



A SOFTWARE APPLICATION INTENDED FOR THE NUMERICAL MODELING OF SERIAL ROBOT STRUCTURES

Adina Veronica DUCA, Radu MORARIU GLIGOR

Abstract: Within this paper are evidenced some important aspects regarding the functionalities of a software application developed for the numerical modeling of any serial robot mechanical structures. The main functions of the presented software application are described in detail and, for exemplification, are determined the forward geometry equations in case of a five degrees of freedom serial robot. The software application also enables the tridimensional modeling of the robot kinematical structure by means of interfacing with the SolidWorks 2010 software. Based on this, were obtained parameterized tridimensional models corresponding to each element from the robot kinematical structure.

Key words: serial robot, forward geometry, 3D modeling, software application, numerical modeling

1. INTRODUCTION

Within this paper are presented some important researches and achievements in the field of numerical and experimental modeling of mechanical robot structures. It is presented a software application developed for geometrical modeling, based on nominal values, of any robot structure. Forward, is presented the software application, which is used to generate the forward geometric model as well as the tridimensional model of the kinematic structure for a five degrees of freedom robot, first, based on the algorithm of locating matrices and the latter by using Solid Works 2010.

2. GENERAL PRESENTATION OF THE SOFTWARE APPLICATION

Forward, is presented a numerical application based on the developed software which is intended for validating the mathematical model of determining the geometrical errors for position and orientation. To this effect, was considered the mechanical structure of 5 d.o.f's robot (FANUC LR Mate 100iB) on which was developed a study on the geometrical errors that affect the accuracy of position and orientation, by considering a

known robot configuration $\bar{\theta}_k$ (according to Fig.1).

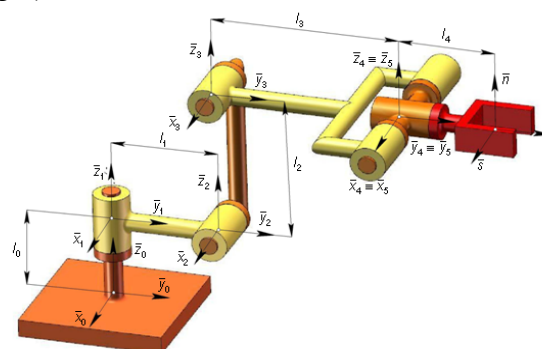


Fig. 2 The kinematical structure of the robot

In the Figure 1 is presented the kinematical structure of the 5 d.o.f's robot subjected to study, which consists of five rotational joints whose position and orientation in case of the nominal configuration is presented in Table1.

The matrix of nominal values Table 1

Element 1 → 6	Joint {R; T}	Position			Orientation		
		$\bar{p}_i^{(0)T}$			$\bar{k}_i^{(0)T}$		
1	R	0	0	0	0	0	1
2	R	150	0	0	0	1	0
3	R	150	0	250	0	1	0
4	R	370	0	250	0	1	0
5	R	370	0	250	1	0	0
6	-	595	0	250	0	1	0

Hereinafter, according to mathematical models presented within [1] and [2] as well as the kinematical schemes from [3], have been developed a software application for determining the equations of the forward geometrical model, for any robot mechanical structure. The main features and functionalities of the developed software application are presented in detail in the following paragraph.

In case of the robot mechanical structure selected for illustrating the functionality of the software application, have been developed the geometrical model for both, nominal and real configurations (the latter affected by errors).

The results were forwards used for determining the transfer matrix of the geometrical errors charactering the given robot configuration.

3. THE GEOMETRICAL MODEL OF 5R ROBOT USING SOFTWARE APPLICATION

For determining the equations of the geometrical model characterizing the nominal configuration of the 5R robot, have been developed a software application based on Visual Basic 6.0 programming language. The application enables the tridimensional modeling of the robot kinematical structure by means of interfacing with the SolidWorks 2010 software.

Based on this, were obtained parameterized tridimensional models corresponding to each element from the robot kinematical structure.

The developed software is able, on the basis of the functionalities offered by SolidWorks 2010 application to bring up-to-date the dimensions of the tridimensional models, according to the new requirements imposed by user.

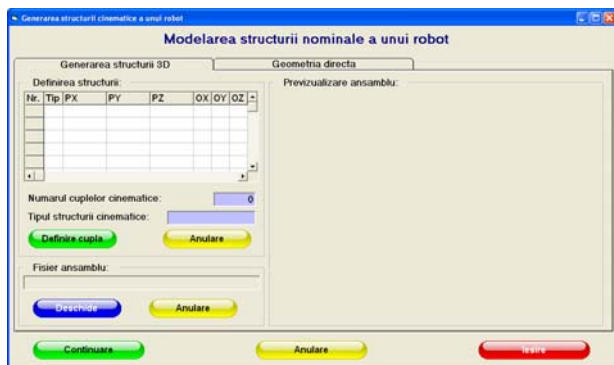


Fig. 2 The interface of the software developed for numerical modeling of serial robot structures

The modality employed in designing the tridimensional elements of the robot components takes into consideration their spatial orientation.

For this reason, in designing the assembly, the components are very easy to be placed on their positions because all the elements are accurately oriented and the insertion points are given by the coordinates of geometrical center corresponding to each kinematical joint. The application is build by considering two important components: the first component which allows the tridimensional design of the kinematical structure of the robot, meanwhile the second component is intended for the calculus of the orientation matrices as for the final location of the end-effector (final position and orientation) within the work space.

In the following is going to be presented an application which illustrates the main steps in building a kinematical structure and in determining the orientation matrices and the final position and orientation of the end-effector.

The mechanical structure subjected to study is a 5R type robot, previously presented in Fig. 1.

The first step in using the application consists in defining using SolidWorks an assembly type folder (having the *.sldasm extension).

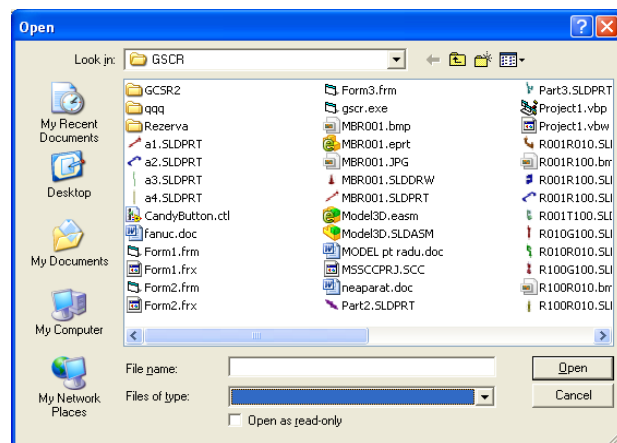


Fig. 3 The SolidWorks folder opening window

By pressing the button “Define joint”, is generated the basis module which can be either rotation or translation module.

In the presented example the basis module corresponds to a rotation module. In the window that defines the joint type is also selected the type of the created module, the initial orientation of the kinematical axis, as well as the coordinates of the geometrical center of the defined kinematical joint (as it results from Fig. 4).

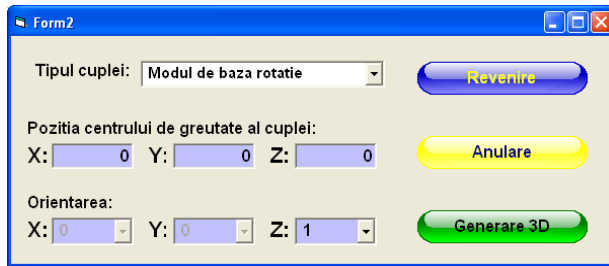


Fig. 5 The window for defining the locating for every kinematical joint

All this data is comprised in the matrix of nominal values, previously defined with Table 1.

Based on the tridimensional model corresponding to the basis rotation module, forward is created a "Part" folder (*.sldprt) – according to Fig. 5. The created component is saved under a name established by the user.

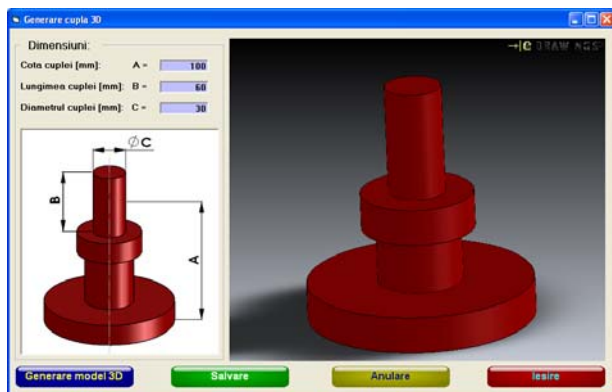


Fig. 5 The window for generating the 3D structure

By pressing the "Exit" button, the user returns to the main window. The previously created folder is added to the final assembly by keeping the spatial orientation of the elements as well as the coordinates of the geometrical center of the kinematical joint. The values established by the user in creating the tridimensional model of the first element are automatically added to the matrix of nominal values (such as in Fig. 6).

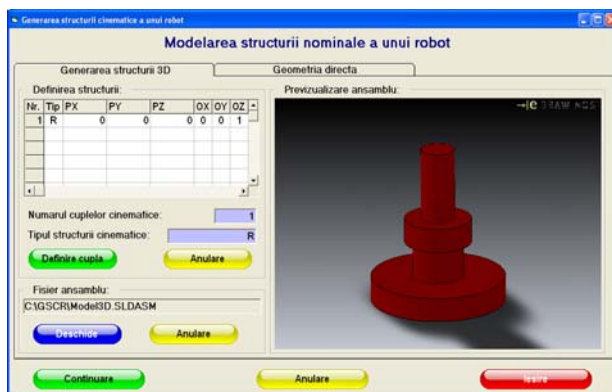


Fig. 6 Returning to the main window

The next step consists in creating the tridimensional model of the next element from the kinematical structure using "Define joint".



Fig. 7 Defining the following kinematical joint

This opens a window that allows the user to select the type of the kinematical joint (rotation/translation) and also to establish its position and orientation (according to Fig. 7).

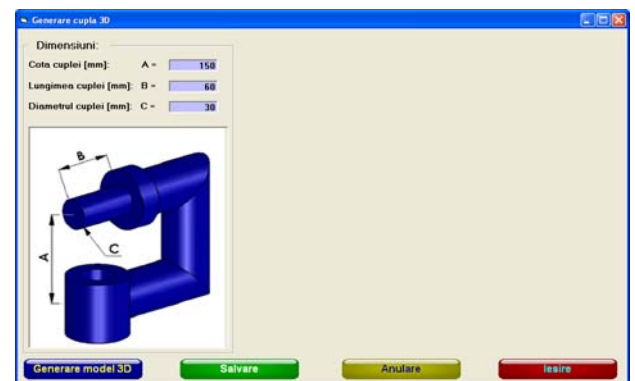


Fig. 8 The window for defining the kinematical joint

By pressing the button "3D structure generation" on the screen will appear the corresponding window, where into the "Dimensions" field the user can visualize the implicit values for the kinematical element dimensions along with a picture of the kinematical joint (Fig. 8). Using the "3D model generation" function, it results the tridimensional model of the defined element (Fig. 9).

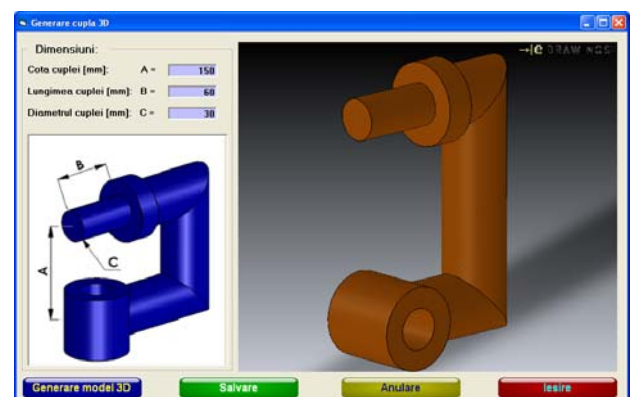


Fig. 9 Generating the 3D model of the kinematical joint

The “Exit” button brings user back to the window for defining the kinematical joint. To return to the main window, the “Back” button has to be pressed. In the main window, the tridimensional model of the robot is updated by adding the newly defined kinematical element. The matrix containing the locating of the first joint is also filled out with the new data. (Fig. 10).

Following the steps previously described, are created all the elements that comprise the kinematical structure of the 5R robot type.



Fig. 10 Generating the provisory 3D model of the robot

The updating of the entire assembly is made, manually, outside the application by using the functions of the SolidWorks software. (Fig.11) The final kinematical structure of the 5R type robot is presented in the following window:

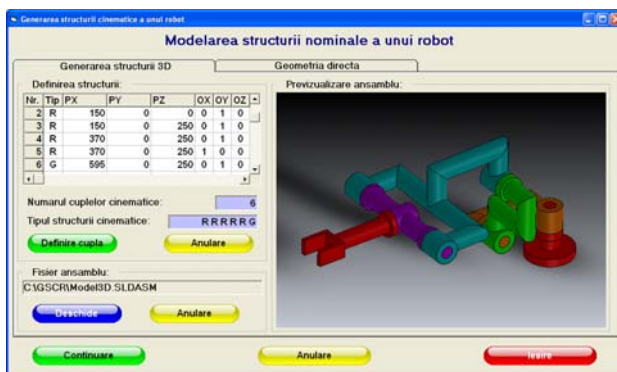


Fig. 11 Generating the 3D model of the robot

In the next phase, according to the mathematical models presented in [4], [5] are determined the nominal values for the position and orientation of the end-effector.

In order to achieve this, the generalized coordinates characterizing each joint have to be previously defined. This is made simply by inserting the values for the generalized

coordinates in the corresponding text field and then pressing OK button. (Fig.12).

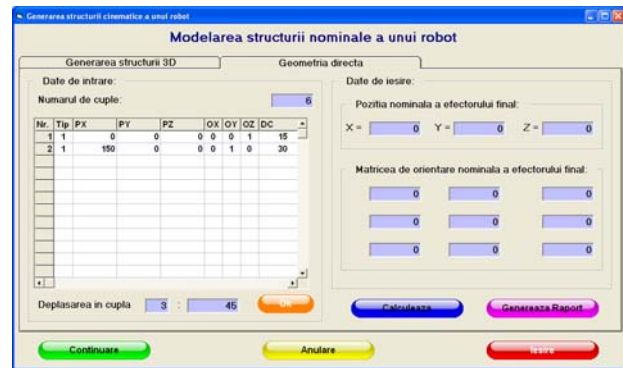


Fig. 12 Inserting the values of the generalized coordinates

After the inserting of the values corresponding to robot generalized coordinates, the homogenous matrices that comprise the position and orientation of each joint, respectively of the end – effector, are obtained by pressing the button “Calculate”. The final results are presented in a table format, such as:

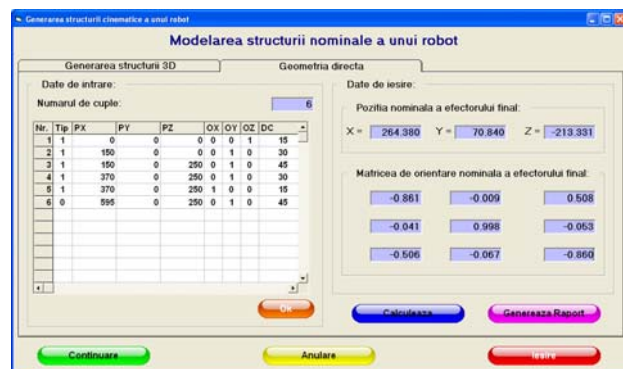


Fig. 13 The final results window

Another function of the developed application is the possibility of generating a report containing the nominal values for the position and orientation of the end – effector.

By pressing the “Generate Report” button, a detailed report will be automatically generated in *.doc format. The results are forward used as input data in determining the transfer matrix for the geometrical errors, for a given configuration.

4. THE TRANSFER MATRIX BASED ON LOCATING MATRICES

The transfer matrices for geometrical errors (position and orientation) of the end-effector based on the locating parameters can be

determined, according to [6] and [7], starting from the following matrix expression:

$$\delta^0 T_n = \sum_{i=1}^n {}^0_i[T] \cdot \delta^i T_{i-1} \cdot {}^0_i[T]^{-1} \quad (1.1)$$

By applying some differential transformations on the matrix expression (1.1) it results:

$$\delta^i T_{i-1} = \left[\begin{array}{c|c} \{\Delta \bar{\psi}_i \times\} & \Delta \bar{p}_{i-1} \\ \hline 0 & 1 \end{array} \right]; \quad (1.2)$$

where (1.2) is the error differential operator. Within the expression (1.1), presented above, by substituting the expression of the error differential operator with (1.2), it is obtained:

$$\delta^0 T_n = \sum_{i=1}^n {}^0_i[T] \cdot \delta^i T_{i-1} \cdot {}^0_i[T]^{-1}, \quad (1.3)$$

$$\delta^0 T_n = {}^0_i[T] \cdot \left[\begin{array}{c|c} \{\Delta \bar{\psi}_i \times\} & \Delta \bar{p}_{i-1} \\ \hline 0 & 1 \end{array} \right] \cdot {}^0_i[T]^{-1}; \quad (1.4)$$

where,
$${}^0_i[T] = \left[\begin{array}{c|c} {}^0_i[R] & \bar{p}_i \\ \hline 0 & 1 \end{array} \right]; \quad (1.5)$$

$${}^0_i[T]^{-1} = \left[\begin{array}{c|c} {}^0_i[R]^{-1} & -{}^0_i[R]^{-1} \cdot \bar{p}_i \\ \hline 0 & 1 \end{array} \right]; \quad (1.6)$$

$$\delta^0 T_n = \left[\begin{array}{c|c} A & B \\ \hline 0 & 0 \end{array} \right]; \quad (1.7)$$

where,

$$A = {}^0_i[R] \cdot \{\Delta \bar{\psi}_i \times\} \cdot {}^0_i[R]^{-1} \quad (1.8)$$

$$B = {}^0_i[R] \cdot \Delta \bar{p}_{i-1} - {}^0_i[R] \cdot \{\Delta \bar{\psi}_i \times\} \cdot {}^0_i[R]^{-1} \cdot \bar{p}_i \quad (1.9)$$

According to [7] it is considered the notation:

$$\bar{\varepsilon} = [\Delta \bar{p}_{i-1} \quad \Delta \bar{\psi}_{i-1}]. \quad (1.10)$$

where $\bar{\varepsilon}$ represents the vector of position and orientation errors. Separating the position errors $\Delta \bar{p}_{i-1}$ and $\Delta \bar{\psi}_{i-1}$ it is obtained a matrix expression that marks out the mathematical connection between a 6x6n matrix, symbolized $E_{d\delta}$, also known as transfer matrix and the

vector of geometrical errors, $[\bar{d}^T \quad \bar{\delta}^T]^T$.

The general form of this matrix can be determined based on the following equations:

$${}^0_i[R] \cdot \{\Delta \bar{\psi}_i \times\} \cdot {}^0_i[R]^{-1} = \begin{bmatrix} 0 & -\delta_{iz} & \delta_{iy} \\ \delta_{iz} & 0 & -\delta_{ix} \\ -\delta_{iy} & \delta_{ix} & 0 \end{bmatrix}; \quad (1.11)$$

where ${}^0_i[R]$ and ${}^0_i[R]^{-1}$ represents the rotation matrix and the corresponding inverse, defined as:

$${}^0_i[R] = \begin{bmatrix} \alpha_{ix} & \alpha_{iy} & \alpha_{iz} \\ \beta_{ix} & \beta_{iy} & \beta_{iz} \\ \gamma_{ix} & \gamma_{iy} & \gamma_{iz} \end{bmatrix}; {}^0_i[R]^{-1} = \begin{bmatrix} \alpha_{ix} & \beta_{ix} & \gamma_{ix} \\ \alpha_{iy} & \beta_{iy} & \gamma_{iy} \\ \alpha_{iz} & \beta_{iz} & \gamma_{iz} \end{bmatrix} \quad (1.12)$$

Computing the product defined with (1.11) are obtained the expressions that characterize each component of the resultant matrix, such as presented in the following:

$$\delta_{ix} = \Delta \alpha_i \cdot (\gamma_{iz} \cdot \beta_{iy} - \gamma_{iy} \cdot \beta_{iz}) + \Delta \beta_i (\gamma_{ix} \cdot \beta_{iz} - \gamma_{iz} \cdot \beta_{ix}) + \Delta \gamma_i (\gamma_{iy} \cdot \beta_{ix} - \gamma_{ix} \cdot \beta_{iy}) \quad (1.13)$$

$$\delta_{iy} = \Delta \alpha_i \cdot (\alpha_{iz} \cdot \gamma_{iy} - \alpha_{iy} \cdot \gamma_{iz}) + \Delta \beta_i (\alpha_{ix} \cdot \gamma_{iz} - \alpha_{iz} \cdot \gamma_{ix}) + \Delta \gamma_i (\alpha_{iy} \cdot \gamma_{ix} - \alpha_{ix} \cdot \gamma_{iy}) \quad (1.14)$$

$$\delta_{iz} = \Delta \alpha_i \cdot (\alpha_{iy} \cdot \beta_{iz} - \beta_{iy} \cdot \alpha_{iz}) + \Delta \beta_i (\beta_{ix} \cdot \alpha_{iz} - \beta_{iz} \cdot \alpha_{ix}) + \Delta \gamma_i (\alpha_{ix} \cdot \beta_{iz} - \alpha_{iz} \cdot \beta_{ix}) \quad (1.15)$$

The column vector of the orientation corresponding to i line from the final matrix, results from separating the coefficients answerable to the basic errors for orientation represented by $\Delta \alpha_i$, $\Delta \beta_i$ and $\Delta \gamma_i$, is:

$$\delta_i = [\delta_{ix} \quad \delta_{iy} \quad \delta_{iz}]^T; \quad (1.16)$$

The components of the column vector that contains the basic errors of position, \bar{d}_i results from computing the second matrix product defined with (1.9), such as presented bellow:

$${}^0_i[R] \cdot \Delta \bar{p}_i - {}^0_i[R] \cdot \Delta \bar{\psi}_i \times \bar{p}_i =; \quad (1.17)$$

$$= [d_{ix} \quad d_{iy} \quad d_{iz}]^T \equiv \bar{d}_i$$

where,

$$d_{ix} = \alpha_{ix} \cdot \Delta p_{xi} + \alpha_{iy} \cdot \Delta p_{yi} + \alpha_{iz} \cdot \Delta p_{zi} + \Delta \alpha_i \cdot (\alpha_{iy} \cdot p_{zi} - \alpha_{iz} \cdot p_{yi}) + \Delta \beta_i \cdot (\alpha_{iz} \cdot p_{xi} - \alpha_{ix} \cdot p_{zi}) + \Delta \gamma_i \cdot (\alpha_{ix} \cdot p_{yi} - \alpha_{iy} \cdot p_{xi}) \quad (1.18)$$

$$d_{iy} = \beta_{ix} \cdot \Delta p_{xi} + \beta_{iy} \cdot \Delta p_{yi} + \beta_{iz} \cdot \Delta p_{zi} + \Delta \alpha_i \cdot (\beta_{iy} \cdot p_{zi} - \beta_{iz} \cdot p_{yi}) + \Delta \beta_i \cdot (\beta_{iz} \cdot p_{xi} - \beta_{ix} \cdot p_{zi}) + \Delta \gamma_i \cdot (\beta_{ix} \cdot p_{yi} - \beta_{iy} \cdot p_{xi}) \quad (1.19)$$

$$d_{iz} = \gamma_{ix} \cdot \Delta p_{xi} + \gamma_{iy} \cdot \Delta p_{yi} + \gamma_{iz} \cdot \Delta p_{zi} + \Delta \alpha_i \cdot (\gamma_{iy} \cdot p_{zi} - \gamma_{iz} \cdot p_{yi}) + \Delta \beta_i \cdot (\gamma_{iz} \cdot p_{xi} - \gamma_{ix} \cdot p_{zi}) + \Delta \gamma_i \cdot (\gamma_{ix} \cdot p_{yi} - \gamma_{iy} \cdot p_{xi}) \quad (1.20)$$

where the expresions (1.18) - (1.20) characterize the cartesian components of the column vector \bar{d}_i .

The following expressions can be written:

$$[d_i \ \delta_i]^T = [d_{ix} \ d_{iy} \ d_{iz} \ \delta_{ix} \ \delta_{iy} \ \delta_{iz}]^T; \quad (1.21)$$

$$\varepsilon_{is} = [\Delta p_{xi} \ \Delta p_{yi} \ \Delta p_{zi} \ \Delta \alpha_i \ \Delta \beta_i \ \Delta \gamma_i]^T; \quad (1.22)$$

where (1.22) represents the vector of locating errors (basic errors of position and orientation)

The resultant error transfer matrix $E_{d,\delta}$ consists of n matrix blocks, each of them having the dimension (6×6) , as it results from:

$$E_{d,\delta} = \begin{bmatrix} E_{d,\delta}^1 & E_{d,\delta}^2 & \dots & E_{d,\delta}^i & \dots & E_{d,\delta}^n \end{bmatrix} \quad (1.23)$$

where $E_{d,\delta}^i$ is substituted by a matrix expression defined and presented within [6] and [7], resulting a $(6 \times 6n)$ matrix, such as:

$$\begin{aligned} [\bar{d}_i^T \ \bar{\delta}_i^T]^T &= E_{d,\delta}^i \cdot \bar{\varepsilon}_{is} = \\ &= [d_{ix} \ d_{iy} \ d_{iz} \ \delta_{ix} \ \delta_{iy} \ \delta_{iz}]^T \end{aligned} \quad (1.24)$$

The transfer matrices play an essential role in what concerns the forward and inverse modeling of position and orientation errors.

5. CONCLUSION

Within this paper have been presented the functionalities of a software application developed for determining the forward geometrical model equations for any robot

mechanical structure. Also, the program can be useful in generating the 3D model of the kinematical structure of the considered robot based on the SolidWorks CAD application. The purpose of developing this software application is to offer to the user a simplifying tool in performing the numerical modeling of nominal geometry for any robot mechanical structure.

The results offered by this software application are forwards used as input data in developing the kinematical and dynamical models for any robot structure subjected to study. The data provided by this application are also important in establishing the transfer matrix for the position and orientation errors.

6. REFERENCES

- [1] Craig, J.J., *Introduction to Robotics*, Addison-Wesley, Amsterdam, 1989.
- [2] Negrean, I., *Contribuții la optimizarea parametrilor cinematici și dinamici în vederea mării preciziei de funcționare a roboților*, Teză de doctorat, Cluj-Napoca, 1995.
- [3] Negrean, I., Vușcan, I., Haiduc, N., *Robotică. Modelarea cinematică și dinamică*, Editura Didactică și Pedagogică, București, 1997.
- [5] Negrean, I., Albeșel, D.G., *The Generalized Matrices in the Robot Accuracy*, Conferința științifică Internațională TMCR 2003, Chișinău, 2003, Vol.3, ISBN 9975-9748-3-X.
- [4] T., Kurfes, *Robotics and automation handbook*, Edited by Thomas R. Kurfess, CRC Press, 2005
- [6] Negrean, I., Duca, A.V., Negrean, D.C., K. Kacso, *Mecanica avansată în robotică*, Editura UT Press, Cluj-Napoca, 2008, ISBN 978-973-662-420-9.
- [7] Duca A.V., *Cercetări și contribuții privind modelarea preciziei roboților cu structură serială*, Teză de doctorat, Cluj-Napoca, Romania, 2012.

Dezvoltarea unei aplicații software destinată modelării numerice a structurilor seriale de roboți

În cadrul acestei lucrări sunt evidențiate o serie de aspect privind funcționarea unei aplicații software care a fost dezvoltată în scopul modelării numerice a structurilor mecanice de roboți seriali. În cadrul acestei lucrări sunt descrise în detaliu funcțiile de bază ale programului, iar pentru validare, sunt determinate ecuațiile geometriei directe (în formă numerică) ce caracterizează structura mecanică a unui robot serial cu cinci grade de libertate. Aplicația software permite de asemenea crearea modelului tridimensional a structurii cinematice de robot supusă analizei datorită interfațării cu softul SolidWorks 2010. Astfel, s-au obținut, modelele tridimensionale parametrizate corespunzătoare fiecărui element cinematic din alcătuirea structurii de robot.

Adina Veronica DUCA, Asist.Ph.D.Eng, Technical University of Cluj - Napoca, Department of Mechanical Systems Engineering, ducaadina@yahoo.com, 0264/401750, Câmpia Turzii, no1. Mureșului Street, 0364261867.

Radu MORARIU GLIGOR, S.L.Ph.D.Eng, Technical University of Cluj - Napoca, Department of Mechanical Systems Engineering, rmogli@yahoo.com@yahoo.com.