SIMULATION OF ROBOT-MANIPULATOR MOTION CONTROL

Eduard Kuznetsov - Anton Panda

Abstract:
This article provides a technology for develop a physical and mathematical simulation of the robot's computer model. The purpose of the work is to learn how to define the movement of robot elements. A complete, controlled computer model of an industrial robot manipulator has been created. Model control is implemented using the modeling environment software. The technology developed is used to master the basics of robotics.

Key words:
Rrobot, Trajectory, Positioning, Coordinates, Nodal points, Movable link, Software

Introduction

An objective reason for the arising and development of modern Robotics is a constant need flexible automation for production, for removing a person from direct participation in native production, for decrease disadvantage of traditional means of automation.

Application of robot opens up broad prospects for developing innovative technological processes without constrains, that direct participation human in them [1]. It means human physical limitations (load, speed, accuracy and repeatability manipulation), and its working conditions. Today direct human involvement in the process is often a major obstacle to the further intensification of production and creating new technologies.

Industrial robots are widely used in various fields of economy: research, transport, agriculture, medicine, in the service sector. The introduction of Robotics gives considerable technical and economic effect of increasing the productivity, technological processes, improve their product quality. Also positive is the release of a person from a difficult, dangerous and monotonous work. Complex automation of production and flexible automated productions is one of the main directions of introduction of Robotics and largely determines the modern Robotics growth rate.

Industrial robots are very expensive, and use them for educational purposes cannot afford any educational institution. However, there is an opportunity to explore controls and dynamics of industrial robot in its computer model. This model is not demanding in maintenance, can be changed easily, and can’t corrupt when receiving invalid movement commands. Using the model, you can quickly optimize the robot-prototype work for their specific application and environmental conditions.

The main objective of this study is to develop a computer model of a controlled robot-manipulator for use in educational process in training students of computer specialties.

Information technology of creation the computer model of a robot-manipulator allows to investigate the movement of work elements for any robot-prototype trajectories, and to analyze the design features of the robot in terms of optimal execution of operational tasks. This technology allows to accelerate the assimilation of the foundations of Robotics and gives the opportunity to apply the skills of programming the controllers in practice.

Object and environment of simulation

The object for computer simulation is a robot-manipulator “FANUC LR Mate 200iC” (Fig. 1). This compact high-speed robot with high positioning accuracy is ideal as a tool for studying the control of robotic systems. Equipping the robot-prototype by special tools allows to learn the various aspects of the use of such systems in the workplace. Robot-manipulator is operated by controller R30iA [2]. It has a closed mechanical design that hides the mechanisms of drives and electronic components.

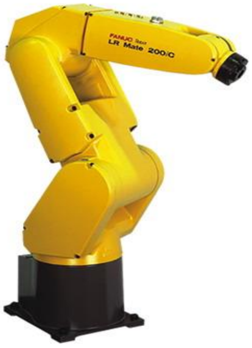

Fig. 1 Robot-manipulator “FANUC LR Mate 200iC”

A computer model created by using an Autodesk Inventor software. This software allows you to easily program the model elements moving along specified path. One of the interesting features of Autodesk Inventor is the creation of custom software components that implement the automate elements of the most frequently performed tasks [3]. These software components are implemented on the basis of application programming interface (API) and advanced object modeling. The most effective way to create custom applications based on AI is the use of Visual Basic for Application programming environment. Autodesk Inventor API determines the modus operandi of the programmer and the selection of appropriate programmatic functionality of operation environment [3]-[4]. In other words, Autodesk Inventor API allows you to create an optimal sequence of commands for Autodesk Inventor [4].

The study develop a robot-manipulator simulator, which includes a computer model of the robot and its movement control. In order to achieve the objectives of the study were assigned the following tasks:

1. To develop a mathematical model of prototype robot-manipulator elements movement.
2. To analyze the programming techniques of the movements of computer models by Autodesk Inventor tools.
3. To develop a motion control algorithm of a robot-manipulator.

Physical model of robot-manipulator motion

Planning the motion trajectory of the robot-manipulator elements is the task of control method selecting to ensure the movement of the manipulator along the specified trajectory. Before starting the manipulator, it is important to know:

1. Are there obstacles in the way of the robot elements movement?
2. Are there constrains on the mobility of robot elements in its area of operation?

When planning trajectory, one of two approaches is used:

1. A set of conditions for positioning, speed and acceleration of the generalized coordinates of the manipulator's working tool in the specific (nodal) points of the trajectory is assigned. The trajectory scheduler chooses from a class of functions (typically among polynomials a function that will route trajectory through the cue points and satisfy the specified constraints in the trajectory. Constrains and plan the trajectory are defined in generalized or Cartesian coordinates.

2. The desired motion trajectory of the manipulator elements is assigned as an analytically described function. The planner develops an approximation of a given trajectory with a mathematical function in generalized or Cartesian coordinates.

Planning the trajectory in generalized coordinates has the following advantages:

1. Specifies the change of coordinates that are directly control the movement of the manipulator.
2. The trajectory can be planned in real time.
3. The trajectories in generalized coordinates are easier to define.

The disadvantages of this planning are difficult to determine the position of the manipulator elements during the movement process. This is necessary to prevent collisions with obstacles.

In general, the basic algorithm for define the nodal points of trajectory h(t) in the generalized coordinates space is very simple and does not require description. Note: the trajectory must be clarified (adjusted) at each nodal point of the motion.

When planning on a trajectory, the following constraints are applied [4]:

1. The cue points should be easily calculated in non-recurrence way.
2. The interim positions should be clearly defined.
3. It shall be ensured that the generalized coordinates and their two first derivatives are inseparable so that the planned trajectory in the generalized space is smooth (no excesses and leaps).

In general, the planning of a trajectory in Cartesian coordinates consists of the following sequential actions:

1. Formation the located along the planned motion trajectory the sequence of cue points in Cartesian space.
2. Select a class of functions that approximate the trajectory segments between the cue points according to some criterion (e.g. straight, arc of circle, parabola, etc.).

The first step gives you a high degree of accuracy along a given trajectory. However, it may require a large amount of computation that will slow down the manipulators movement due to recalculating the Cartesian coordinates into generalized coordinates if there are no sensors of the Cartesian coordinates of elements position. Therefore, the following action is to recalculate Cartesian coordinates of the cue points into the corresponding generalized coordinates, and interpolation by the polynomials of a low degree further.

Simulating the geometric and kinematic parameters of a manipulator

In order to program the manipulator motion in Autodesk Inventor, it is uses the auxiliary (transient) geometry of the Autodesk Inventor application programming interface (API). The auxiliary geometry is designed to solve by API a set of mathematical tasks, mostly geometric. Elements of auxiliary geometry are created using calls to the appropriate methods of the objects and are widely used in Autodesk Inventor API. Unlike most other geometric API objects, these objects do not have visual display tools. Most often, these are abstract mathematical proxy objects that are designed to manage the “real” geometry of Autodesk inventor.

To describe rotational and progressive connections between the neighboring links, Denavit and Hartenberg proposed a matrix method of sequentially building the coordinate systems associated with each link in the kinematic chain. The idea of the Denavit-Hartenberg presentation is to create a homogeneous transformation matrix that describes the position of each link's coordinate system relative to the previous link's coordinate system. It enables to consistently convert the coordinates of the manipulator elements from the last link coordinate system into the basic coordinate system, which is the inertial coordinate system for this dynamic system. Each coordinate system is based on the following three rules:

1. The zi-1 axis is directed along the i-th axis of the element.
2. Axis xi is perpendicular to the zi-1 axis and directed from it.
3. The yi axis supplements the xi and zi axes to the right Cartesian coordinate system.

The kinematic structure of the robot Fanuc LR Mate 200iC is presented in Figure 2. The specified local coordinate system is associated with the i-th link. The configuration coordinates are specified through qi. The Denavit-Hartenberg parameters are presented in Table 1.

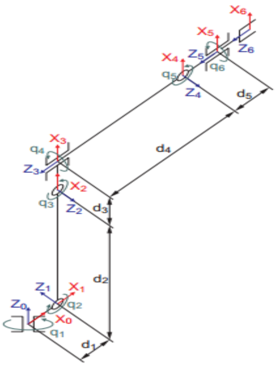

Fig. 2 Kiinematic chain of robot-manipulator

Table 1 Denavit-Hartenberg parameters

№ link	$\mathbf{a_i}$	$\boldsymbol{\alpha_i}$	$\mathbf{d_i}$	$\boldsymbol{\theta_i}$
1	d_1	$-(\pi / 2)$	0	q_1
2	d_2	π	0	$q_2 - (\pi / 2)$
3	d_3	$-(\pi / 2)$	0	$q_2 + q_3$
4	0	$\pi / 2$	$-d_4$	q_4
5	0	$-(\pi / 2)$	0	q_5
6	0	0	$-d_5$	q_6

The following geometric parameters of robot elements are used in Table 1:
d1 - The offset between the coordinates of x0, y0, z0 and x1, y1, z1 (mm).
d2 - Length of the first robot element (mm).
d3 - The offset between x2, y2, z2 and x3, y3, z3 (mm).
d4 - Length of the second robot Element (mm).
d5 - Length of the third robot element (mm).

Mathematical model of robot-manipulator motion

To move a manipulator tool along a specified trajectory, a sequence of intermediate manipulator elements positions is specified. At the beginning, the coordinate arrays of these elements intermediate positions on the motion trajectory are defined. The next action is to define a set of rotation angles for all the manipulator elements for each coordinate on the trajectory. These coordinates must conform to the current position of the element on the trajectory (Fig. 3).

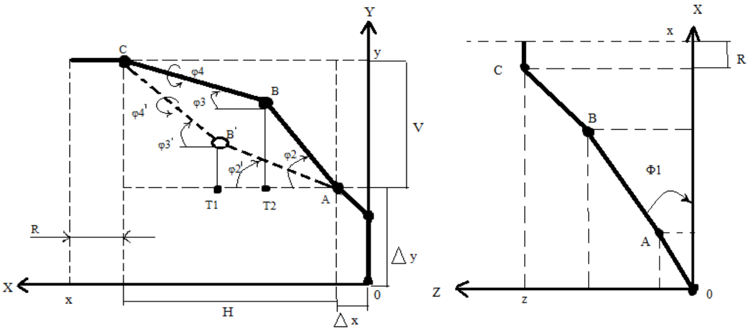

Fig. 2 Rotation angle values in planes XY and XZ

For Example, to locate the point T, uses a classic task of crossing two circles solution:

$$T = K\,H \pm \sqrt{K^2 H - \left(V^2 + H^2\right)\left(K^2 - (AB)^2 V^2\right)} \Big/ \left(V^2 + H^2\right),$$

$$K = (AB)^2 - (BC)^2 + H^2 + B^2/2$$

There are two ways to implement model in Autodesk Inventor a controlled computer: to develop an application or to create a macro. It have chosen a method that consists of writing a macro by Visual Basic for application programming language, due to no additional software is required: Inventor has a built-in Visual Basic for Application language editor.

Conclusion

In the course of carrying out the work, an analysis of the existing information technologies for the control of robot-prototype computer models was carried out. The functioning controlled computer model of an industrial robot-manipulator and developed technology for its creation is a result. The application of the resulting technology in the education process enables to learn how to control robotic systems, to examine the functional requirements of the model development environment, and to reinforce the programming skills of the controller systems.

Acknowledgments: *The authors would like to thank the KEGA grant agency for supporting research work the project KEGA: 004TUKE-4/2017*

References

[1] I. MAKAROV, Y. TOPCHEEV, Robotics: history and perspective. Moscow, Nauka, 2003. (In Russian)

[2] LR MATE 200IC INFORMATION [electronic source] URL: https://www.robots.com/fanuc/lr-mate-200iC

[3] V. KONTSEVICH, Solid modeling in Autodesk Inventor. Kyyv, DiaSoft, 2007. (In Russian)

[4] N. SVIRNEVSKY, Developing applications for Autodesk products. Khmelnitsky, KNU, 2016. (In Russian)

Contact address

(1.) Senior lecturer Eduard Kuznetsov, PhD
Sumy State University, Faculty of electronics and information, Rimskogo-Korsakova 2, 40007, Sumy, Ukraine, e-mail: kuznetsov@opm.sumdu.edu.ua
(2.) prof. Ing. Anton Panda, PhD.
Technical university in Košice, Faculty of manufacturing technologies with the seat in Prešov, Štúrova 31, 080 01 Prešov, Slovakia, e-mail: anton.panda@tuke.sk