BULETINUL INSTITUTULUI POLITEHNIC DIN IAŞI
Publicat de
Universitatea Tehnică „Gheorghe Asachi” din Iaşi,
Tomul LX (LXIV), Fasc. 3, 2014
Secţia
CONSTRUCŢII DE MAŞINI

INTERFACE DESIGN FOR A SERIAL MANIPULATOR USING A VIRTUAL SIMULATION
I. PROGRAM DESCRIPTION

BY

GEORGETA CIOBĂNAŞU*

“Gheorghe Asachi” Technical University of Iaşi,
Department of Engineering Graphics

Received: November 14, 2014
Accepted for publication: November 24, 2014

Abstract. The paper focuses on the positional aspect of the sequentially linked elements of an open kinematic chain, constituting a serial manipulator. The factors that influence the performance of a robot are: dimensions, precision, profitability, number of free degrees, response time, weight, volume of workspace, capabilities of the control system, speed, working environment, transportable load and the existence of several working arms. The paper proposes a solution for testing and comparing robot arm trajectories, depending on the configuration of the structural parameters of a serial manipulator: length of composing elements and number of joints.

Key words: manipulator serial; Denavit-Hartenberg; trajectory; MATLAB.

1. Introduction

Industrial robots are a critical part of today’s manufacturing processes, fact that translates into a growing need of research activities, in the direction of extending robots usability area and also for increasing their performances in software and hardware applications (Teresko, 2004; Beltran Blanco *et al.*, 2004).

The main characteristics of the industrial robots are:

*Corresponding author; *e-mail*: ciobanasu2004@yahoo.com

− robots have the purpose of executing operations of manipulation, movement and transportation, that require speed and precision, but with limited strength;

− they posses different numbers of free degrees (between 2 and 6) such that they can execute complex operations, every movement being commanded by the control unit;

− they are autonomous, functioning without systematic human intervention.

The mechanical system of a robot represents a configuration of rigid bodies, system elements, sequentially linked together through rotational (revolute) or prismatic (sliding) joints. The relative positions of these elements determine the overall position of the mechanical arm in the assembly, this position actually representing one of the functional conditions of the robot.

The most commonly known types of mechanical joints present in the robotic systems are represented by opened kinematic chains, where position, speed and acceleration of an element can be recursively determined based on the parameters of the previous element. Usually each element possesses one degree of freedom (DOF) in relation to the previous element, thus the transformation matrix between elements contains only one variable parameter.

The "cascading" of all transformations associated with each element enables the determination of the motion parameters of the entire mechanical configuration, and consequently of the end-effector of the robot.

This paper handles the subject of trajectory computation based on the position of the constituting elements of a manipulator, using the Denavit-Hartenberg method.

2. Conceptual Design

2.1. System of Coordinates for Kinetic Chains

A manipulator with n joints will have (n+1) elements, considering that each joint connects two links. The joints are numbered from 1 to n, and the elements from 0 to n, starting from the base. Using this notation, joint i connects link (i−1) to link i. The location of joint i is fixed in relation to link (i−1). By activating joint i, link i is moved. On that account, the first link (number 0), is fixed, and is not influenced by the movement of the other joints.

Joint i^{th} has a corresponding associated joint variable, denoted by q_i. The following notations will be used throughout the paper:

$$q_i = \begin{cases} d_i, & \text{displacement of prismatic joint } i, \\ \theta_i, & \text{angle of rotation of rotational joint } i. \end{cases} \tag{1}$$

In order to realize the kinematic analysis, a coordinate frame is associated to each link, more specifically link i has the corresponding attached frame $o_ix_iy_iz_i$. Any motion the robot executes, the coordinates of each point on link i are constant in relation to the i^{th} coordinate frame. Moreover, movement of joint i translates in a resulting motion of link i and its corresponding frame $o_ix_iy_iz_i$.. The first frame, $o_0x_0y_0z_0$, corresponding to the robot base, represents the inertial frame. A manipulator with several links and their attached frames can be observed in Fig. 1.

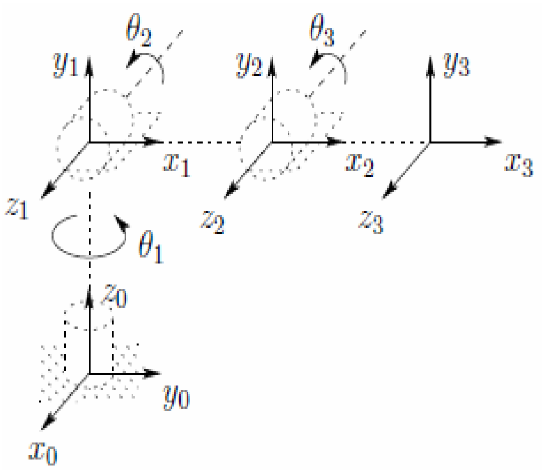

Fig. 1 – System of coordinates for an elbow manipulator.

2.2. Denavit-Hartenberg Method

The axis of a joint is defined as the spatial line around which element i rotates in report to element $i-1$. An element is characterized by: the relative position of the two axes that connect the element and the angle between the axes, referred to as the element angle and denoted by α_{i-1}, as seen in Fig. 2.

In order to emphasize the element angle a perpendicular common to the directions of the two axes is drawn. This angle is measured in the direction of rotation of the right-hand rule oriented after a_{i-1}, such that axis $i-1$ overlaps axis i.

However, for a rigorous kinematic analysis the two defined measures are not sufficient to position an element part of the kinematic chain. An additional interconnection parameter is necessary. This parameter is denoted by d_i and is measured along the common axis of the two elements.

The method which uses these measures is known as Denavit-Hartenberg convention (Chircor & Curaj, 2001; Sciavicco & Siciliano, 2000). The Denavit-Hartenberg (D-H) parameters are:

- a_i – link length, length of each common normal;
- α_i – link twist, angle between two successive z-axes;
- d_i – joint offset, distance on the z-axis;
- θ_i – joint angle, rotation about the z-axis.

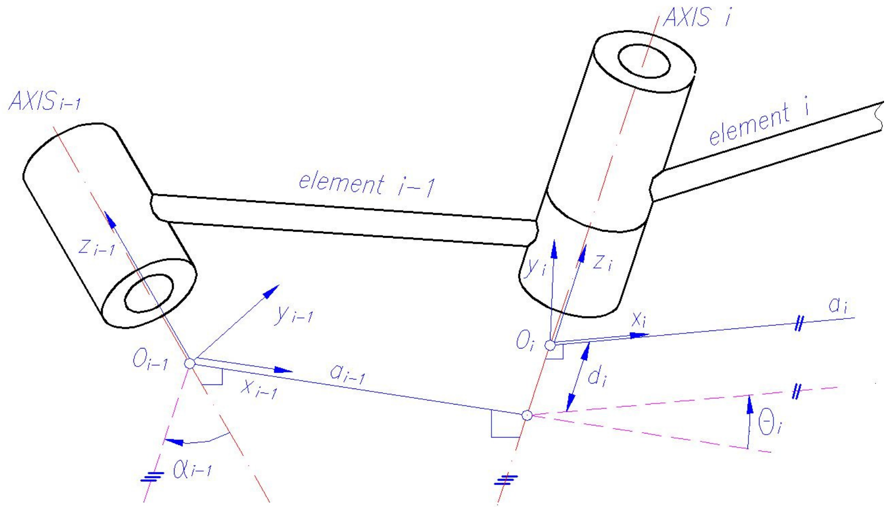

Fig. 2 – The relative position of 2 successive axes and the angle between them α_{i-1} (Popescu *et al.*, 1994).

In the Denavit-Hartenberg representation (Craig, 2013; Doroftei, 2002), each homogeneous transformation A_i is represented as a product of four basic transformations, as shown by:

$$A_i = R_{z,\theta_i} \text{Trans}_{z,d_i} \text{Trans}_{x,a_i} R_{x,\alpha_i} =$$
$$\begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \tag{2}$$
$$\times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The proposed program performs a positional analysis which enables computing the position and orientation of elements constituting an industrial serial robot, based on the coordinates ($o_ix_iy_iz_i$, i = 1,n) of the joints, using Denavit-Hartenberg parameters.

Table 1
Denavit-Hartenberg Input Parameters for Joint i

i	α_{i-1}	a_{i-1}	d_i	θ_i

3. Program Description

The program **manip_new** is realized using MATLAB (Hunt *et al*., 2006; Păstravanu & Barabulă, 2002) and is organized in twelve modules, of which the most important are mentioned: calcul_coord.m – computes the coordinates of the joints; calcul_transfo.m – computes the transformation matrix based on the Denavit-Hartenberg input parameters; Animate.m – constructs and outputs the image of the resulting trajectory.

Fig. 3 illustrates the program user interface, the possible commands, for inserting the input data and for determining the robot trajectory.

For introducing the Denavit-Hartenberg values of each joint the button "Initializare" (*Initialization*) shall be used, Fig. 4. The type of the added joint can be selected from a drop-down list, to be either translation or rotation joint. Button "Urmatoarea" (*Next*) is used for adding a new joint and "STOP" for finalizing the input data insertion.

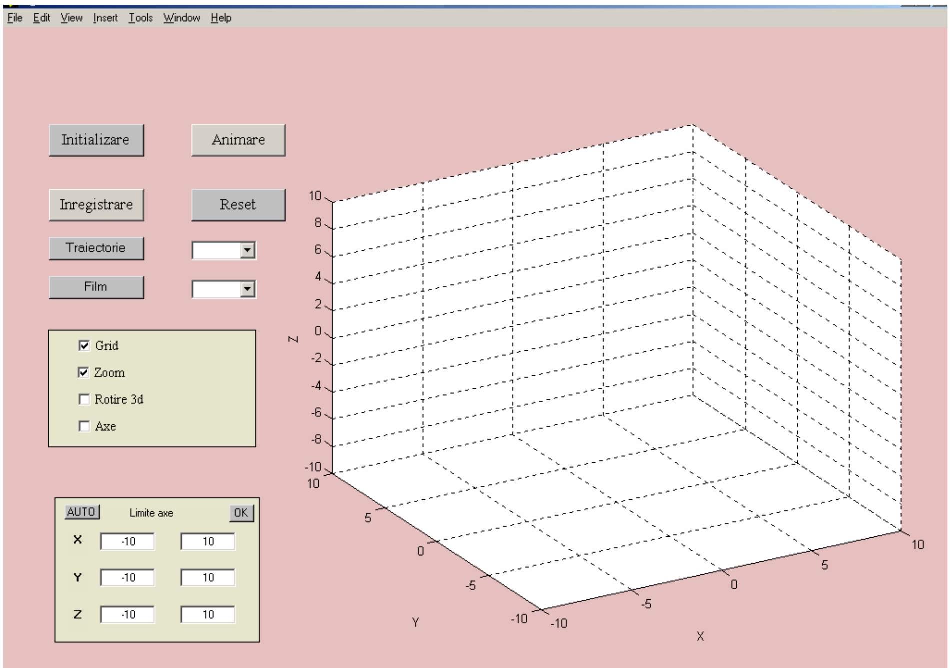

Fig. 3 – Program user interface.

The user has the option of storing the history of the data, using button "Inregistrare" (*Record*). Button "Traiectorie" (*Trajectory*) enables the output of the trajectory of the selected joint.

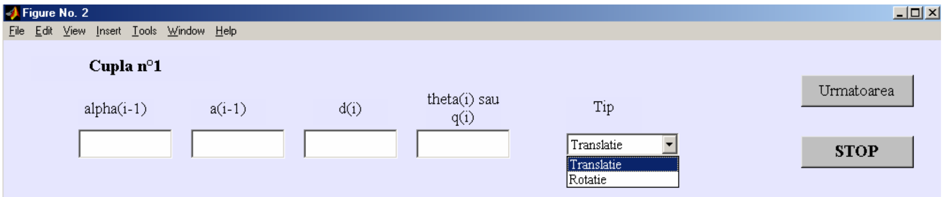

Fig. 4 – Window for setting Denavit-Hartenberg values for each joint.

Fig. 5 shows the interface for configuring the animation of robot trajectory, button “Animare” (*Animation*). For each joint, three parameters have to be introduced: the start angle, the stop angle and the number of steps. A new joint configuration is added using the “Urmatoarea” (*Next*) button.

Button “START” enables the spatial movement of the robot, corresponding to the input data.

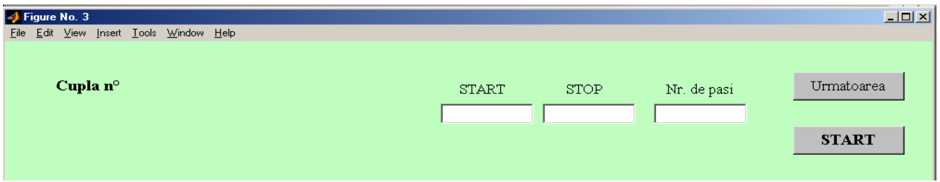

Fig. 5 – Animation configuration parameters.

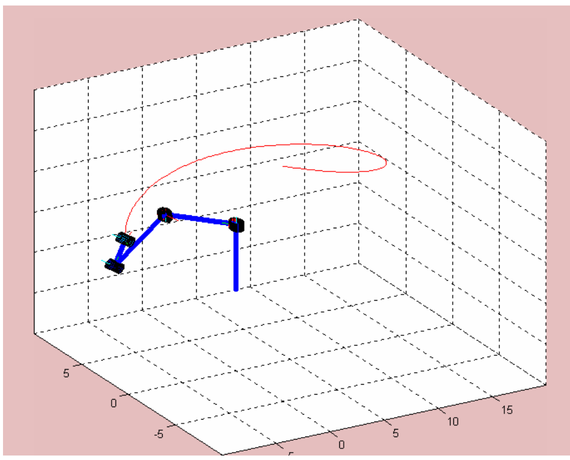

Fig. 6 – Trajectory animation of a serial manipulator.

An example of a resulting animation is shown in Fig. 6. Part two of this paper handles the application of the program following a case study and illustrates the obtained trajectories.

4. Conclusions

1. The present paper proposes a program developed in MATLAB that allows robot trajectory analysis, based on Denavit-Hartenberg input parameters.

2. The program can be used for any serial manipulator, independent of the number and types of joints.

3. The virtual simulation can be used for educational purposes.

4. The interface is user friendly, commands for simulated motion are intuitive and distinct colours highlight the different robot elements and trajectory.

REFERENCES

Beltran Blanco M., Feliu B., Jorge J. *et al.*, *Simurob. Simulador del Robot IRB -1400*. XXV Jornadas de Automática, Ciudad Real, 2004.

Chircor M., Curaj A., *Elemente de cinematică, dinamică şi planificarea traiectoriilor roboților industriali.* Bucureşti, Edit. Academiei Române, 2001.

Craig J.J., *Introduction to Robotics: Mechanics and Control*. 3rd Edition, Pearson, 2013.

Doroftei I.D, *Arhitectura şi cinematica roboților.* Edit. Tehnică, Stiințifică şi Didactică CERMI, Iaşi, 2002.

Hunt B.R., Lipsman R.L. *et al.*, *A Guide to MATLAB: For Beginners and Experienced Users*. Second Edition, Cambridge University Press, 2006.

Păstravanu O., Barabulă A., *Sisteme automate – elemente de curs.* Edit. Gh. Asachi, 2002.

Popescu P., Negrean I., Vuşcan I., Haiduc N., *Mecanica manipulatoarelor si roboților, Probleme, vol. 2, Modelul geometric direct.* Bucureşti, Edit. Didactică şi Pedagogică, 1994.

Sciavicco L., Siciliano B., *Modelling and Control of Robot Manipulators*. Springer Verlag, London, 2000.

Teresko J., *Robots Revolution.* Industry Week, 2004 (www.industryweek.com).

SIMULAREA VIRTUALĂ A MIŞCĂRII UNUI MANIPULATOR SERIAL
I. Descrierea programului

(Rezumat)

Prezenta lucrare are drept obiectiv abordarea uneia dintre cele mai dificile dar şi mai importante probleme de analiză cinematică a roboților şi anume problema pozițională. În cadrul problemei directe a pozițiilor (modelul geometric direct) se consideră cunoscute caracteristicile geometrice ale robotului şi legile de variație ale coordonatelor generalizate, urmărindu-se determinarea legilor de variație ale poziției absolute a punctului caracteristic.

Lucrarea propune un program conceput în MATLAB care permite analiza traiectoriei oricărui lanț cinematic deschis, costituind un manipulator serial, indiferent de numărul elementelor şi articulațiilor şi indiferent de tipurile articulațiilor.

Programul a fost dezvoltat pe cosiderentul că sistemul mecanic al unui robot este format dintr-o succesiune de corpuri rigide şi anume elementele sistemului, legate între ele prin articulaţii de rotaţie şi sau translaţie. Poziţiile relative ale elementelor determină poziţia braţului mecanic, (end-effector), îndeplinind una dintre condiţiile funcţionale ale robotului. Cu alte cuvinte, poziţia unui element poate fi obţinută recursiv din poziţia elementului precedent.

Drept parametri de intrare au fost utilizaţi parametrii Denavit-Hartenberg.

Programul propus poate determina şi trasa 3D traiectoria punctului caracteristic totodată realizându-se şi animaţia mişcării; se pot face vizualizări din diverse direcţii, proiecţii pe plane şi se pot reprezenta grafic traiectoriile cuplelor, precum şi traiectoria puncului caracteristic P, în funcţie de paşii consideraţi. Programul permite consultarea unui istoric al mişcării.

Simularea virtuală se pretează a fi utilizată în diverse scopuri, cel educaţional fiind unul dintre acestea. Interfaţa programului este prietenoasă pentru utilizator, comenzile pentru simularea mişcărilor sunt intuitive, iar culorile sunt folosite în mod distinct pentru elementele robotului, articulaţii şi traiectorii.

Partea a doua a lucrării prezintată modul de utilizare a programului propus pe un studiu de caz.