

# Kinematic modeling of a 6R industrial robot

M Ratiu<sup>1</sup>, A Rus<sup>1</sup> and M L Balas<sup>2</sup>

<sup>1</sup> Department of Mechanical Engineering and Automotive, University of Oradea,  
410087 Oradea, Romania

<sup>2</sup> Department of Product Design, Mechatronics and Environment, Transilvania  
University of Brasov, 500068 Brasov, Romania

E-mail: mratiu@uoradea.ro

**Abstract.** In this paper, we present the process of kinematic modeling and simulation, in ADAMS MBS of MSC software, of an articulated robot with six revolute joints, through direct and inverse kinematics. First, by using the STEP function, we define a spatial trajectory of the end effector, and, through inverse kinematics, we determine the motion laws of the six revolute joints. Then, we apply the inverse process on another virtual model of the same robot, by imposing the motion laws to the joints, to obtain the desired trajectory of the end effector, through direct kinematics. This work is a small part of our research regarding modeling and optimization of the industrial robots' motion.

## 1. Introduction

Nowadays, in the context of using the industrial robots at a wide scale, across all industries, and due to the rapid and significant progress of computer designing, virtual prototyping is more and more used for analyzing and optimization of the robots' motion. As presented in [1], there are a lot of studies and several complex software packages on the modeling of industrial robots.

In our work, we use ADAMS (Automatic Dynamic Analysis of Mechanical Systems) from MSC software, which is one of the most widely used Multi-Body System (MBS) software for the modeling, analysis, and simulation of the complex mechanical systems' motion.

In paper [2] is presented the process of the modeling in ADAMS of the mechanical system of a 6DOF (degree of freedom) articulated robot, with six revolute joints, starting from the geometric 3D CAD model of the robot ABB IRB 1600-6/1.2, in PARASOLID format, available on the manufacturer's website [3].

In continuation of our work, in this paper, we present the process of kinematic modeling and simulation of the robot, through direct and inverse kinematics, in ADAMS/View, by following the steps outlined in [4, 5]. The kinematic model of the robot contains the kinematic elements of the robot, connected through that six revolute joints, the geometric parameters specific of the mechanism, represented by the locations of the joints, and the kinematic restrictions - the motion generators [6].

Following the procedure proposed in [7], first, by using the STEP function, we define a spatial trajectory of the end effector, and, through inverse kinematics, we determine the motion laws of the six revolute joints. Then, we apply the inverse process on another virtual model of the same robot, by imposing the motion laws to the joints, to obtain the desired trajectory of the end effector, through direct kinematics.

## 2. Kinematic analysis of the robot

The kinematic analysis of a robot means calculation of the kinematic parameters of the mechanical elements of the robot, without considering their masses, or the forces and torques that cause their motion.

The kinematic analysis of any robotic manipulator can be done by direct (forward) and inverse kinematic analysis. The goal of the direct kinematics is the determination of the robot's end effector position and orientation when all robot joints variables are known. In contrast, the goal of the inverse kinematics is the determination of the joints variables, when are known the position and orientation of the end effector.

While for the direct kinematic analysis the Denavit-Hartenberg convention is well known and widely used, the inverse kinematic analysis is complex and time-consuming because of the lack of a unique solution. The inverse kinematics is the basis for the motion planning and control of the robot trajectory and forms the basis for dynamic analysis.

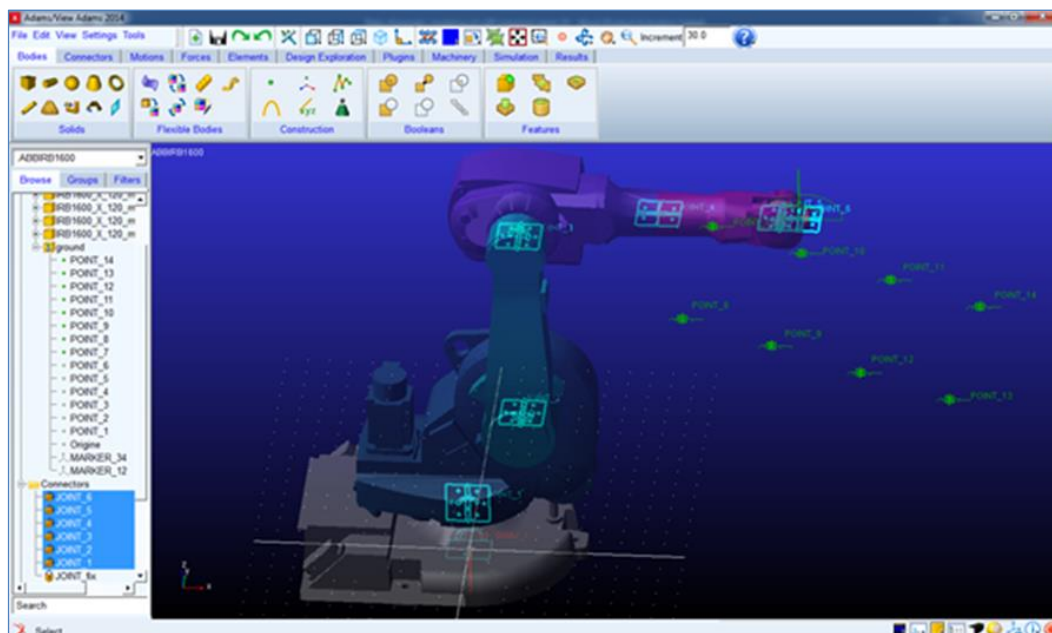
## 3. Inverse kinematic analysis of the robot

The problem of inverse kinematics for the robot consists of determining the motion laws of the six revolute joints for a particular trajectory of the end-effector.

In order to impose a spatial trajectory of the end effector, for example, UO-shaped, are created eight points, with the coordinates presented in table 1 and illustrated in figure 1:

**Table 1.** Defining points on the trajectory.

Point number	Coordinates (mm)		
	X	Y	Z
7	600	-100	1000
8	500	100	600
9	700	150	500
10	800	-50	900
11	1000	0	800
12	900	200	400
13	1100	250	300
14	1200	50	700



**Figure 1.** Creation in ADAMS the points for the imposed trajectory.

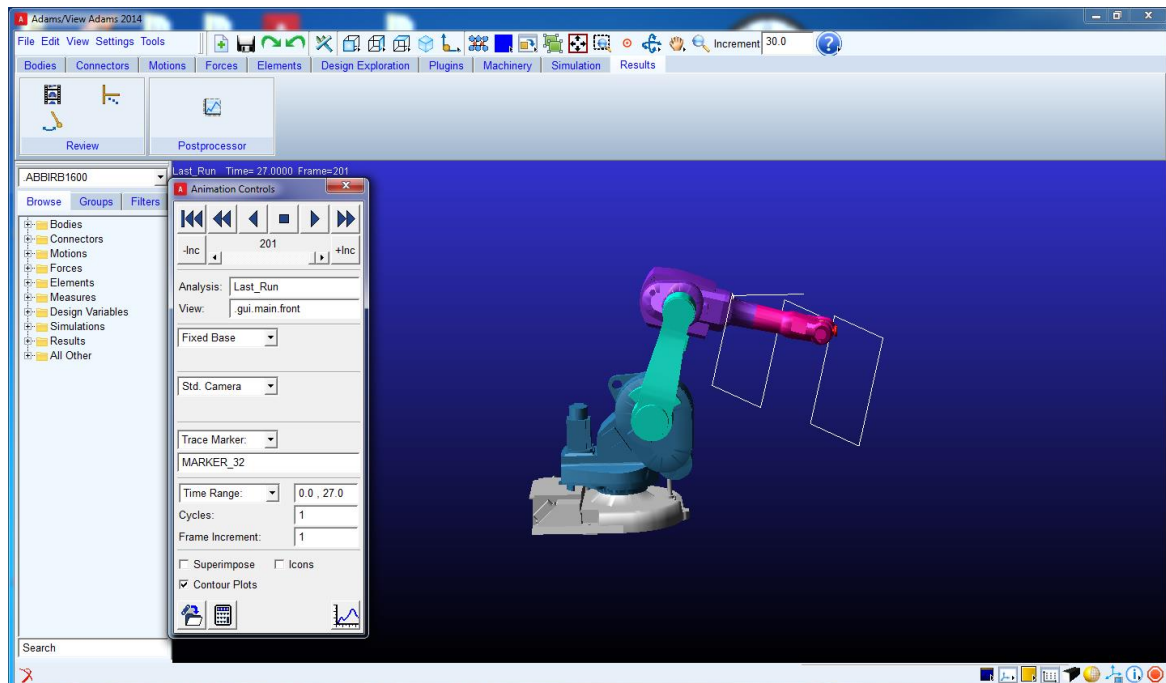
In ADAMS / View, with General Motion, using the STEP function, the end-effector displacements on X, Y, and Z are defined and presented in table 2:

**Table 2.** Defining STEP functions.

Displacement on X	Displacement on Y	Displacement on Z
STEP(time, 0, 0, 0.001, -290) +	STEP(time, 0, 0, 2.001, -100) +	STEP(time, 0, 0, 5, 38.5) +
STEP(time, 4.999, 0, 5, -100) +	STEP(time, 6.999, 0, 7, 200) +	STEP(time, 5, 0, 5.001, -400) +
STEP(time, 6, 0, 6.001, 200) +	STEP(time, 8, 0, 8.001, 50) +	STEP(time, 5.999, 0, 6, -100) +
STEP(time, 10.999, 0, 11, 100) +	STEP(time, 12.999, 0, 13, -200) +	STEP(time, 11, 0, 11.001, 400) +
STEP(time, 12, 0, 12.001, 200) +	STEP(time, 14, 0, 14.001, 50) +	STEP(time, 11.999, 0, 12, -100) +
STEP(time, 16.999, 0, 17, -100) +	STEP(time, 18.999, 0, 19, 200) +	STEP(time, 17, 0, 17.001, -400) +
STEP(time, 18, 0, 18.001, 200) +	STEP(time, 20, 0, 20.001, 50) +	STEP(time, 17.999, 0, 18, -100) +
STEP(time, 22.999, 0, 23, 100) +	STEP(time, 24.999, 0, 25, -200) +	STEP(time, 23, 0, 23.001, 400) +
STEP(time, 24, 0, 24.001, -200)	STEP(time, 26, 0, 26.001, -50)	STEP(time, 23.999, 0, 24, 100)

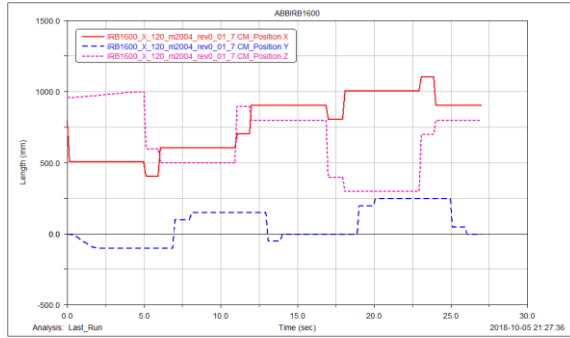
In ADAMS, the STEP function approximates an ideal mathematical STEP function (but without discontinuities). A STEP function is used when a value has to be changed from one constant to another.

An animation of the robot motion, for 27 s and 200 steps, and the trajectory followed by the end effector, are presented in figure 2:

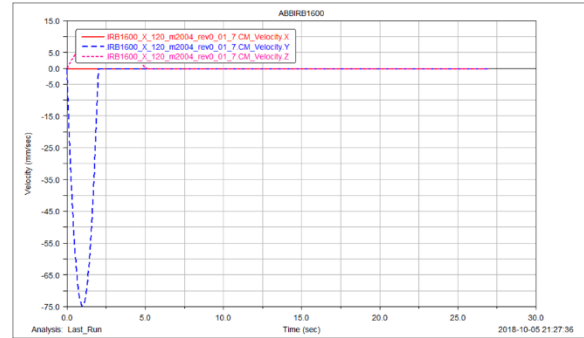


**Figure 2.** Simulation of imposed trajectory.

By using Adams / Postprocessor, the variation of the main kinematic parameters (position and velocity) of the end effector mass center, represented by Marker 32, for a simulation of 27 s and 200 steps are obtained. In figure 3a can be seen the displacement curves of the end effector on X, Y and Z. The curves that represent the variation of the end effector' velocity on X, Y, and Z, are presented in figure 3b.



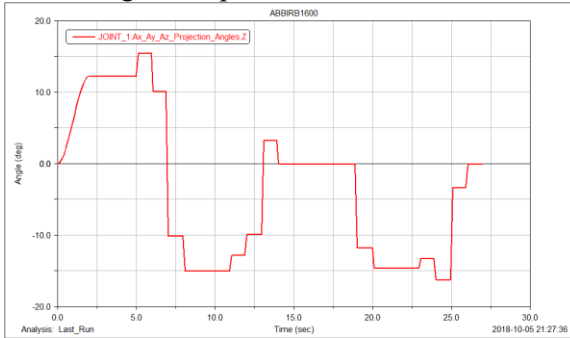
a)



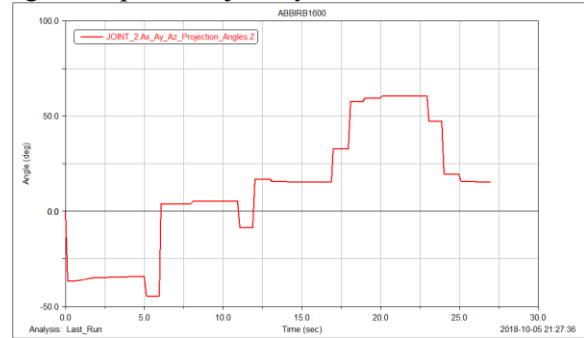
b)

**Figure 3.** The variation of the end-effector position (a) and velocity (b).

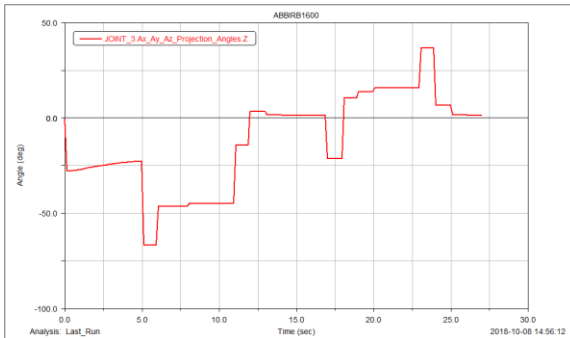
The motion laws of the six revolute joints of the robot, 1, 2, 3, 4, 5, and 6 (illustrated in figure 1), are presented in figure 4. They represent the variation curves of the angular displacements in each revolute joint, during the displacement of the end effector along the imposed trajectory.



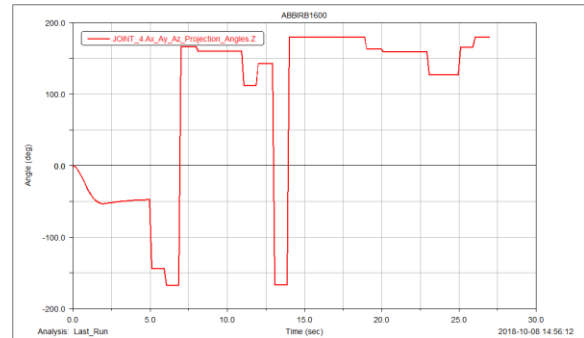
Joint 1



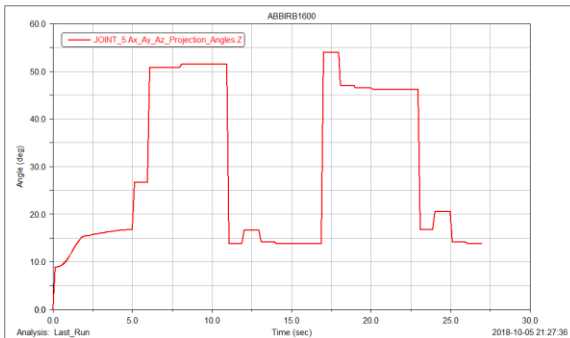
Joint 2



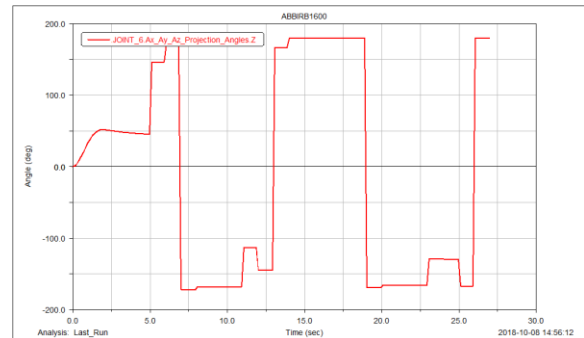
Joint 3



Joint 4



Joint 5



Joint 6

**Figure 4.** The motion laws of the six revolute joints, 1, 2, 3, 4, 5, and 6.

#### 4. Direct kinematic analysis of the robot

The problem of direct kinematics in ADAMS consists in imposing the motion laws of the six revolute joints, in obtaining a certain trajectory of the end effector, by going through the following steps:

- measures for the angular displacements of the six revolute joints are created;
- the motion laws of angular displacement of the revolute joints are accessed from the simulation results of the inverse kinematics;
- spline curves are created;
- the motion laws of the six revolute joints are imposed/modified. The analysis of the kinematic model of the robot, presented in figure 5, will be the next step in our research.

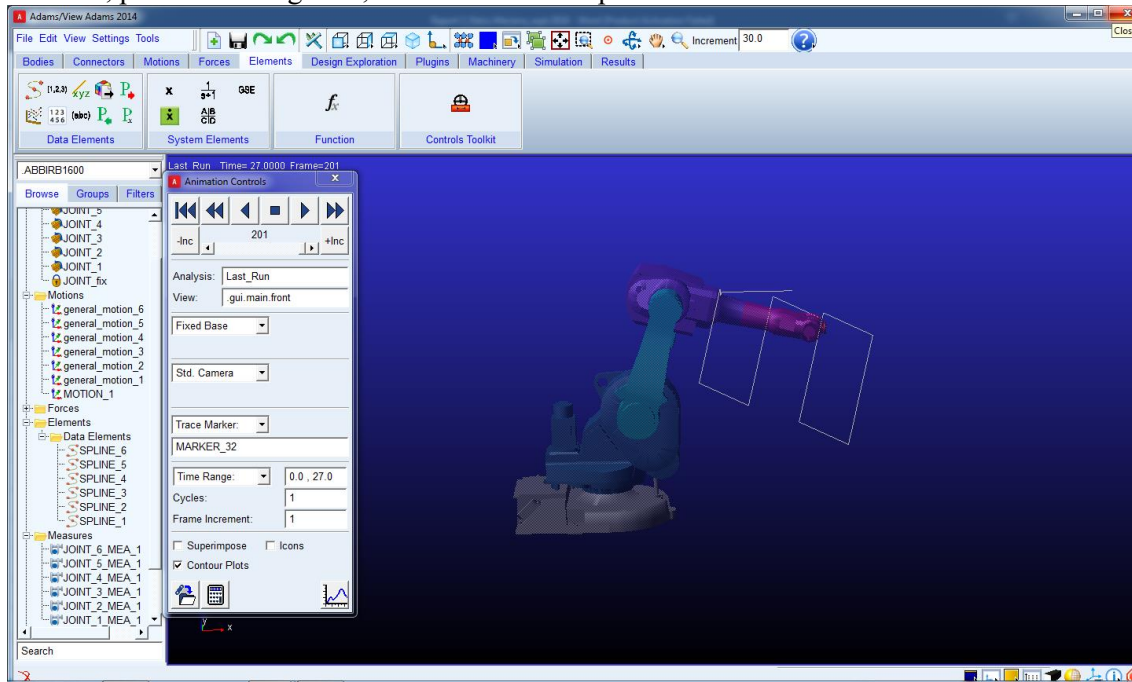


Figure 5. Kinematic model of the robot in ADAMS.

#### 5. Conclusion

We presented this paper, which represents a small step in our larger research regarding modeling and optimization of the industrial robots' motion, in ADAMS MBS software, by hoping that it will be useful to the students or other interested people who start learning to model such robots in ADAMS.

#### References

- [1] Ratiu M and Rus A 2017 Modeling of the trajectory-generating equipments *IEEE publication 14th International Conference on Engineering of Modern Electric Systems (EMES)* DOI: 10.1109/EMES.2017.7980418
- [2] Ratiu M, Rus A and Balas M L 2018 Modeling in ADAMS of a 6R industrial robot, *Annual Session of Scientific Papers IMT ORADEA 2018 (MATEC Web of Conferences)* **184**, 02006
- [3] [www.abb.com/robotics](http://www.abb.com/robotics) accessed in July 2017
- [4] *Getting started using ADAMS/View 2014* retrieved from [mscsoftware.com](http://mscsoftware.com) on September 2015
- [5] *Adams Tutorial Kit for Mechanical Engineering Courses (Third Edition)* 2015 retrieved from [www.mscsoftware.com](http://www.mscsoftware.com) on July 2017
- [6] Alexandru C 2006 Modeling in mechatronic concept and virtual testing of orientation mechanisms used in solar panels *Mechanism and Manipulators Magazine* **2/2** pp 49-54
- [7] Enescu M and Alexandru C 2011 Modeling and simulation of a 6 DOF robot *Proceedings of 2011 International Conference on Optimization of the Robots and Manipulators* (Sinaia, Romania: University Politehnica of Bucharest)