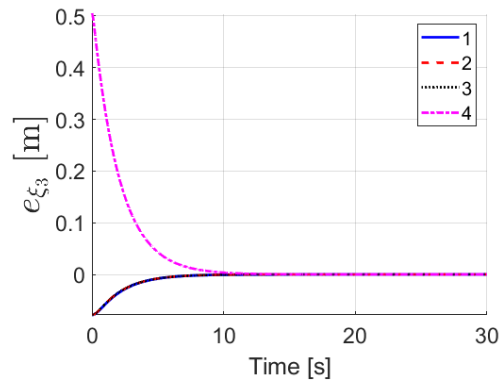
(b) Zoomed view of the graph

**Figure 6.** Plots of path following error  $e_{\xi_2} = d_2 - d_{2d}$   
 1 – Serret–Frenet parametrization,  
 2 – Bishop parametrization based on the Serret–Frenet frame,  
 3 – Bishop parametrization initialized as the Serret–Frenet frame,  
 4 – Bishop parametrization initialized arbitrarily

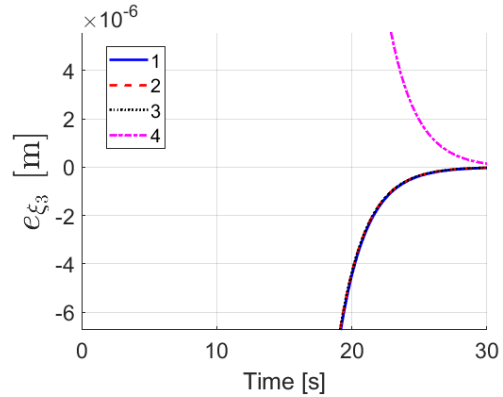
The considered control task was following the cylindrical helix defined with equation [27]

$$\mathbf{r}(s) = \left( a \cos \frac{s}{c} \quad a \sin \frac{s}{c} \quad \frac{bs}{c} \right)^T, \quad (57)$$

where  $c = \sqrt{a^2 + b^2}$ . The following values of the helix parameters were assumed:  $a = 1$ ,  $b = 0.1$ . Different definitions of the local frame associated with the given curve were taken into account: the Serret–Frenet frame, the Bishop frame based on the definition (4), and the Bishop frame derived from the Serret–Frenet frame according to the transformation (7). Furthermore, the Bishop frame was initialized in two configurations. One of them was equal to the Serret–Frenet frame in the initial state. For the second case the frame was initialized with an arbitrary choice of the normal vectors, different from the Serret–Frenet frame. For the transformation (7), the value  $\theta(0) = 0$  was assumed. As a result, for this case the Serret–Frenet frame and the Bishop frame, which is created based on the previous one, were also equal to each other in the initial state. Moreover, the following coefficient matrices were taken into account:  $\mathbf{K}_k = \text{diag}_{3 \times 3} \{0.5\}$ ,  $\mathbf{K}_d = \text{diag}_{3 \times 3} \{100\}$ . Simulations lasted  $t = 30$ s. The robot was expected to reach the desired path and



(a) Comparison of different parametrization methods



(b) Zoomed view of the graph

**Figure 7.** Plots of path following error  $e_{\xi_3} = d_3 - d_{3d}$   
 1 – Serret–Frenet parametrization,  
 2 – Bishop parametrization based on the Serret–Frenet frame,  
 3 – Bishop parametrization initialized as the Serret–Frenet frame,  
 4 – Bishop parametrization initialized arbitrarily

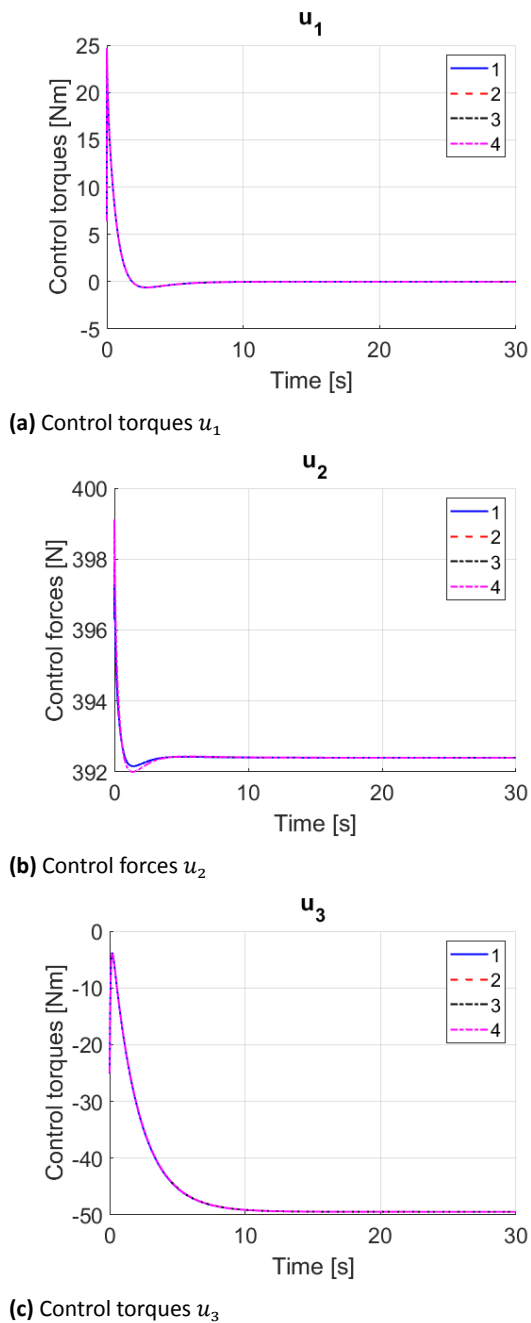
move along it. The velocity profile along the curve defined for the local reference frame resulting from the chosen parametrization method was constant during the whole motion and equal to

$$\dot{s}_d = \frac{1}{5}. \quad (58)$$

### 5.1. Singularity on the Path

Firstly, the simulation case when the orthogonality singularity is located on the path was analyzed. In Figures 5–7 the path following errors are presented for different parametrization methods. For every graph the magnification of the view is also provided. The presented graphs confirm that the robot approaches the path correctly as the end-effector position errors in the local frame, associated with the given curve, converge toward zero. It is noteworthy that the control system may behave differently depending on the chosen parametrization method. In particular, it may be observed for the Bishop frames defined for various initial conditions. However, their evolution is always the same and described with equation (4).

It may be observed in Figures 6 and 7 that due to different initial conditions of the Bishop frame, there



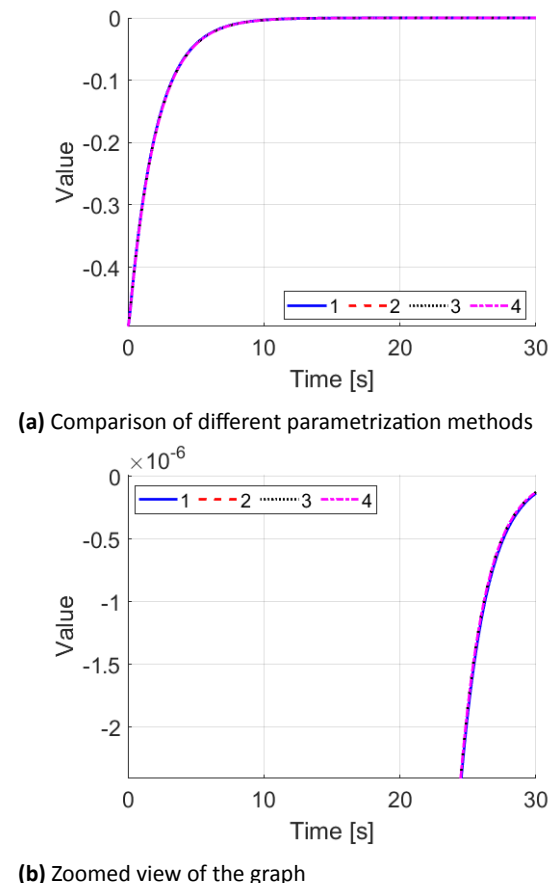
**Figure 8.** Plots of control torques and forces  $\mathbf{u}$   
 1 – Serret–Frenet parametrization,  
 2 – Bishop parametrization based on the Serret–Frenet frame,  
 3 – Bishop parametrization initialized as the Serret–Frenet frame,  
 4 – Bishop parametrization initialized arbitrarily

are different trajectories of path following errors in the normal vectors directions. However, for every case the errors converge to zero. In contrast, the trajectories for the  $e_{\xi_1}$  error are the same. It means that the local frame moves along the curve in the same way in spite of different evolution definitions. It results from the fact that all considered parametrization methods share the definition of the tangential vector  $\mathbf{T}$  and the same velocity profile along the path is taken into account. The differences may be observed in the frame

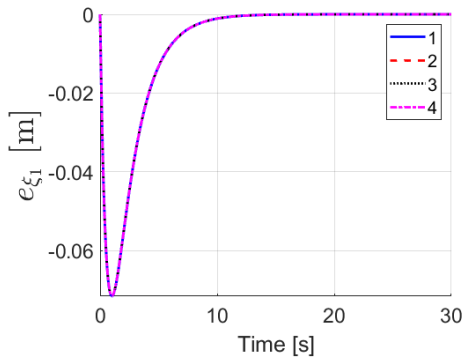
rotations with respect to the curve, which are visualized in Figure 18.

Furthermore, Figure 8 shows control generalized forces for every joint which allowed the robot to achieve the presented performance. It is noteworthy that the differences between parametrization methods are not relevant.

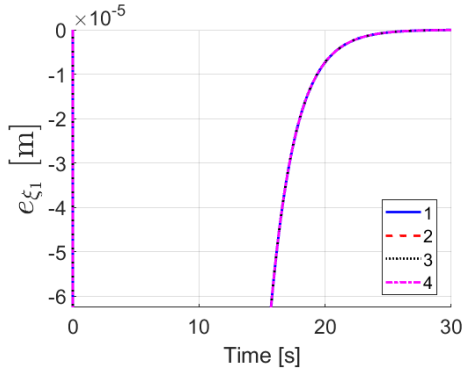
However, it might be worrying that at a certain time moment the trajectories presented in Figure 5b begin to move away from the zero value. It means that the constraint of the orthogonal projection, i.e., the end-effector and the virtual object that are in the closest proximity, is disrupted. Thus, it may be taken under discussion if the assumption that the manipulator tip is located in the normal plane of the local frame is satisfied. This phenomenon may be interpreted as difficulties of the local frame associated with the curve in maintaining the pace of the end-effector motion. Such behavior results from the fact that the system gets close to the orthogonal singularity defined with equation (18). The values of the denominator of this expression for different parametrization methods are shown in Figure 9. It is worth noticing that the



**Figure 9.** Value of the denominator of the expression (18) defining curvilinear velocity  $\dot{s}$   
 1 – Serret–Frenet parametrization,  
 2 – Bishop parametrization based on the Serret–Frenet frame,  
 3 – Bishop parametrization initialized as the Serret–Frenet frame,  
 4 – Bishop parametrization initialized arbitrarily



(a) Comparison of different parametrization methods



(b) Zoomed view of the graph

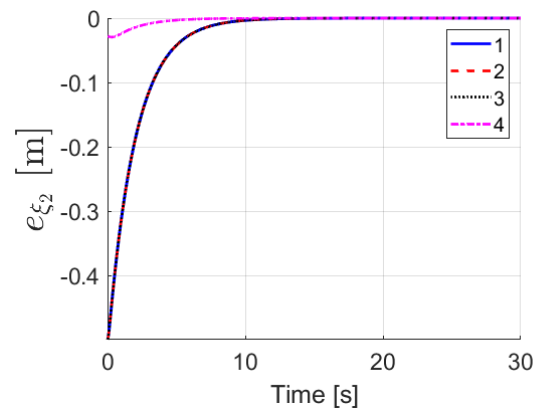
**Figure 10.** Plots of path following error  $e_{\xi_1} = s - s_d$  for the avoided singularity

- 1 – Serret–Frenet parametrization,
- 2 – Bishop parametrization based on the Serret–Frenet frame,
- 3 – Bishop parametrization initialized as the Serret–Frenet frame,
- 4 – Bishop parametrization initialized arbitrarily

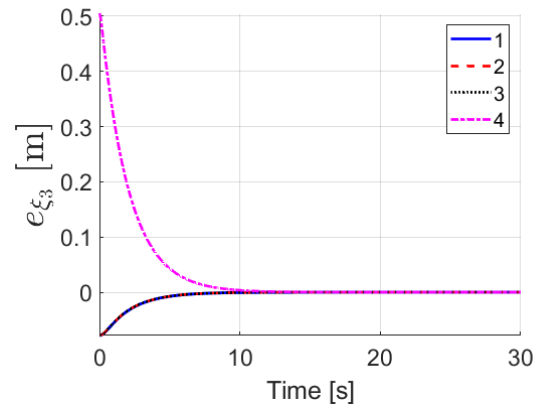
value of this expression converges to zero while the manipulator end-effector gets closer to the path. It confirms that the presented solution may be applied only for asymptotic approaching the path. The closer to the singularity the system is, the higher inaccuracy may be observed in the trajectories associated with error  $e_{\xi_1}$ . However, if the algorithm is robust to getting close to the singularity, the presented approach may be applied to reach any positions outside the given path. The geometrical interpretation of the denominator form for the Serret–Frenet parametrization, which is given in equation (19), is quite clear. It is strictly connected with the motion in the normal vector  $N$  direction. Thus, taking into account proper limitations for motion in this direction may guarantee that the singularity is not reached. However, in the Bishop frame version of the denominator, which is presented in equation (24), the combined motion in both normal vectors' directions is considered. Hence, it may be more difficult to define the manipulator motion avoiding the orthogonal singularity for this method.

### 5.2. Orthogonal Singularity Avoidance

Another way to avoid the orthogonal singularity is to translate it away from the path. Such an approach



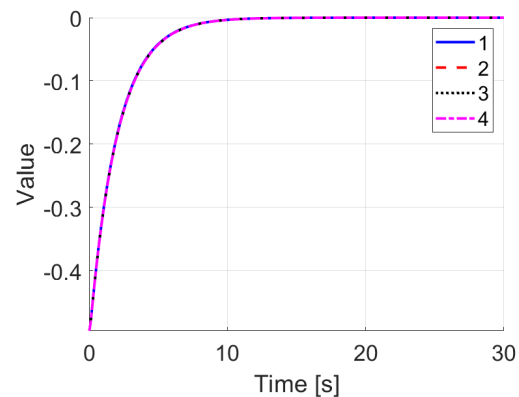
(a) Plots of path following error  $e_{\xi_2} = d_2 - d_{2d}$



(b) Plots of path following error  $e_{\xi_3} = d_3 - d_{3d}$

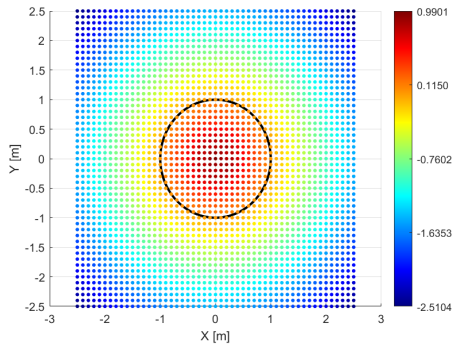
**Figure 11.** Plots of path following errors in the normal vectors directions for the singularity avoided

- 1 – Serret–Frenet parametrization,
- 2 – Bishop parametrization based on the Serret–Frenet frame,
- 3 – Bishop parametrization initialized as the Serret–Frenet frame,
- 4 – Bishop parametrization initialized arbitrarily

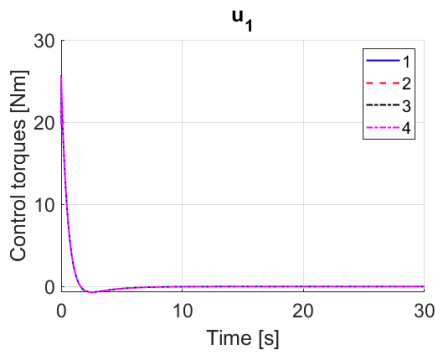


**Figure 12.** Value of the expression  $\left(\frac{dR}{ds}, \mathbf{p} - \mathbf{r}\right)$  for the singularity avoided

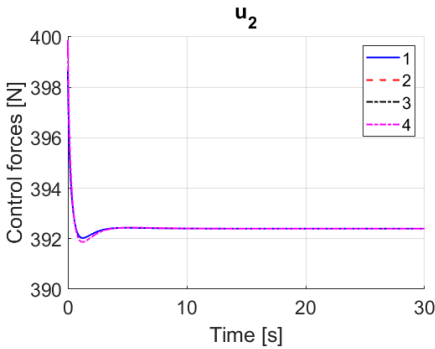
- 1 – Serret–Frenet parametrization,
- 2 – Bishop parametrization based on the Serret–Frenet frame,
- 3 – Bishop parametrization initialized as the Serret–Frenet frame,
- 4 – Bishop parametrization initialized arbitrarily



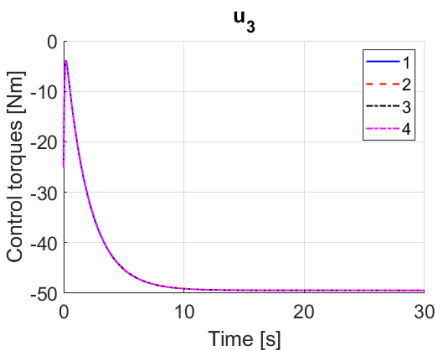
**Figure 13.** Value of the expression  $\left(\frac{dT}{ds}, \mathbf{p} - \mathbf{r}\right)$  for different initial configurations of the RTR manipulator



**(a)** Control torques  $u_1$



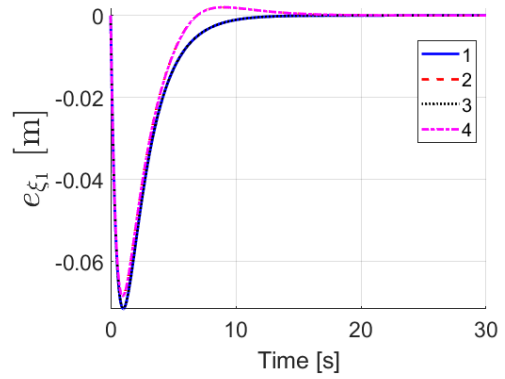
**(b)** Control forces  $u_2$



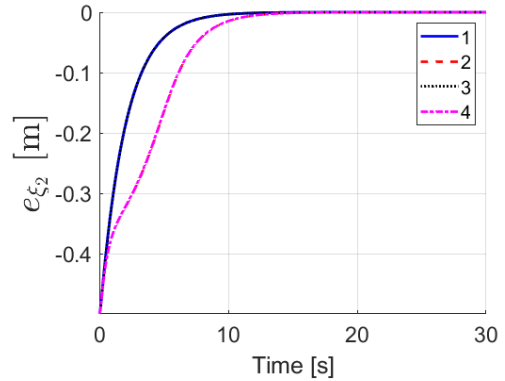
**(c)** Control torques  $u_3$

**Figure 14.** Plots of control torques and forces  $\mathbf{u}$  for the singularity avoided

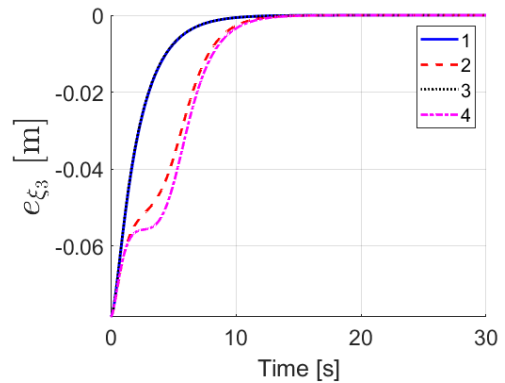
- 1 – Serret–Frenet parametrization,
- 2 – Bishop parametrization based on the Serret–Frenet frame,
- 3 – Bishop parametrization initialized as the Serret–Frenet frame,
- 4 – Bishop parametrization initialized arbitrarily



**(a)** Path following errors  $e_{\xi_1}$



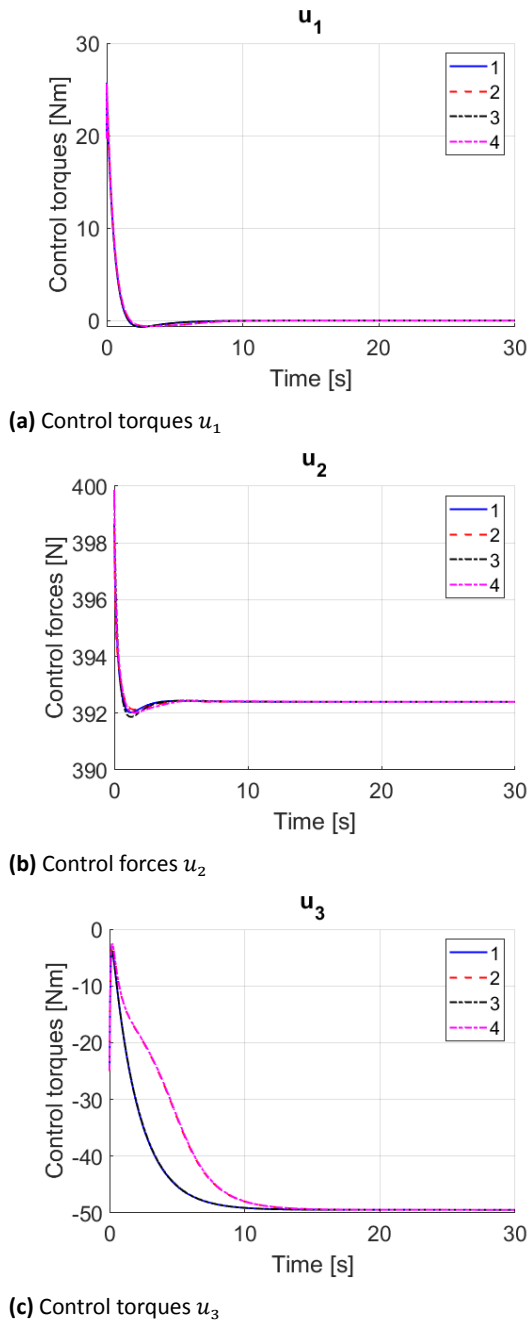
**(b)** Path following errors  $e_{\xi_2}$



**(c)** Path following errors  $e_{\xi_3}$

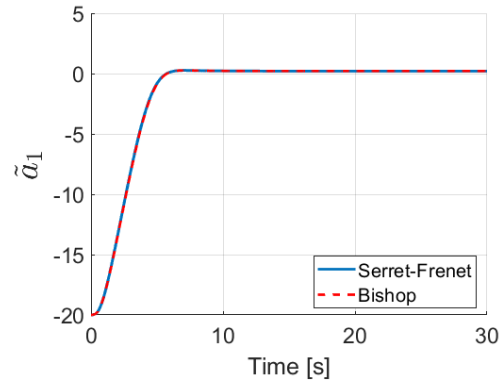
**Figure 15.** Plots of path following errors  $e_{\xi}$  for the adaptive and non-adaptive controllers  
 1 – Serret–Frenet parametrization (non-adaptive case),  
 2 – Serret–Frenet parametrization (adaptive case),  
 3 – Bishop parametrization (non-adaptive case),  
 4 – Bishop parametrization (adaptive case)

has been described in section 3.3. In Figures 10 and 11 the path following errors for this case are shown. It is noteworthy that the proposed reformulation does not affect the errors in the normal vectors directions. However, significant changes may be observed for the error  $e_{\xi_1}$ . This error preserves its monotonicity and converges to zero. It is the result of the singularity translation. The denominator of the expression (30) tends to  $-1$  as the robot approaches the path. The value of the expression  $\left(\frac{dT}{ds}, \mathbf{p} - \mathbf{r}\right)$  is presented in Figure 12. That value is lower than zero during the

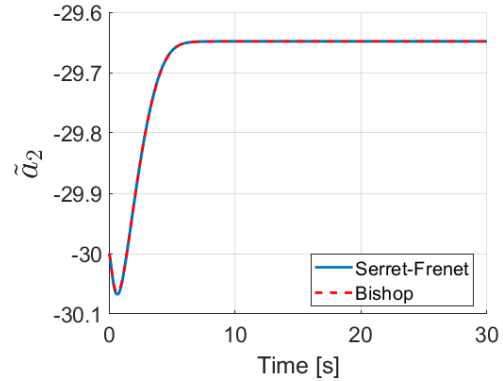


**Figure 16.** Plots of control torques and forces  $\mathbf{u}$  for the adaptive and non-adaptive controllers  
 1 – Serret–Frenet parametrization (non-adaptive case),  
 2 – Serret–Frenet parametrization (adaptive case),  
 3 – Bishop parametrization (non-adaptive case),  
 4 – Bishop parametrization (adaptive case)

whole motion. As a result, the local constraints given by equations (36) and (39) are satisfied. It shows that it is quite easy to meet those assumptions for the cylindrical helix, even though the form for the Bishop parametrization is more complicated. Moreover, in Figure 13 values of the expression  $\left\langle \frac{d\mathbf{r}}{ds}, \mathbf{p} - \mathbf{r} \right\rangle$  for different initial configurations of the RTR manipulator are presented. They are the same for both parametrization methods. It may be observed that the highest values are located in the middle of the helix,



(a) Estimation error  $\tilde{a}_1$



(b) Estimation error  $\tilde{a}_2$

**Figure 17.** Plots of estimation errors  $\tilde{\mathbf{a}}$

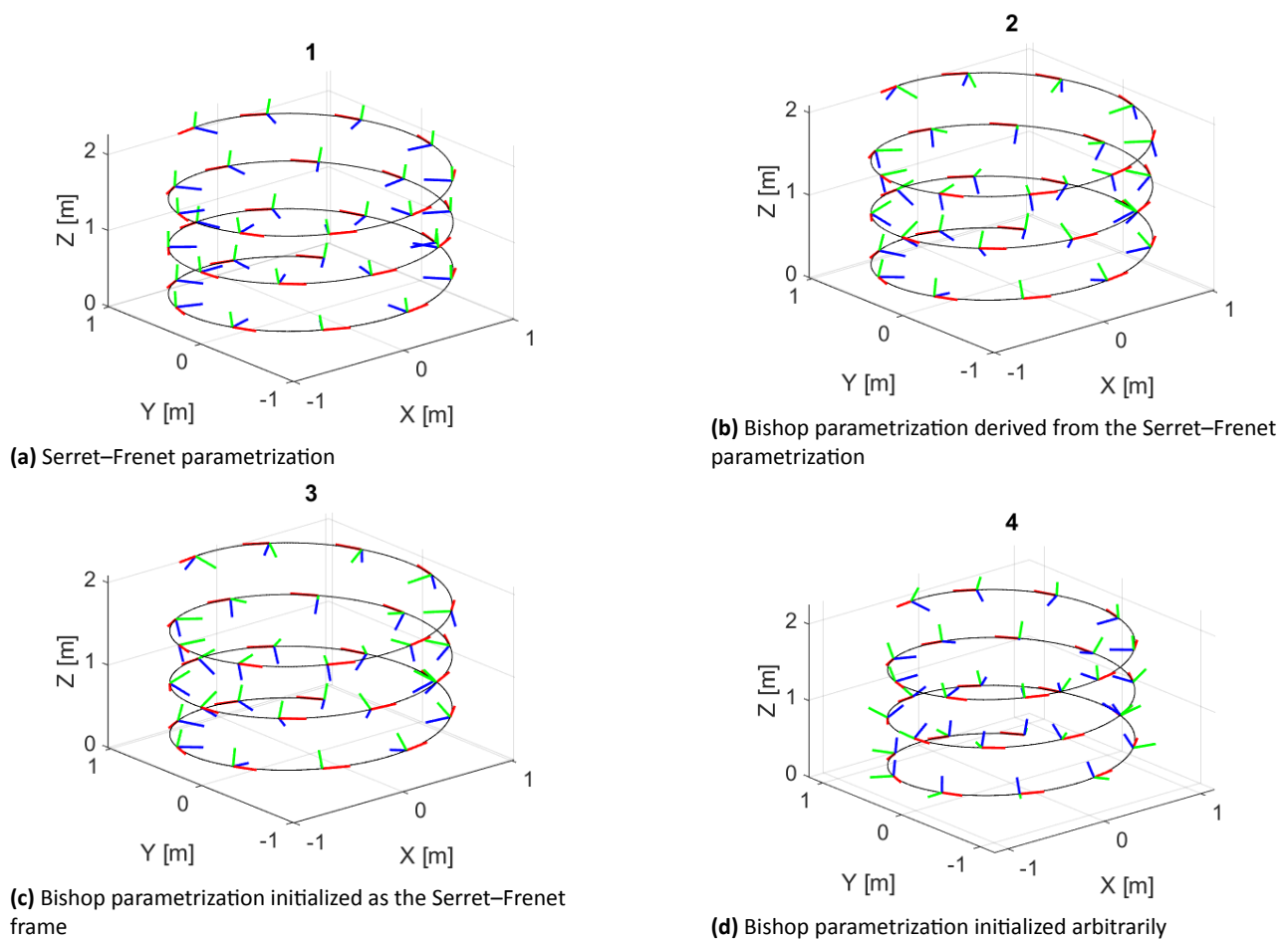
but they are lower than 1. It means that for the considered path there is no singularity resulting from the orthogonal projection.

In Figure 14 generalized control forces are presented. They allowed the manipulator to perform the path following task. It is noteworthy that they are the same as that of the base solution case (see Fig. 8), although the form of the description of the robot with respect to the path has changed.

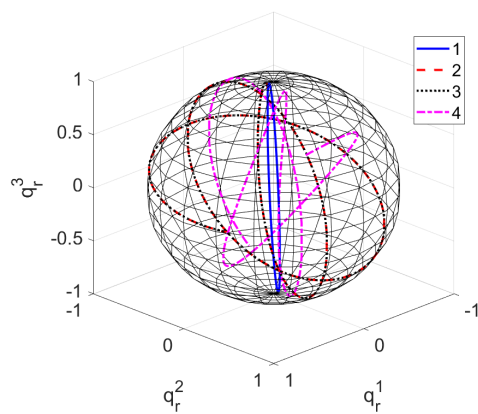
### 5.3. Adaptive Controller

In the last simulation case study the partial knowledge of the dynamic parameters of the manipulator was assumed. This section proves that the proposed control algorithms may be applied even if the dynamic model is not fully known. At the second stage of the control cascade the control law given with equation (51) was considered. It was assumed that there are two parameters which values are unknown:  $a_1 = m_3 l_3$ ,  $a_2 = m_3 l_3 l_2$ . The gain matrix of the adaptation law (52) was equal to  $\mathbf{\Gamma} = \text{diag}_{2 \times 2}\{50\}$  and the estimated values  $\hat{\mathbf{a}}$  were initialized as zero values, i.e.,  $\hat{\mathbf{a}}(0) = (0 \ 0)^T$ .

In Figure 15 the path following errors are presented. The performance of the adaptive controller was compared with the performance of the non-adaptive one applied to the model with the fully known dynamics. Results for both the Serret–Frenet parametrization and the Bishop parametrization are shown. It is noteworthy that for the adaptive case



**Figure 18.** Evolution of the local frame along the given curve depending on the parametrization method



**Figure 19.** Evolution of the local frame orientation presented in the unit quaternion space (real part of the quaternion neglected)  
 1 – Serret-Frenet parametrization,  
 2 – Bishop parametrization based on the Serret-Frenet frame,  
 3 – Bishop parametrization initialized as the Serret-Frenet frame,  
 4 – Bishop parametrization initialized arbitrarily

the pace of convergence is slightly slower. However, the errors converge to zero and the orthogonal singularity is avoided. Furthermore, the choice of the parametrization method is not so crucial for the

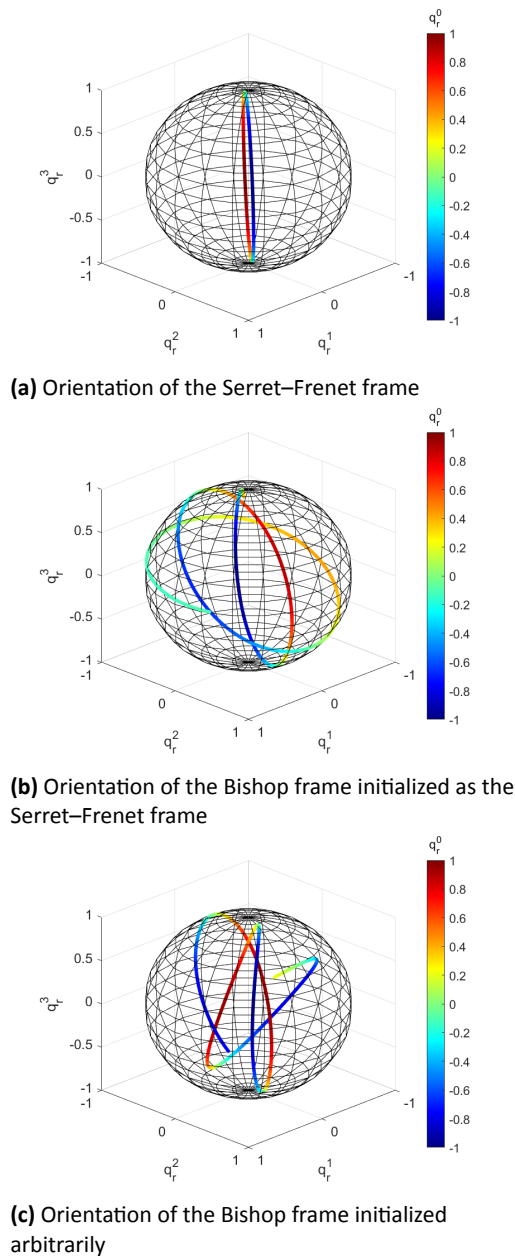
manipulator performance. The achieved results are comparable.

In Figure 16 the control signals are compared. The adaptive controller hardly changes the generated values. The initial peak, resulting from errors in the initial state, is quite similar and the control generalized forces tend to the same values as for the non-adaptive case.

Finally, in Figure 17, the estimation errors of the unknown parameters are presented. It is worth emphasizing that the estimated values do not converge to the real ones. As mentioned in section 4.2, the estimated values are limited, but the proposed control law does not guarantee the correctness of the estimation. Despite that fact, the path is approached and followed properly. Thus, the adaptive controller may be applied when the manipulator dynamic parameters are only partially known or even completely unknown. Moreover, the choice of the parametrization method on the kinematic level does not have any impact on the estimation performed by the adaptive controller on the dynamic level of the control system cascade.

#### 5.4. Local Frame Orientation Analysis

Verification of the frames behavior along the path is also a crucial aspect of the comparative analysis. Evolution along the path of all considered local frames is presented in Figure 18. The tangent vector  $T$  is always denoted with a red colour, while



**Figure 20.** Orientation of the local frame presented in the unit quaternion space

normal vectors of the local reference frame are denoted with blue and green. For the Serret-Frenet parametrization, they refer to the normal vector and the binormal vector, respectively. For the Bishop,

frame the blue vector corresponds with the vector  $N_1$  and the green one with the vector  $N_2$ . It is noteworthy that the Serret–Frenet frame preserves its normal vector directed to the center of the helix. It results from the fact that the frame comes back to the same orientation every time when the curve goes through the same axis of the coordinate frame [10]. The Bishop frame cannot be characterized with this feature. Non-zero torsion of the helix results in frame rotations with a constant speed. However, its evolution guarantees reduction of the performed rotations around the curve during the whole motion [30]. Figure 18 shows that the same evolution may be achieved by transforming the Serret–Frenet frame to the Bishop frame (Fig. 18b) or by defining the Bishop frame directly from the definition based on the relatively parallel transport of vectors (Fig. 18c). Different initial conditions of the Bishop frame are visualized in Figure 18d. The frame in the initial state is rotated in comparison with the frames presented in Figures 18b and 18c. However, it may be observed that it evolves along the curve exactly in the same manner.

As mentioned earlier, the Bishop parametrization minimizes rotations of the local frame along the curve. It may be observed in the unit quaternion space. In Figure 19, trajectories of frames' rotation evolution are presented. In the figure the trajectories are placed in the unit sphere. Moreover, the value of the real part of the quaternion is neglected. It is worth noticing that the trajectory of the Serret–Frenet frame rotation is a closed curve. It corresponds with the fact that the normal vector is directed to the center of the curvature all the time. Furthermore, the initial point of the Bishop frame initialized in the Serret–Frenet configuration (no. 2 and 3) is identical as the Serret–Frenet initial point. However, the evolution is completely different. In addition, characteristics of the motion of the Bishop frame is similarly independent of the initial configuration.

Finally, in Figure 20 the values of quaternions for respective local frames are presented. It may be noticed that the real part value is quite high, in the sense of absolute value, for the trajectories parts that are far away from the unit sphere surface. This may be observed particularly for the Serret–Frenet frame. The real part value of the Bishop frame is significantly lower as the trajectories are often placed on the sphere

**Table 1.** Comparison of the orthogonal parametrization methods

Feature	Serret–Frenet parametrization	Bishop parametrization
Minimal class of a curve	$\mathcal{C}^3$	$\mathcal{C}^2$
Defined for zero-curvature points	No	Yes
Definition in the initial state	Determined	Arbitrary
Rotations minimization	Around the normal vector $N$	Around the tangent vector $T$ (along a curve)
Geometrical invariants	$\kappa, \tau$	$k_1, k_2$
Orthogonal projection constraint	$\kappa d_2 \neq 1$	$k_1 d_2 + k_2 d_3 \neq 1$
Robot description with respect to a curve	$\dot{\xi} = P\dot{p} + R$	$\dot{\xi} = P\dot{p} + R$



surface. The lengths of these trajectories confirm the fact that the Bishop frame minimizes rotations along the curve. The length of the trajectory of the Serret–Frenet frame is equal to 14.9226, whereas for the Bishop frame it is equal to 14.8539 independently of its initial condition. The difference may be even more significant for other curves or curve parameters.

## 6. Conclusion

In this paper, the solution to the path following problem for stationary manipulators has been addressed. As a tool for following a curve in the three-dimensional space two different orthogonal parametrizations have been checked, namely the Serret–Frenet parametrization and the Bishop parametrization.

The orthogonal parametrization methods allow one to minimize the control problem dimensionality as the position in the tangent vector direction is fixed. However, they are correct only locally due to some constraints resulting from the orthogonal projection assumption.

There are some important differences between parametrization methods. They may play a crucial role while considering some more sophisticated paths than the cylindrical helix presented in the simulations. Despite a more intuitive geometrical interpretation of the Serret–Frenet frame, it imposes more constraints on a parametrized curve. First of all, Serret–Frenet parametrization is undefined in the zero-curvature points which means that many paths, including straight lines, cannot be parametrized. The Bishop frame is a solution for this problem. Moreover, the Bishop frame orientation may be chosen arbitrarily in the initial state. Although the presented behavior of the Serret–Frenet frame, which always points its normal vector to the center of the helix, may be desirable in various applications, the usage of the Bishop frame, which minimizes rotations around the curve, may appear much more beneficial for the direct control of orientation in the  $\mathbb{R}^3$  space. The brief comparison of the analyzed orthogonal parametrization methods is presented in Table 1.

It is very important to remember that orthogonal parametrizations are valid only locally due to singularities occurring in the description of the methods. In the previous approach the singularity was located on the path. Due to that fact, the proposed control algorithms allowed the robot only to approach the path asymptotically, not to move along the path. In this paper, it has been shown that the orthogonal parametrization may be transformed, i.e., that the singularity can be translated to a different location. Due to that fact, a path may be reached and a robot may move precisely along it. The presented translation of the orthogonal singularity solves the problem of the system excitation on the path. Not only can a robot move along the path, but also following zero-curvature paths for the Bishop frame is possible. It is strongly recommended to consider the robot description version with the orthogonal singularity outside the path for control applications.

The simulation study presented in the article let us verify the orthogonal path following algorithms based on different parametrization methods (Serret–Frenet, Bishop). For the cases with the orthogonal projection singularity outside the path the asymptotic convergence of the path following errors to zero was confirmed. Thus, the path following task was fulfilled correctly. The proposed control algorithms allowed the robot to approach the path and follow it while the desired velocity profile along the path was asymptotically tracked. In fact, the manipulator performance was independent of the chosen parametrization method as only the end-effector position was taken into account in the control task. It may be noticed that the generated control signals were similar for all simulation cases. Moreover, the form of the general description of the robot with respect to the local frame associated with the path (the first stage of the cascade) is the same for both methods.

In this article, the case with the partially known dynamics was also considered. Although the pace of error convergence was slower for the adaptive controller, it allowed the robot to follow the desired path correctly. It confirmed that despite the lack of knowledge of the dynamic parameters, the proposed control algorithms allow a robot to follow the desired path. It is worth noticing that the assumption of the fully known dynamics of the manipulator is not crucial for the presented algorithms. In general, it shows that any dynamic controller, which guarantees the asymptotic convergence of the velocity profiles' errors to zero, lets us succeed in the path following task. The object description with respect to the local frame associated with the given path is not affected by the change of the dynamic controller. Indeed, the kinematic controller designed for the first stage of the cascade is responsible for generating velocity profiles that guarantee motion along the given curve. The second stage of the control system cascade may be chosen arbitrarily according to the other requirements, e.g., the knowledge of the dynamic parameters.

In further research, it may be important to investigate control algorithms which allow controlling the robot orientation with respect to the local frame associated with the path. However, the considered robot will have to have more degrees of freedom than the RTR manipulator. It will be a verification of the presented features of the rotation changes of the considered frames. One may benefit from the Bishop frame properties in order to choose the local frame behavior appropriate for the considered application.

Furthermore, the impact of measurement noise on the robot behavior should be verified. On the first stage of the cascade, the kinematic level, the system may be quite close to the orthogonal singularity. Thus, some uncertainty may play a vital role in the manipulator performance. Even if the singularity is translated, the system may reach the singularity when approaching the path. Due to that fact, some research may be conducted according to [15].

Finally, some global algorithms should be taken into consideration. Hence, algorithms based on the

non-orthogonal parametrization, in particular the Bishop parametrization, may be worth developing as they do not introduce any additional constraints on robots, although the task dimensionality increases.

## AUTHORS

**Filip Dyba\*** – Department of Cybernetics and Robotics, Faculty of Electronics, Photonics and Microsystems, Wrocław University of Science and Technology, Janiszewskiego Street 11/17, Wrocław, 50 - 372, Poland, e-mail: filip.dyba@pwr.edu.pl.

**Alicja Mazur** – Department of Cybernetics and Robotics, Faculty of Electronics, Photonics and Microsystems, Wrocław University of Science and Technology, Janiszewskiego Street 11/17, Wrocław, 50 - 372, Poland, e-mail: alicja.mazur@pwr.edu.pl.

\*Corresponding author

## References

- [1] R. L. Bishop. "There is more than one way to frame a curve," *The American Mathematical Monthly*, vol. 82, no. 3, 1975, pp. 246–251.
- [2] M. Breivik, and T. Fossen. "Principles of guidance-based path following in 2D and 3D," *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, Seville, Spain, 2005, pp. 627–634; doi: 10.1109/CDC.2005.1582226.
- [3] C. Canudas de Wit, G. Bastin, and B. Siciliano, *Theory of Robot Control*, Springer-Verlag: London, 1996.
- [4] D. Carroll, E. Köse, and I. Sterling. "Improving Frenet's Frame Using Bishop's Frame," *Journal of Mathematics Research*, vol. 5, 2013, pp. 97–106; doi: 10.5539/jmr.v5n4p97.
- [5] L. Consolini, M. Maggiore, C. Nielsen, and M. Tosques. "Path following for the PVTOL aircraft," *Automatica*, vol. 46, no. 8, 2010, pp. 1284–1296; doi: 10.1016/j.automatica.2010.05.014.
- [6] F. Dyba, and A. Mazur. "Comparison of Curvilinear Methods in Path Following Task for a Holonomic Manipulator," A. Mazur and C. Zieliński, eds., *Advances in Robotics*, vol. 1, Warsaw University of Technology Publishing House, 2022, pp. 33–44, (in Polish).
- [7] P. Encarnacao, and A. Pascoal. "3D path following for autonomous underwater vehicle," *Proceedings of the 39th IEEE Conference on Decision and Control*, Sydney, 2000, pp. 2977–2982; doi: 10.1109/CDC.2000.914272.
- [8] F. Frenet. "Sur les courbes à double courbure," *Journal de Mathématiques Pures et Appliquées*, 1852, pp. 437–447.
- [9] M. Galicki. "Adaptive Control of Kinematically Redundant Manipulator along a Prescribed Geometric Path," K. Kozłowski, ed., *Robot Motion and Control. Lecture Notes in Control and Information Sciences*, vol. 335, Springer, 2006, pp. 129–139.
- [10] A. J. Hanson, and H. Ma, *Parallel Transport Approach to Curve Framing*. Technical report, Indiana University, 1995.
- [11] N. Hung et al. "A review of path following control strategies for autonomous robotic vehicles: Theory, simulations, and experiments," *Journal of Field Robotics*, vol. 40, no. 3, 2023, pp. 747–779; doi: 10.1002/rob.22142.
- [12] M. Krstić, I. Kanellakopoulos, and P. V. Kokotovic, *Nonlinear and Adaptive Control Design*, John Wiley & Sons, Inc., 1995.
- [13] X. Li, G. Zhao, and B. Li. "Generating optimal path by level set approach for a mobile robot moving in static/dynamic environments," *Applied Mathematical Modelling*, vol. 85, 2020, pp. 210–230.
- [14] Y.-L. Liao, M.-J. Zhang, and L. Wan. "Serret–Frenet frame based on path following control for underactuated unmanned surface vehicles with dynamic uncertainties," *Journal of Central South University*, vol. 22, 2015, pp. 214–223.
- [15] U. Libal, and J. Płaskonka. "Noise sensitivity of selected kinematic path following controllers for a unicycle," *Bulletin of the Polish Academy of Sciences: Technical Sciences*, vol. 62, no. 1, 2014, pp. 3 – 13; doi: 10.2478/bpasts-2014-0001.
- [16] H. Liu, and D. Pei. "Singularities of a space curve according to the relatively parallel adapted frame and its visualization," *Mathematical Problems in Engineering*, 2013; doi: 10.1155/2013/512020.
- [17] I. Lugo-Cárdenas, S. Salazar, and R. Lozano. "Lyapunov Based 3D Path Following Kinematic Controller for a Fixed Wing UAV," *IFAC-PapersOnLine*, vol. 50, no. 1, 2017, pp. 15946–15951; doi: 10.1016/j.ifacol.2017.08.1747, 20th IFAC World Congress.
- [18] A. Mazur. "Hybrid adaptive control laws solving a path following problem for non-holonomic mobile manipulators," *International Journal of Control*, vol. 77, no. 15, 2004, pp. 1297–1306; doi: 10.1080/0020717042000297162.
- [19] A. Mazur. *Model-based control for non-holonomic mobile manipulators*, Publishing House of Wrocław University of Science and Technology, 2009, (in Polish).
- [20] A. Mazur, J. Płaskonka, and M. Kaczmarek. "Following 3D paths by a manipulator," *Archives of Control Sciences*, vol. 25, no. 1, 2015, pp. 117–133; doi: 10.1515/acsc-2015-0008.
- [21] A. Mazur, and D. Szakiel. "On path following control of nonholonomic mobile manipulators," *International Journal of Applied Mathematics*

- and *Computer Science*, vol. 19, no. 4, 2009, pp. 561–574.
- [22] A. Micaelli, and C. Samson. “Trajectory tracking for unicycle-type and two-steering-wheels mobile robots,” *Technical Report No. 2097*, Sophia-Antipolis, 1993.
- [23] M. M. Michałek, and D. Pazderski, *Mobile robots control. Laboratory*, Publishing House of Poznan University of Technology, 2012, (in Polish).
- [24] M. M. Michałek. “A highly scalable path-following controller for N-trailers with off-axle hitching,” *Control Engineering Practice*, vol. 29, 2014, pp. 61–73; doi: 10.1016/j.conengprac.2014.04.001.
- [25] M. M. Michałek, and T. Gawron. “VFO path following control with guarantees of positionally constrained transients for unicycle-like robots with constrained control input,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 89, no. 1-2, 2018, pp. 191–210; doi: 10.1007/s10846-017-0482-0.
- [26] A. Morro, A. Sgorbissa, and R. Zaccaria. “Path following for unicycle robots with an arbitrary path curvature,” *IEEE Transactions on Robotics*, vol. 27, no. 5, 2011, pp. 1016–1023; doi: 10.1109/TRO.2011.2148250.
- [27] J. Oprea. *Differential Geometry and Its Applications*, Prentice Hall, 2007.
- [28] 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.
- [29] M. Rokonuzzaman, N. Mohajer, S. Nahavandi, and S. Mohamed. “Review and performance evaluation of path tracking controllers of autonomous vehicles,” *IET Intelligent Transport Systems*, vol. 15, no. 5, 2021, pp. 646–670; doi: 10.1049/itr2.12051.
- [30] J. M. Selig, and Y. Wu. “Interpolated rigid-body motions and robotics,” *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2006, pp. 1086–1091; doi: 10.1109/IROS.2006.281815.
- [31] J.-A. Serret. “Sur quelques formules relatives à la théorie des courbes à double courbure,” *Journal de Mathématiques Pures et Appliquées*, 1851, pp. 193–207.
- [32] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*, Springer, 2008.
- [33] 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*, IEEE, 2003, pp. 1765–1770; doi: 10.1109/CDC.2003.1272868.
- [34] K. Tchoń, A. Mazur, I. Duleba, R. Hossa, and R. Muszyński, *Manipulators and Mobile Robots: Models, Motion Planning, Control*, Academic Publishing House PLJ, 2000, (in Polish).