SIMULATION OF ROBOT RV-2AJ USING SOFTWARES SOLIDWORKS AND NI LabVIEW

Muhamed SERDAREVIĆ1,*, Malik ČABARAVDIĆ1, - Fuad HADŽIKADUNIĆ1 - Amra TALIĆ-ČIKMIŠ1 - Dženana TOMAŠEVIĆ1
1 University of Zenica, Faculty of Mechanical Engineering, Zenica, Bosnia and Herzegovina

Received (23.04.2018); Revised (04.06.2018); Accepted (06.06.2018)

Abstract: In this paper, the attention is given to the simulation of the industrial robot Mitsubishi RV-2AJ with the help of program tools that include two software packages – SolidWorks and NI LabVIEW. The main purpose of the simulation is to visualize the CAD model management through a control panel, which will be created by using a software NI LabVIEW. The control panel will include the controls for angle and coordinate entry, speed input, and indicators that will be showing the movement and stopping of the robot, as well as its exact position in the space at all times at any point.

Key words: robotic arm, SolidWorks, NI LabVIEW, motion analysis

1. INTRODUCTION

A robotic arm is a type of mechanical arm, usually programmable, with similar functions like a human hand, and also a robotic arm can be a total sum of the mechanisms or it can be part of a more complex robot. Mitsubishi RV-2AJ robot model which is used for simulation is shown in Figure 1.

Fig.1. Mitsubishi RV-2AJ [1]

The rapid development of personal computers enabled the success of modeling and simulation in all branches of science and technology, so increasing the speed and capacity of computer memory enables successful solving of practical problems by experimenting through simulations.

2. ROBOT KINEMATICS

The n-joint manipulator will have n+1 links, because each joint is connected to two segments. The joints are counted from 1 to n, and the links from 0 to n starting from the base. With this convention, the joint i connects link i-1 and link i, considering the position of the joint i that is fixed relative to the segment i-1. When the joint i is controlled the link i rotates. Furthermore, segment 0 (first segment) is fixed, i.e. does not move when it is operated with the joints.

2.1. Direct kinematics

The goal of direct kinematics is to find the coordinates of the end-effector based on the given joint coordinates [2]. A set of conventions have been developed that offer systematic procedures to perform this analysis. It is, of course, possible to accomplish a robot kinematics analysis without following the rules of the convention. However, the kinematic analysis of n-link manipulators can be extremely complex and the conventions further simplify the kinematic analysis. Moreover, they build the universal language by which robotics engineers can communicate.

Fig.2. Denavit-Hartenberg kinematic parameters [3]
 Mostly in use is Denavit-Hartenberg convention. With this convention, every homogeneous transformation of $A_i$ is presented as the product of four basic transformations:

$$
A_i = \begin{bmatrix}
c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\
s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\
0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\
0 & 0 & 0 & 1
\end{bmatrix}
$$

(1)

2.2. Inverse Kinematics

The problem of inverse kinematics is determination of joint coordinates when end-effector position and orientation are given. The Iterative methods use the Jacobian matrix, which is a linear approximation of the differential function close to a given point [3]. Similar to the analytical solution, there is no function for each joint that describes in relation to their relative position of movement relative to the acceptor. However, since both of these functions are nonlinear, the linear approximation is derived from a relative position such that it describes the partial movement of the acceptor. From these approximations, Jacobian's matrix can be written [4]:

$$
J(\theta) = \frac{df}{d\theta}
$$

(2)

Each Jacobian matrix column describes an approximate change in the position of the end-effector when the corresponding scalar is changed. By multiplying the Jacobian matrix with a set of scales $\Delta \theta$ describing the change in each scalar in the configuration, the approximate change $\Delta P$ of the end-effector position will be solved.

$$
\Delta P = J(\theta) \Delta \theta
$$

(3)

Equation (3) results in an estimated change in the acceptance that is similar to a direct kinematic problem [4]. The approximating change of the scalar $\Delta \theta$ can be solved by the inversion function $J(\theta)$. The equation (4) can be written as:

$$
\Delta \theta = J(\theta)^{-1} \Delta P
$$

(4)

In other words this means that for a given rotational structure with set n of scalar $\theta$ and acceptor P, a Jacobian matrix can be formed using the equation (2). The inverting Jacobian matrix approximated by the scalar $\Delta \theta$ can be solved by the equation (4).

3. CAD MODELLING IN SOLIDWORKS

SolidWorks is CAD and CAE solid modelling computer program which runs on Microsoft Windows operating systems. Modelling in SolidWorks usually is started with 2D sketch. The sketch consists of geometries such as points, lines, arcs, polygons, etc. Dimensions are added to the sketch to define the size and location of the geometry. Also, the program contains the relations used to define geometry properties such as tangency, parallelism, verticality, and concentration. Parametric nature of SolidWorks means that dimensions and relations are controlled by geometry, and not vice versa [5]. Robot RV-2AJ parts modelled in SolidWorks are shown in Figure 3. below.

In assemblies, mates are relations analogous to the sketches. Just as the relations of the sketch define conditions such as tangency, parallelism and concentration, respecting the geometry of the sketch, mates define equal relations by respecting individual parts or components, allowing simple construction of the assembly.

3.1. Motion Study

Motion Studies are graphic motion simulations for assemblies. It is possible to incorporate visual properties such as the camera’s lightening and perspective in Motion Study. Motion Study do not change an assembly model or its properties, but only simulate and animate the movements of the previously described model. For this simulation it is necessary to set Motion Analysis active. Motion Analysis can be used to accurately simulate and analyze the effects of parts movement (including forces, springs, mufflers and friction) on the assembly [5]. Motion Analysis uses a strong computer kinematic solver, and calculates material properties such as mass and inertia in computation.

4. NI LabVIEW

LabVIEW is a system-designed platform and development environment for the visual programming language of National Instruments Company. LabVIEW is a system engineering software that is used for applications requiring testing, measurement and control with fast access to hardware and data. The LabVIEW programming environment simplifies the integration of hardware for engineering applications so that we have a consistent method of collecting data from NI and other hardware [6]. LabVIEW reduces complexity of programming so that we can focus on a unique engineering problem.

4.1. Graphic programming

LabVIEW combines the creation of user interfaces, called front panels, within the development cycle [6]. LabVIEW programs-subprograms are called virtual instruments.
Each virtual instrument has three components: a block diagram, a front panel, and a link plate. The front panel was built using controls and indicators. Controls are input units and they allow the user to enter information in virtual instrument. Indicators are output units and they display results based on entered input data in virtual instrument. The back panel, which is a block diagram, contains the original graphic code. All objects that are located on the front panel will appear on the back panel as terminals. The back panel also contains the structures and functions that perform the operations on the controls and transmit data to the indicators. Structures and functions can be found on functional pallets and can be placed on the back panel. Collective controls, indicators, structures and functions will be considered as nodes. The knots connect one another with the use of wires.

4.2. NI SoftMotion module

The NI SoftMotion module delivers a graphical environment for the normal application of motion control. With the NI SoftMotion module, we can use LabVIEW projects to set the axes of movement, test the configurations, set the engines, and quickly integrate motion control into the application [6].

5. SIMULATION APPLICATION

To successfully connect two programs, you need to take several steps:

1. Run SolidWorks and open the CAD robot model robot rv-2aj.SLDASM. Include the add-ons “SOLIDWORKS Motion” and “SOLIDWORKS Simulation”. Click on the “Motion Study 1” field in the bottom left corner and select the type of simulation “Motion Analysis”.

2. Launch the NI LabVIEW program and open the “robot rv-2aj.lvproj”. In the Project Explorer window, click the right mouse button on "My Computer" and add CAD model by pressing “Add SolidWorks Assembly”. It's practical to have already opened model in SolidWorks that allows NI LabVIEW to automatically recognize a file that is saved as a model assembly.

3. To simulate SolidWorks engines included in the model, it is necessary to connect the engines with the NI SoftMotion axis. This step is done by right clicking on “My computer” in the Project Explorer window and selecting “New” then “NI SoftMotion Axis”. Select “Add New Axis”. The new axis automatically connects to the SolidWorks engine.

4. In the Project Explorer window, elements are marked (combination of CTRL + left mouse click) on “My Computer”, SolidWorks assembly and virtual axes. When all the elements are marked, right-click and select "Deploy". If some sort of conflict resolution is displayed, select “Apply”.

5. Right click on the SolidWorks assembly in the Project Explorer window and select “Synchronize to Assembly”. Then, right-click on the SolidWorks assembly and select “Start simulation” to start the simulation in SolidWorks.

The following figure shows the windows in SolidWorks and NI LabVIEW, where the robot model Mitsubishi RV-2AJ is simulated.
6. CONCLUSION

Robot simulation is extremely valuable because it allows testing of solutions for complex tasks in virtual environment where it is easier to manipulate the robot and examine different layouts, scenarios and situations without danger of damaging real equipment or human injury. In such environment it is much easier to manipulate robot, since user can easily pan, zoom and rotate the view, define robot movement. In this work development of the CAD model in SolidWorks and virtual instrument in NI LabVIEW are presented, that enables motion simulation. This application should serve as a basis for development of a more robust robot simulation application. The application structure and used methods for kinematics solving allow relatively easy addition of different robot models. In the end, the application in future should support dynamic simulation, which will be a basis for testing different control algorithms.

REFERENCES