

Jaroslav Mlýnek; Tomáš Martinec; Michal Petruš

Calculation of industrial robot trajectory in frame composite production

In: Jan Chleboun and Pavel Kůs and Petr Příklad and Karel Segeth and Jakub Šístek and Tomáš Vejchodský (eds.): Programs and Algorithms of Numerical Mathematics, Proceedings of Seminar. Janov nad Nisou, June 19-24, 2016. Institute of Mathematics CAS, Prague, 2017. pp. 81–88.

Persistent URL: <http://dml.cz/dmlcz/703001>

Terms of use:

© Institute of Mathematics CAS, 2017

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library*
<http://dml.cz>

CALCULATION OF INDUSTRIAL ROBOT TRAJECTORY IN FRAME COMPOSITE PRODUCTION

Jaroslav Mlýnek¹, Tomáš Martinec², Michal Petru²

¹ Department of Mathematics and Didactics of Mathematics, FP
jaroslav.mlynek@tul.cz

² Institute for Nanomaterials, Advanced Technologies and Innovations
tomas.martinec@tul.cz, michal.petru@tul.cz
Technical University of Liberec
Studentská 2, 461 17 Liberec, Czech Republic

Abstract: This article is focused on calculating the trajectory of an industrial robot in the production of composites for the automotive industry. The production technology is based on the winding of carbon fibres on a polyurethane frame. The frame is fastened to the end-effector of the robot arm (i.e. robot-end-effector, REE). The passage of the frame through the fibre processing head is determined by the REE trajectory. The position of the fibre processing head is fixed and is composed of three fibre guide wheels with coils of carbon fibres. The fibre processing head winds three layers of filaments onto the frame. The polyurethane frame is determined by the local Euclidean coordinate system E_3 , which has its origin in the REE. We use a mathematical model and matrix calculus to compute the trajectory of the REE to guarantee the desired passage of the frame through the fibre processing head. The translation and rotation matrices of the local coordinate system (of the REE) are calculated with respect to the base coordinate system of the robot.

Keywords: robot trajectory, transformation matrix, Euler angle of rotation, orthogonal group

MSC: 14P10, 15A04, 15A24

1. Introduction

Composite materials are extensively used in many branches of industry. These materials successfully replace traditional materials. The technology based on the winding of carbon fibres by an industrial robot on a polyurethane frame is now widespread in the manufacturing of composites. After the winding process, the composite is thermally hardened.

The fibre processing head is fixedly placed in the workspace of the industrial robot and its coordinates are specified in the base coordinate system of the robot.

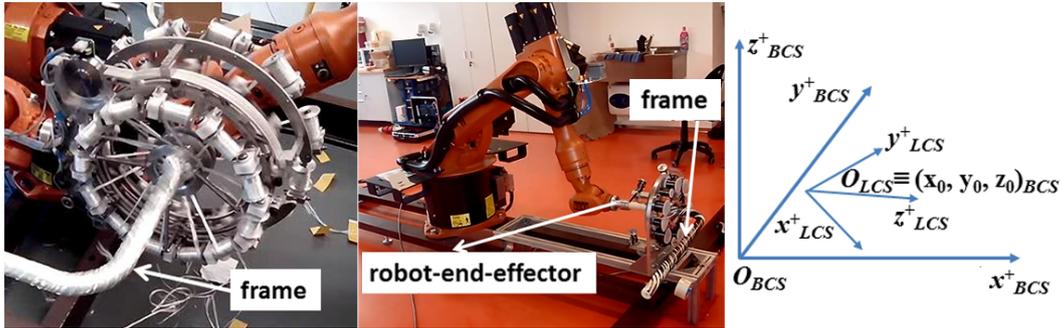


Figure 1: a) The fibre processing head with the three guide wheels for the filament winding of three layers on the frame. b) The robot with the frame attached to the REE and fibre processing head only with one fibre guide wheel. c) The base coordinate robot system and the local coordinate system of the REE.

The fibre processing head contains three fibre guide wheels, each of which includes twelve fixed coils along the periphery (see Fig. 1a). The outer fibre guide wheels rotate around their common axis, the central fibre guide wheel is static. The fibre processing head winds gradually three layers of filaments at the angles of 45° , 0° and -45° on the frame during its passage through the fibre processing head (see Fig. 1a). The frame is attached to the REE (see Fig. 1b). The passage of the frame through the fibre processing head is controlled by the movement of the REE.

Industrial robots suppliers often offer commercial software modules to control the robots. These modules are used in areas such as welding, pressing, cutting and packaging. However, the available software tools are not suitable for our purposes.

2. Mathematical model

A mathematical model of the passage of the frame through the fibre processing head developed in order to calculate the REE trajectory is described in this chapter. Using the robot base right-hand Euclidean coordinate system E_3 (BCS), we will describe the REE movements and rotations during the passage of the frame through the fibre processing head. The local right-hand Euclidean coordinate system E_3 (LCS) of the REE (see Fig. 1c) is also taken into account. To avoid any confusion, the points and vectors with the coordinates in the BCS are labelled with the subscript $_{BCS}$, while the points and vectors with the coordinates in the LCS are labelled with the subscript $_{LCS}$.

2.1. Industrial robot

The origin of the LCS is positioned in the REE. The actual position of the LCS with respect to the BCS is determined by the tool-centre-point (TCP). The robot central unit controls the movement of the REE while using the current TCP . The TCP contains six values $TCP = (x, y, z, a, b, c)$. The first three parameters

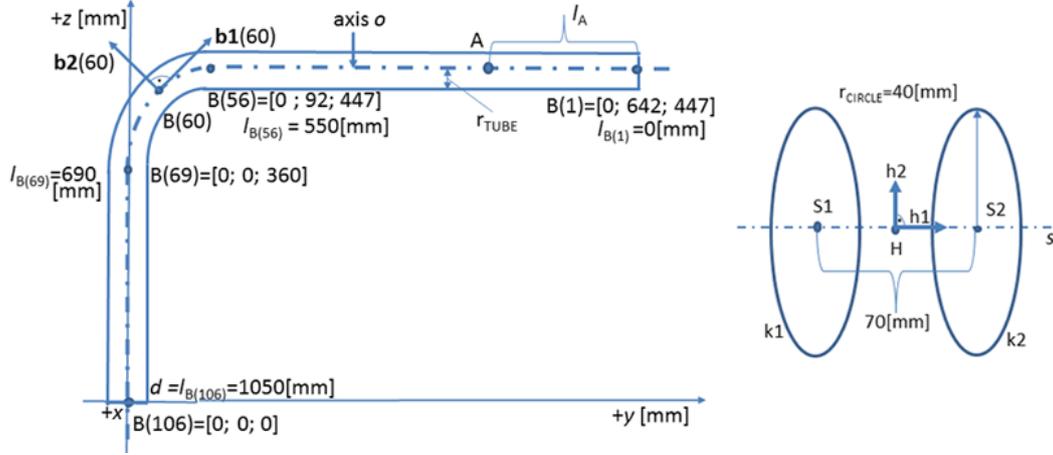


Figure 2: a) Example of vertical section through a polyurethane frame (see Fig. 1b) in LCS . The frame is connected to the REE in the point $B(106)$, $N = 106$. b) Model of the fibre processing head in BCS .

specify the coordinates of the origin of the LCS with respect to the BCS . The parameters a , b and c indicate the angles of the rotations of the LCS around the z , y and x axes with respect to the BCS .

2.2. Composite frame with a circular cross section

The polyurethane frame is described by its central axis o and its radius r_{TUBE} (see Fig. 2a). The central axis o is defined in the LCS of the REE through a discrete set of points $B(i)_{LCS}$ and the unit tangent vectors $\mathbf{b1}(i)_{LCS}$ at that points, $1 \leq i \leq N$. The initial point $B(1)_{LCS}$ and the end point $B(N)_{LCS}$ can coincide in the case of a closed frame. In addition, the unit vector $\mathbf{b2}(i)_{LCS}$ ($1 \leq i \leq N$) lies in the plane orthogonal to the vector $\mathbf{b1}(i)_{LCS}$ and defines the upward direction at the moment when the point $B(i)_{BCS}$ passes through the fibre processing head.

The points $B(i)_{BCS}$ and vectors $\mathbf{b1}(i)_{LCS}$, $\mathbf{b2}(i)_{LCS}$ are prescribed by a composite designer to ensure an optimal passage of the frame through the centre of the fibre processing head. We assume that the discrete set of points $B(i)_{LCS}$ defines the shape of the frame with a sufficient accuracy. If one or more parts of the axis o are line segments, then it is sufficient to define only the end points of these line segments. The variable l represents the distance between the point $B(1)_{LCS}$ and a point on the axis o . The distance is measured as the o -arc length (see Fig. 2a, a general point A and distance l_A).

2.3. Fibre processing head

The position of the fibre processing head is supposed to be fixed. It consists of three fibre guide wheels which wind three layers of fibres on the frame under angles of 45° , 0° and -45° . The outside rotating wheels are indicated in Fig. 2b as circles $k1$

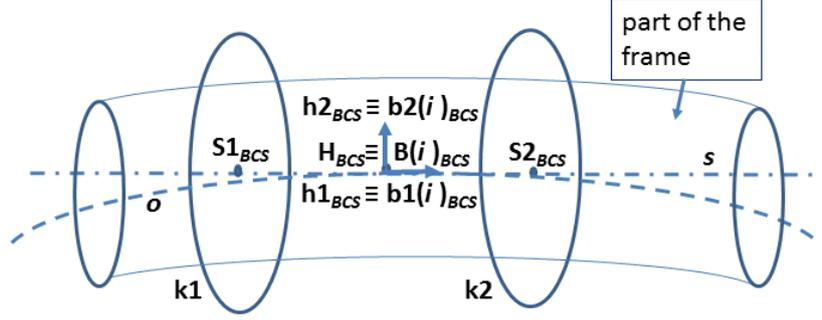


Figure 3: The scheme of the passing frame through the fibre processing head in the i -th step.

and $k2$ with centres $S1_{BCS}$ and $S2_{BCS}$, respectively. Both $k1$ and $k2$ have the same radius $r_{CIRCLE} > r_{TUBE}$ and their centres lie on the axis s of the head. The static middle fibre guide wheel is not important for our model. The best results can be achieved if the frame central axis o passes the head through its central point H_{BCS} and the tangent vector $\mathbf{b1}(i)_{BCS}$ ($1 \leq i \leq N$) of the axis o is aligned with the axis s of the head, represented by a unit vector $\mathbf{h1}_{BCS}$. The longitudinal rotation of the frame is governed by the angle of the vectors $\mathbf{h2}_{BCS} = (0, 0, 1, 0)^T$ and $\mathbf{b2}(i)_{BCS}$ ($1 \leq i \leq N$). Thus, each point of the frame central axis o should pass through H_{BCS} . The desired orientation of the frame is then given by the vectors $\mathbf{h1}_{BCS}$ and $\mathbf{h2}_{BCS}$.

3. Calculation of the trajectory

The main idea of calculating the REE trajectory is described in this chapter. We remind that frame is fixed to the REE. The goal is to calculate the REE trajectory that ensures a gradual passage of the axis o through the centre H_{BCS} of the head in the desired direction $\mathbf{h1}_{BCS}$. The frame's initial point of passage is $B(1)_{LCS}$ and the end point is $B(N)_{LCS}$. The REE trajectory is determined by the sequence of the TCP_i values, where $1 \leq i \leq N$. The initial position of the REE corresponds to the value TCP_0 .

In the admissible REE position, the two orthogonal vectors and their common initial point originally defined in the LCS are in the same position in the BCS as the two fixed orthogonal vectors and their common initial point specified in the BCS (see Fig. 3); therefore

$$H_{BCS} \equiv B(i)_{BCS}, \mathbf{h1}_{BCS} \equiv \mathbf{b1}(i)_{BCS}, \mathbf{h2}_{BCS} \equiv \mathbf{b2}(i)_{BCS}. \quad (1)$$

The BCS position and orientation of the REE in the i -th step of the passing of the frame through the fibre processing head are uniquely determined by the relation (1).

The central unit of the robot changes the position of the REE on the basis of the actual values of the TCP . The movement of the REE occurs in the transition from the TCP_{i-1} to the TCP_i , where the linear or cubic interpolation of the control parameters is applied.

3.1. Procedure of the TCP_i calculation

In this part we focus on the calculation of TCP_i , where $1 \leq i \leq N$. Points, vectors and matrices are represented in a homogeneous form (i.e. general point $A = (x_A, y_A, z_A, 1)^T$, vector $\mathbf{a} = (x_a, y_a, z_a, 0)^T$, this form is suitable for differentiation of operations with points and with vectors). The Euclidean norm of vector is further also used.

We calculate transformation matrix \mathbf{T}_i from LCS to BCS for the i -th step of passing the frame through the fibre processing head. The transformation matrix \mathbf{T}_i is generally the product of the translation matrix \mathbf{L}_i and the rotation matrix \mathbf{Q}_i , i.e.

$$\mathbf{T}_i = \mathbf{L}_i \cdot \mathbf{Q}_i . \quad (2)$$

After calculating the rotation matrix \mathbf{Q}_i we can decompose \mathbf{Q}_i and determine the Euler angles of the LCS rotations with respect to BCS . The knowledge of the translation matrix \mathbf{L}_i and the Euler angles leads to the determining of the TCP_i .

3.1.1. Determination of the rotation matrix Q_i in the relation (2)

We determine the rotation matrix Q_i ensuring the validity of the last two identifications in the relation (1).

To determine \mathbf{Q}_i we suppose temporarily that the origins of the BCS and LCS are identical and that $TCP_{i-1} = (x_{i-1}, y_{i-1}, z_{i-1}, a_{i-1}, b_{i-1}, c_{i-1})$ is specified. The matrix \mathbf{Q}_{i-1} is defined by the relation (see [4], p. 31)

$$\mathbf{Q}_{i-1} = \mathbf{Rot}(z, a_{i-1}) \cdot \mathbf{Rot}(y, b_{i-1}) \cdot \mathbf{Rot}(x, c_{i-1}),$$

where $\mathbf{Rot}(z, a_{i-1})$ is the orthogonal rotation matrix of the LCS around the axis z by angle a_{i-1} and similarly for the orthogonal matrices $\mathbf{Rot}(y, b_{i-1})$ and $\mathbf{Rot}(x, c_{i-1})$. These three rotation matrices are

$$\begin{pmatrix} ca & -sa & 0 & 0 \\ sa & ca & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} cb & 0 & sb & 0 \\ 0 & 1 & 0 & 0 \\ -sb & 0 & cb & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & cc & -sc & 0 \\ 0 & sc & cc & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (3)$$

where sa and ca indicate $\sin a_{i-1}$, $\cos a_{i-1}$ and similarly for cb , sb , cc , sc .

Subsequently, we perform the following steps.

1) The vector $\mathbf{b1}(i)_{LCS}$ is expressed in the BCS as $\mathbf{b1}(i)_{BCS} = \mathbf{Q}_{i-1} \cdot \mathbf{b1}(i)_{LCS}$. The deviation α of the vectors $\mathbf{h1}_{BCS}$ and $\mathbf{b1}(i)_{BCS}$ is determined by using their scalar product.

2) We then calculate the cross product $\mathbf{n}_{BCS} = \mathbf{h1}_{BCS} \times \mathbf{b1}(i)_{BCS}$. The vector \mathbf{n}_{BCS} is orthogonal to both the vectors $\mathbf{h1}_{BCS}$ and $\mathbf{b1}(i)_{BCS}$. The vector \mathbf{n}_{BCS} is normalized, i.e. $\mathbf{n}_{BCS} = \mathbf{n}_{BCS} / \|\mathbf{n}_{BCS}\|$.

3) Now, the vector $\mathbf{b1}(i)_{BCS}$ is rotated by the angle α around the vector \mathbf{n}_{BCS} to vector $\mathbf{h1}_{BCS}$ (after rotation, the vector $\mathbf{b1}(i)_{BCS}$ coincides with the vector $\mathbf{h1}_{BCS}$). If we denote the components of the unit vector $\mathbf{n}_{BCS} = (n_1, n_2, n_3, 0)^T$ then the matrix $\mathbf{Rot}(\mathbf{n}_{BCS}, \alpha)$ is of the form (see [4], p. 34)

$$\mathbf{Rot}(\mathbf{n}_{BCS}, \alpha) = \begin{pmatrix} c + n_1^2(1-c) & n_1n_2(1-c) - n_3s & n_1n_3(1-c) + n_2s & 0 \\ n_1n_2(1-c) + n_3s & c + n_2^2(1-c) & n_2n_3(1-c) - n_1s & 0 \\ n_1n_3(1-c) - n_2s & n_2n_3(1-c) + n_1s & c + n_3^2(1-c) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (4)$$

where s and c denote $s = \sin \alpha$, $c = \cos \alpha$. Then it is true that

$$\mathbf{h1}_{BCS} \equiv \mathbf{b1}(i)_{BCS} := \mathbf{Rot}(\mathbf{n}_{BCS}, \alpha) \cdot \mathbf{b1}(i)_{BCS} = \mathbf{Rot}(\mathbf{n}_{BCS}, \alpha) \cdot \mathbf{Q}_{i-1} \cdot \mathbf{b1}(i)_{LCS}.$$

At the same time the vector $\mathbf{l}_{BCS} = \mathbf{Rot}(\mathbf{n}_{BCS}, \alpha) \cdot \mathbf{Q}_{i-1} \cdot \mathbf{b2}(i)_{LCS}$ is calculated.

4) The deviation β of the vectors $\mathbf{h2}_{BCS}$ and \mathbf{l}_{BCS} is determined by using their scalar product. We define the rotation matrix $\mathbf{Rot}(\mathbf{h1}_{BCS}, \beta)$ describing the rotation of the vector \mathbf{l}_{BCS} around $\mathbf{h1}_{BCS}$ to $\mathbf{h2}_{BCS}$. Then it is true

$$\mathbf{h2}_{BCS} \equiv \mathbf{b2}(i)_{BCS} := \mathbf{Rot}(\mathbf{h1}_{BCS}, \beta) \cdot \mathbf{Rot}(\mathbf{n}_{BCS}, \alpha) \cdot \mathbf{Q}_{i-1} \cdot \mathbf{b2}(i)_{LCS}.$$

The resulting rotation matrix

$$\mathbf{Q}_i = \mathbf{Rot}(\mathbf{h1}_{BCS}, \beta) \cdot \mathbf{Rot}(\mathbf{n}_{BCS}, \alpha) \cdot \mathbf{Rot}(z, a_{i-1}) \cdot \mathbf{Rot}(y, b_{i-1}) \cdot \mathbf{Rot}(x, c_{i-1}), \quad (5)$$

where the elements of the matrix $\mathbf{Rot}(\mathbf{h1}_{BCS}, \beta)$ are defined analogously as the elements of the matrix $\mathbf{Rot}(\mathbf{n}_{BCS}, \alpha)$ in (4). Now, it is true that $\mathbf{h1}_{BCS} \equiv \mathbf{b1}(i)_{BCS} = \mathbf{Q}_i \cdot \mathbf{b1}(i)_{LCS}$ and $\mathbf{h2}_{BCS} \equiv \mathbf{b2}(i)_{BCS} = \mathbf{Q}_i \cdot \mathbf{b2}(i)_{LCS}$ in the relation (1).

3.1.2. Calculation of the Euler angles

Any right-hand rotation of the Euclidean space E_3 around a given unit vector \mathbf{v} is determined by the orthogonal matrix $\mathbf{Q} = \mathbf{Rot}(\mathbf{v}, \vartheta)$, where ϑ is the angle of rotation. It is true that $\det(\mathbf{Q}) = 1$ and the elements of the matrix \mathbf{Q} are of the form (4). These rotation matrices create the orthogonal group $\text{SO}(3)$ (see [1]). Each rotation matrix \mathbf{Q} can be written in the form (see [4], p. 32)

$$\mathbf{Q} = \mathbf{Rot}(z, a) \cdot \mathbf{Rot}(y, b) \cdot \mathbf{Rot}(x, c), \quad (6)$$

where the matrices $\mathbf{Rot}(z, a)$, $\mathbf{Rot}(y, b)$, and $\mathbf{Rot}(x, c)$ are the orthogonal matrices of rotations around the axes z , y , x and are of the form (3). Values a , b , and c are the corresponding Euler angles. We note that the Euler angles are not uniquely determined by relation (6) (see [3]).

Now, we describe the procedure to determine the Euler angles entering the matrix \mathbf{Q}_i in (5). The rotation matrix \mathbf{Q}_i can be decomposed in accordance with (6) in the form $\mathbf{Q}_i = \mathbf{Rot}(z, a_i) \cdot \mathbf{Rot}(y, b_i) \cdot \mathbf{Rot}(x, c_i)$. Both sides of the equation (6) written for the matrix \mathbf{Q}_i are multiplied by the matrix $\mathbf{Rot}(z, a_i)^T$. The matrix $\mathbf{Rot}(z, a_i)$ is orthogonal, therefore $\mathbf{Rot}(z, a_i)^{-1} = \mathbf{Rot}(z, a_i)^T$. The equation (6) then reads

$$\mathbf{Rot}(z, a_i)^T \cdot \mathbf{Q}_i = \mathbf{Rot}(y, b_i) \cdot \mathbf{Rot}(x, c_i). \quad (7)$$

The rotation angles a_i , b_i and c_i are calculated by comparing suitably selected corresponding elements from the matrix product on the left and right side of equation (7). By writing the rotation matrix \mathbf{Q}_i in the form

$$\mathbf{Q}_i = \begin{pmatrix} q_{11}(i) & q_{12}(i) & q_{13}(i) & 0 \\ q_{21}(i) & q_{22}(i) & q_{23}(i) & 0 \\ q_{31}(i) & q_{32}(i) & q_{33}(i) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (8)$$

we obtain Euler angles a_i , b_i , and c_i by the following expressions

$$\begin{aligned} a_i &= \text{ATAN2}(q_{21}(i), q_{11}(i)), \\ b_i &= \text{ATAN2}(-q_{31}(i), q_{11}(i) \cdot \cos a_i + q_{21}(i) \cdot \sin a_i), \\ c_i &= \text{ATAN2}(q_{13}(i) \cdot \sin a_i - q_{23}(i) \cdot \cos a_i, q_{22}(i) \cdot \cos a_i - q_{12}(i) \cdot \sin a_i). \end{aligned} \quad (9)$$

The $\text{ATAN2}(arg1, arg2)$ function (common in many programming languages) calculates the value of the arctangent function for the argument $arg1/arg2$. The signs of both input parameters are involved in the determining of the output angle of the ATAN2 function ($-\pi < \text{ATAN2}(arg1, arg2) \leq \pi$).

3.1.3. Determination of the translation matrix \mathbf{L}_i in the relation (2)

In general, the origin of the BCS and the origin of the LCS have different positions. We have to translate the LCS relative to the BCS so that $B(i)_{BCS} \equiv H_{BCS}$. We determine the translation vector $\mathbf{u}(i)_{BCS}$ as follows

$$\mathbf{u}(i)_{BCS} := H_{BCS} - \mathbf{Q}(i) \cdot B(i)_{LCS} - (x_{i-1}, y_{i-1}, z_{i-1}, 0)^T, \quad (10)$$

where $\mathbf{Q}(i)$ is given by (5) and x_{i-1} , y_{i-1} , and z_{i-1} are the first three parameters of TCP_{i-1} . Then, see (2),

$$\mathbf{L}_i = \begin{pmatrix} 1 & 0 & 0 & x_{\mathbf{u}(i)} \\ 0 & 1 & 0 & y_{\mathbf{u}(i)} \\ 0 & 0 & 1 & z_{\mathbf{u}(i)} \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

where $x_{\mathbf{u}(i)}$, $y_{\mathbf{u}(i)}$, and $z_{\mathbf{u}(i)}$ are the components of the vector $\mathbf{u}(i)_{BCS}$, see (10).

The REE is in the admissible position (1) after the transformation (2) of the LCS of the REE.

The described procedure allows to determine $TCP_i = (x_{\mathbf{u}(i)}, y_{\mathbf{u}(i)}, z_{\mathbf{u}(i)}, a(i), b(i), c(i))$ for $1 \leq i \leq N$ and thereby the whole REE trajectory when the frame passes through the fibre processing head.

4. Conclusion

The algorithm described in Chapter 3 calculates the 3D trajectory of the REE of an industrial robot during the production of composites using fibre winding technology. The described algorithm allows to determine the exact trajectory of the REE, which provides a significant advantage over the users of the extended teach-in principle (technician searches for a suitable trajectory using the robot control panel — teach pendant). Also, the possibility to accurately determine the desired trajectory of the REE by the presented algorithm can be beneficial for optimizing the REE trajectory. The application of the algorithm is completely independent of the type of robot and software tools.

The use of the procedure for determining the trajectory of the REE does not increase production costs and can significantly speed up the determination of the robot trajectory.

The practical results of the robot trajectory calculation during the winding process of composite production are described in [2].

Acknowledgements

This work was supported by grant No. TF02000051 of the Institute for Nanomaterials, Advanced Technologies and Innovations, Technical University of Liberec.

References

- [1] Baker, A.: *Matrix Groups: An introduction to lie group theory*. Springer-Verlag, London, 2002.
- [2] Martinec, T., Mlýnek, J., and Petrů, M.: Calculation of the robot trajectory for the optimum directional orientation of fibre placement in the manufacture of composite profile frames. In: *Robotics and Computer-Integrated Manufacturing*, vol. 35 (2015), 42–54.
- [3] Salabough, G.G.: Computing Euler angles from a rotation matrix. www.soi.city.ac.uk/sbbh635/publications/euler.pdf (1999).
- [4] Sciavicco, L. and Siciliano, B.: *Modelling and control of robot manipulators*. Springer-Verlag, London, 2004.