Simulation of Vision-Based Model of Robotic Work Cell (6-Axis)

Mitul Gajbhiye, Amol Chaudhary and Sangharatna Ramteke

October 6, 2021

Simulation of vision-based model of robotic work cell (6-axis)

Mitul Milind Gajbhiye[1], Amol Chaudhary[2] & Sangharatna Ramteke[3]
[1]Research Scholar, Department of Mechanical Engineering, G. H. Raisoni College of Engineering, Affiliated to
R.T.M.N.U., Nagpur, Maharashtra, India
[2]Assistant Professor, Department of Mechanical Engineering, G. H. Raisoni College of Engineering, Affiliated to
R.T.M.N.U., Nagpur, Maharashtra, India
[3]Assistant Professor, Department of Mechanical Engineering, G. H. Raisoni College of Engineering, Affiliated to
R.T.M.N.U., Nagpur, Maharashtra, India

Abstract:

In this paper, a technique is demonstrated which will notice the end-effectors Denavit Hartenberg (DH) parameters table as per the necessity of labour envelope. For this purpose, the authors have used Roboanalyzer software. The manipulators' gripper is designed to move to perform a particular task in a specific way. Earlier, simulation for the ARISTO robot functioning for welding application of a curved surface has been discussed. We aim to design an MTAB ARISTO robot integrated with a vision module that suits perfectly for pick and place operation on the conveyor, upon which kinematics and simulation is done for the respective joints and links for the end effector. Motion analysis of manipulator is shown on graphs as a time sequence of links and joints. Each of which is done using RoboAnalyzer software.

Keywords: Denavit Hartenberg (DH) parameters, forward kinematics, inverse kinematics, manipulator, RoboAnalyzer.

1. Introduction

An articulated mechanism may be a robot with rotary joints articulated robots can vary from straightforward two-jointed structures to systems with ten or a lot of interacting joints and materials. They're high-powered by a range of means, as well as electrical motors. Some sorts of robots, resembling robotic arms, may be articulated or non-articulated. ARISTO has a simulation software package that allows the users to learn the functions, applications, and programming of the robot. MTAB ARISTO robots are widely accepted in completely different work applications like pick & place, palletizing, machine loading & unloading, and for fastening incurved methods. This automation is often equipped with grippers of two types i.e., pneumatic and electrical grippers with varieties of path, potential of square measures say defined destined point to point, Linear and circular interpolation methods. The required power for the automation is 230V AC, 50/60 Hz, 5A and the payload capability is 2.5 kilograms. The robot can be used directly by calling from the software library and can be modified as per the preferences for various practical and simulation purposes. Individual link lengths and joint parameters can be introduced in the robot. Performing various position-based analyses and carrying out the individual centre of gravity, link graphic properties over the whole process of performing the task.

Rajeevlochana & Saha gave a brief overview over the tools and software available in the market and research domain either free or paid for robot visualization [1]. Khan et. al. proposed other advanced robot visualization tools such as RVS4W [2] and similarly Botero et. al. proposed ROBOMOSP [3] which can specifically perform the forward and inverse kinematics and trajectory planning. Sadanand et. al. proposed the software named RoboAnalyzer which allows a user to analyse multiple inverse kinematic solutions and allowing to observe the motion among any of the two possible solutions [4].

This article presents the simulation and graphical properties of the MTAB ARISTO for the vision based robotic work cell over the number of steps involved in the simulation to generate a smooth path for trajectory generated by the movement of the manipulator and gripper. It also provides the plotting of graphs for various variables like acceleration v/s time, force/torque v/s time.

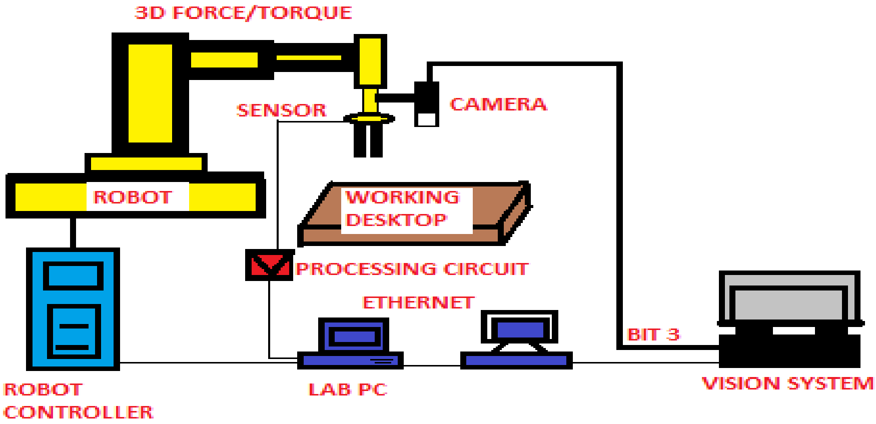


Figure 1. System hardware configuration.

2. Background

2.1. Vision system:

ITEX 15040 has been adopted. It includes a monochrome camera with a mono-focal lens, an image monitor, IMA-150/40 advanced image manager with 4 MB reconfigurable memory, an AM-VS variable scan acquisition module, a DM-PC pseudo color display module, a CM-PA programmable accelerator, and a BIT 3 model 466 Sbus-VME adaptor. The camera is attached to the robot's final axis, as shown in Figure 2. It can then tilt freely as the robot twists its wrist.

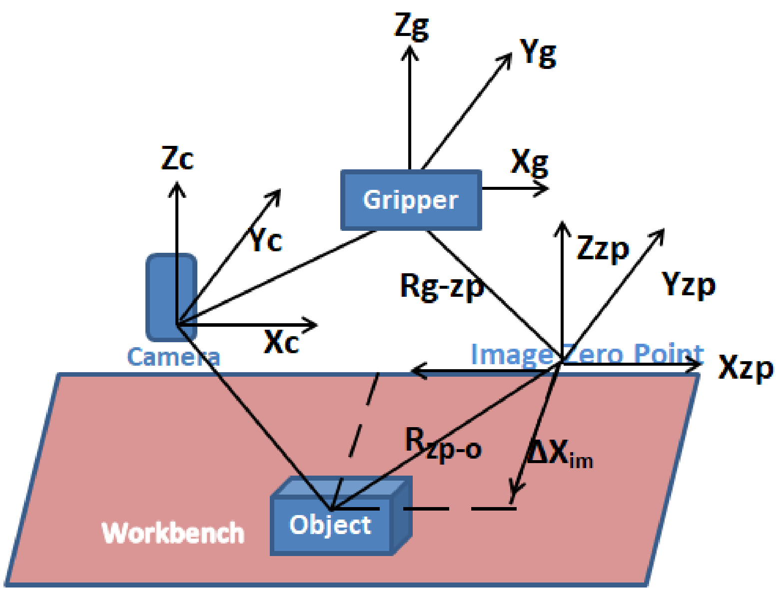


Figure 2. Coordinate system for robot and vision system.

2.4. Force / torque sensing module

When the automation has been reaching its destination concerning offline simulation, the F/T sensing module automatically takes management to observe and modify position errors. It consists of neural networks and mathematical equations that have already been converted into C programs for watching the signals from sensors and causation a correct response to the acting programs.

2.4. Acting programs

The acting program module coordinates the time and working sequence of the image module, the F/T sensing module, and the automation management. It additionally handles system information passing, knowledge extracting and change, and system-level deciding. Getting the simulated robot path knowledge, transferring the knowledge among different coordinate systems, and then transferring the information to the automation controller to regulate automation, the movement also is the responsibility of this module. The information generated by the subsystems is employed to inform the automation of all the task requests and work cell status, like that machine, right to be served, what work piece right to be searched and loaded, which the machine is faulty, and wherever the new machine is extra in.

2.4. Denavit and Hartenberg (DH) parameters

Denavit–Hartenberg (DH) parameters are a convention used by robotics experts to mathematically describe the size and range of a robot's arms.

a. **Joint offset (b):**
   Offset along previous z to the common normal.
b. **Joint angle (θ):**
   Angle about previous z, from old x to new x.
c. **Link length (a):**
   Length of the common normal (aka α, but if using this notation, do not confuse with α). Assuming a revolute joint, this is the radius about previous z.
d. **Twist angle (α):**
   Angle about common normal, from old z axis to new z axis.

Hence, for revolute/prismatic joint, the two terms used for a variable and other three constant parameters are joint variable and link parameters respectively. [9]

2.5 Forward and inverse kinematics

Both forward and inverse kinematics is vital to Artificial Intelligence. Forward kinematics refers to the process of computing the position of the end-effector from a given value for the joint parameters by using the kinematic equations of a robot.

These kinematic equations are used in animation, robotics, and computer games. And the reverse process of achieving a desired position of the end-effector by computing the joint parameters is known as inverse kinematics.

- Joint Variables ––> Pose of the End effector of a Robotic Arm = Forward Kinematics
- Pose of the End effector of a Robotic Arm ––> Joint Variables = Inverse Kinematics

2.5.1 Forward kinematics

Here link parameters (link lengths) and joint variables (typically angles) are given and one has to find out the position and orientation of the end-effector (EE). We used known joint variables (i.e. servo motor angles, displacement of a linear actuator, etc.) to calculate the position and orientation of the end effector of a robotic arm (e.g. robotic gripper, robotic hand, vacuum suction cup, etc.) in 3D space. This is called forward kinematics.

2.5.2 Inverse Kinematics

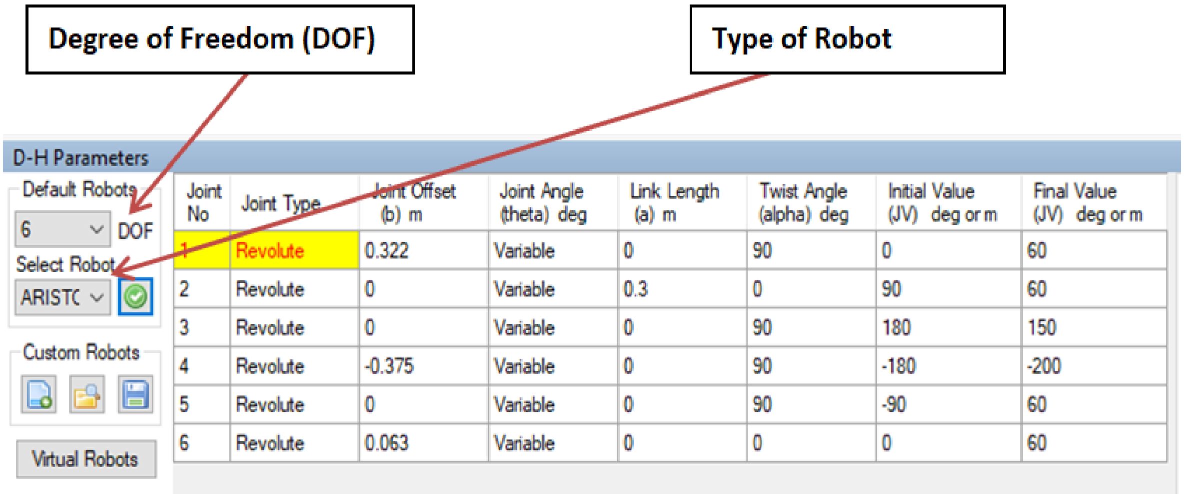
In robotics, inverse kinematics uses the kinematics equations to find the joint settings that allow each of the robot's end-effectors to achieve the required configuration. Motion planning is the process of determining a robot's movement in order for its end-effectors to go from one configuration to another. The motion is transformed using inverse kinematics.

The conversion of the position and orientation of a manipulator end-effector from Cartesian area to joint area is named as inverse mechanics downside.

3. Experimental details

3.1 Selection of Robot:

The ARISTO robot in RoboAnalyzer software gives an advantage over creating a CAD model and then working over it. Importing the robot type from the library: at the preliminary step, we must give the degree of freedom (DOF) as shown in Figure 3. Available robots of every type will appear in the list below. Choosing the working condition of the robot and then incorporates the necessary configurations.



Joint No	Joint Type	Joint Offset (b) m	Joint Angle (theta) deg	Link Length (a) m	Twist Angle (alpha) deg	Initial Value (JV) deg or m	Final Value (JV) deg or m
1	Revolute	0.322	Variable	0	90	0	60
2	Revolute	0	Variable	0.3	0	90	60
3	Revolute	0	Variable	0	90	180	150
4	Revolute	-0.375	Variable	0	90	-180	-200
5	Revolute	0	Variable	0	90	-90	60
6	Revolute	0.063	Variable	0	0	0	60

Figure 3.Model of MTAB ARISTO robot.

Pupils find the mathematics involved in robotics, such as forward and inverse kinematics, difficult to comprehend at first, and teachers find it challenging to explain the essential of robotics mathematics to students. This is owing to the fact that forward and inverse kinematics, for example, need 3D transformations. The industrial robots are also depicted using Denavit and Hartenberg (DH) characteristics, which are difficult to comprehend and visualise in 3D. RoboAnalyzer is designed to help students and instructors overcome the challenges listed above. In other words, study/teach the physics of robotics using the fun of RoboAnalyzer animations before attempting to understand the mathematics of robotics, which is addressed in the book “introduction to robotics” by S. K. Saha. [10]

3.2 Parametric setup for analysis:

The setting up of the parameters for inverse kinematics through the two available options i.e., joint control and cartesian control as shown in Figure 4. The path generated by both methods allows you to compare and analyze the curve. These methods allow you to get full control of the robot and move as per the specific position or angle as described below.

The output of these methods will be the end effector coordinates and the total homogeneous transformation of the robot.

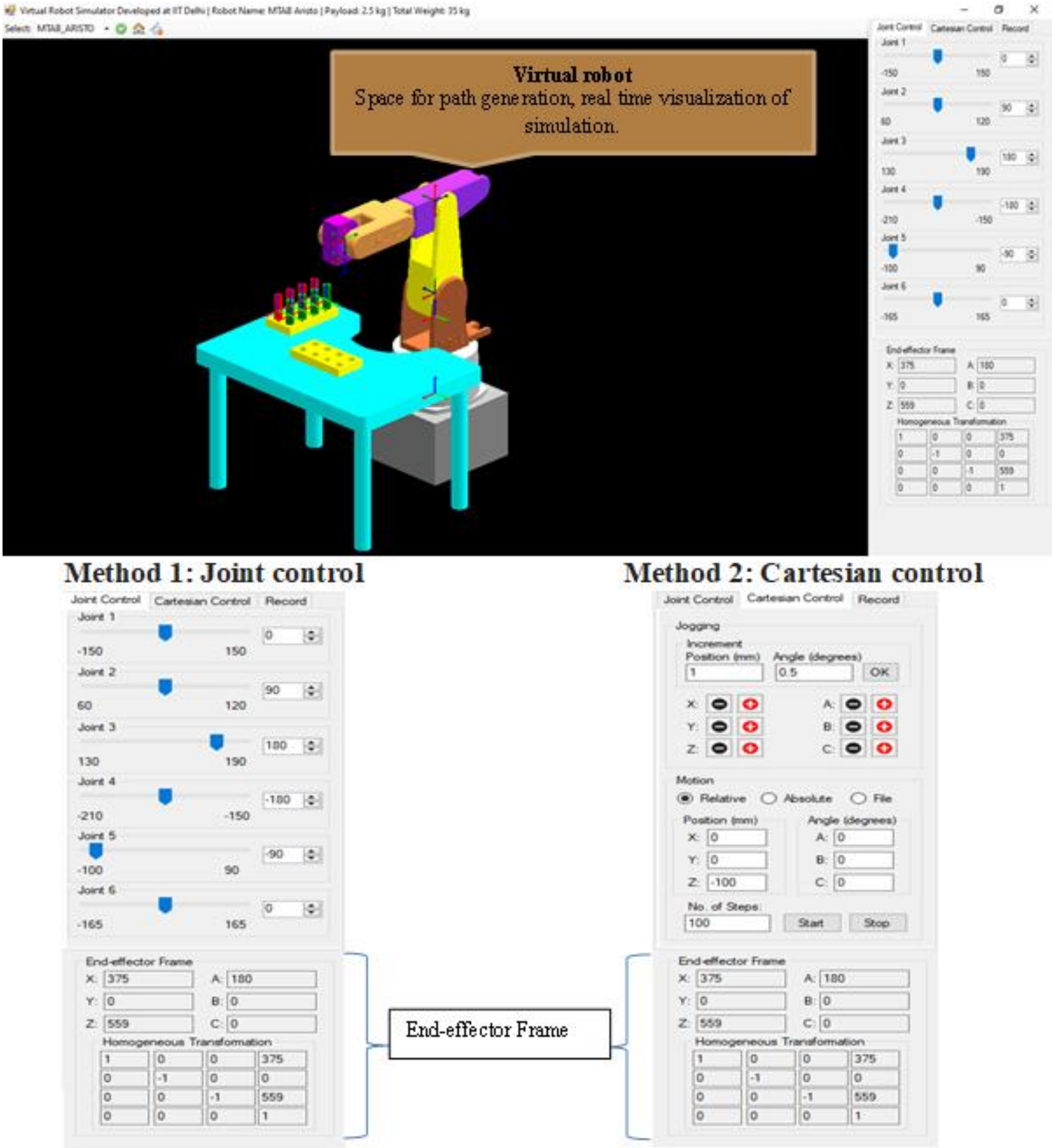


Figure 4. The photographic view shows the motion of robot.

Method 1: Joint control- each joint i.e., joint 1- joint 6 will let you drive the joints in accordance to the angle concerning the links on which it is mounted upon. Respective joints and links let you drive the gripper to the exact position of the object over the workbench.

Angle related to each joint is given below (degree): joint 1: -150 to 150; joint 2: 60 to 120; joint 3: 130 to 190; joint 4: -210 to -150; joint 5: -100 to 90; joint 6: -165to 165.

Method 2: Cartesian control- this technique particularly uses the jogging method with increment in X, Y & Z direction, and motion i.e., relative motion and absolute motion which takes X, Y & Z axis into account for the motion in millimeter (mm) and angle (degree).

4. Results & discussions

4.1 Inverse Kinematics

Figure 5 represents eight different solutions for the position where the end effector has to reach. However, the robot prefers only a single path to approach and reach the target point and keeps all the other seven options in the reading history. This table helps us to detect the variations in all the joint angles which enables us to find the difference between the highest and the lowest values for each joint from all the solutions. It also provides the joint offset (m), link length (m), twist angle (deg) & end effector's position (m) for each joint separately.

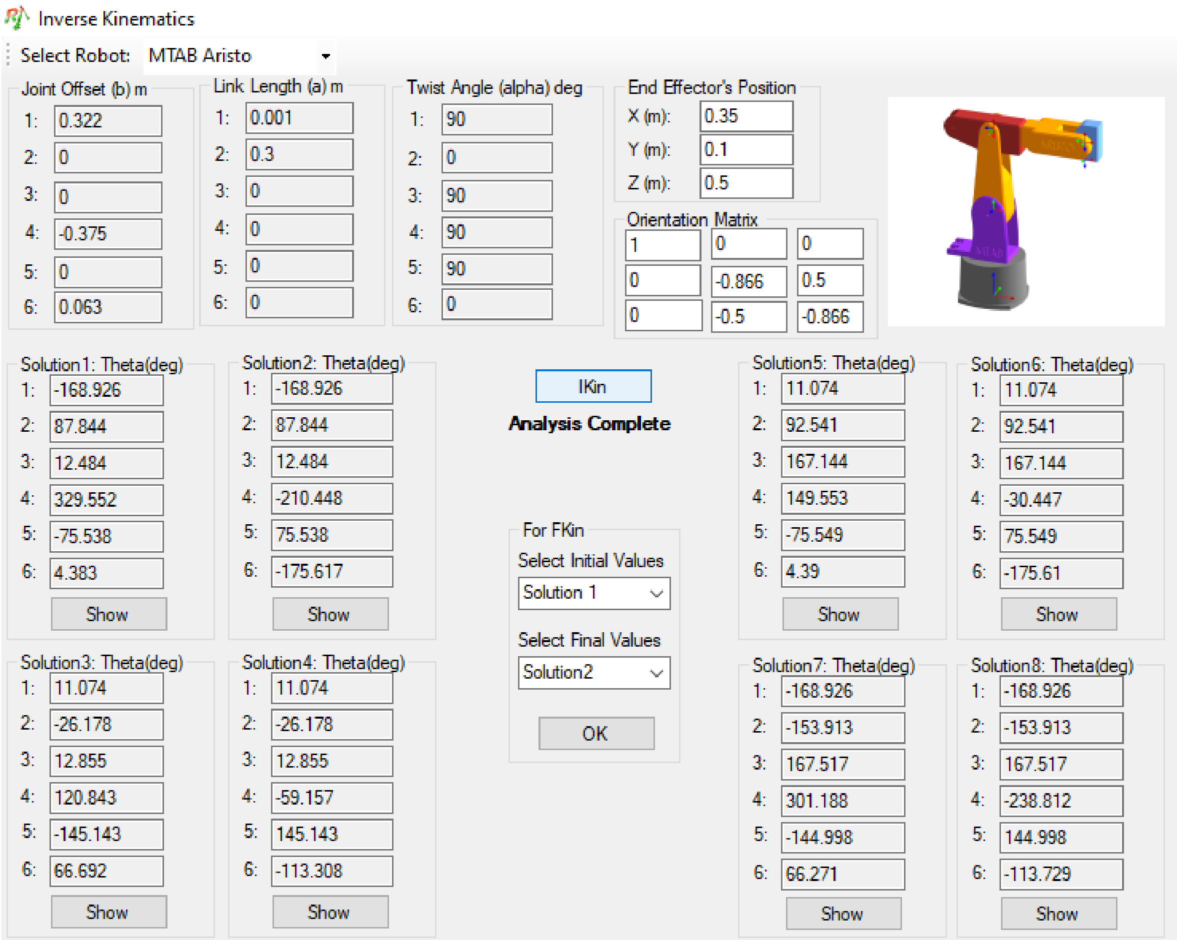


Figure 4. Inverse kinematics of the ARISTO.

Figure 6 shows the line graph which consists of three lines for X, Y & Z-direction representing the end effector distance of each link per unit time (s) respectively. Each link has three lines representing their end effectors' X, Y & Z directional movement. This graphical representation of the data about all the six links of the ARISTO robot gives easy access to compare data with individual link movements, for example, link 1 and link 6; we can compare and see the difference of the distances moved by their links in X, Y & Z directions in a time interval of 0.2 seconds.

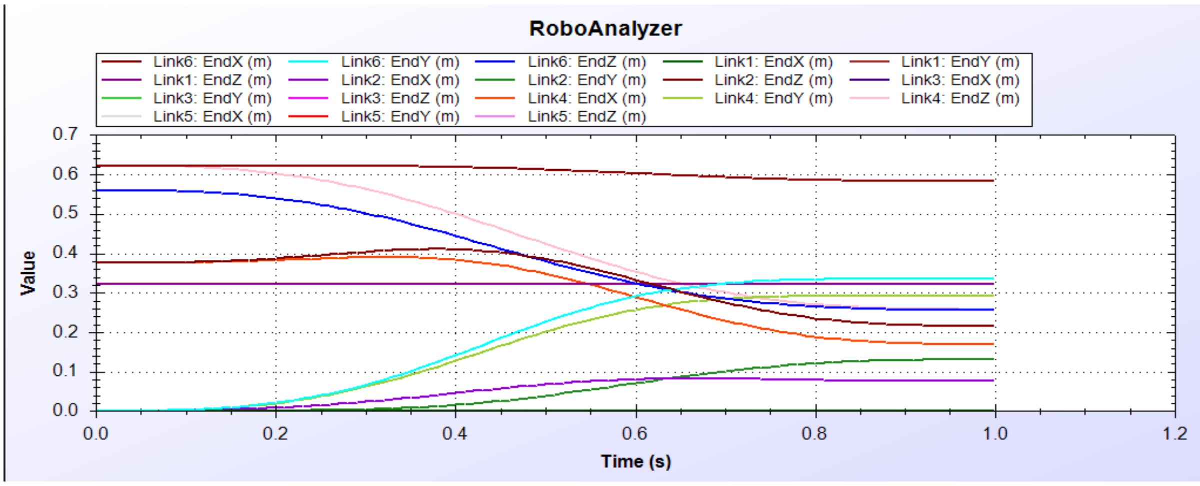


Figure 5. End effector distance in X, Y, Z direction for links 1-6.

Figure 7 shows the information about the joint values, joint velocities, and joint acceleration, and force/torque. The almost straight line shows very low deflection or change from initial values. The sinusoidal wave curves show the acceleration curves for joints 1, 2, and 3 respectively where joint 1 is highly accelerated as compared to joint 2 and joint 3. Joint 3 shows the least acceleration and closest to the origin axis.

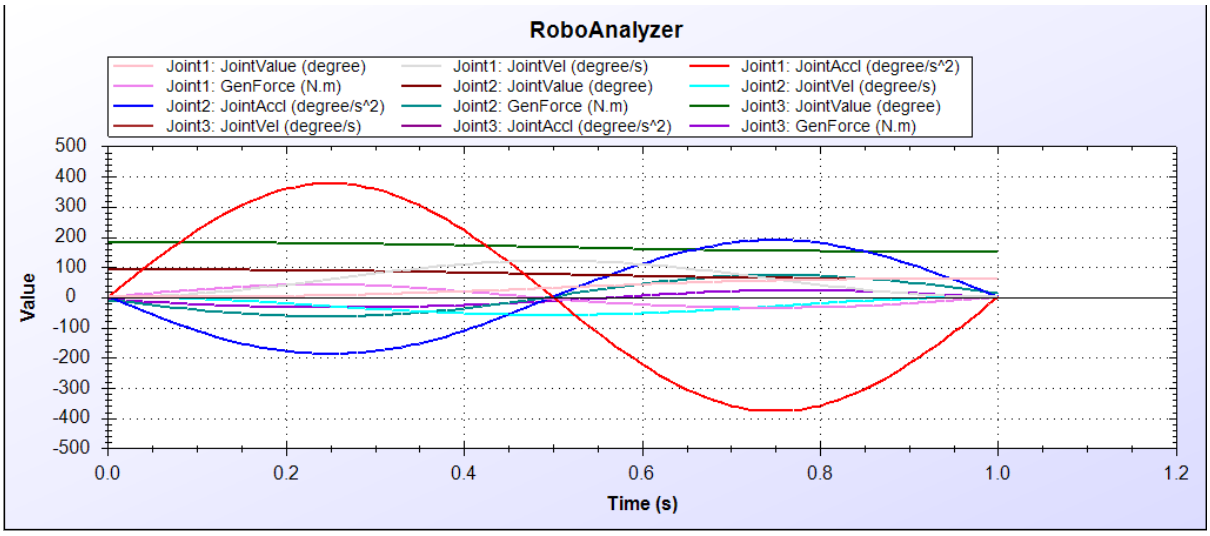


Figure 6. velocity, acceleration for joint 1, 2, and 3.

Figure 8 indicates the difference in the values for the joints that increases as compared to the difference in values for joints 1, 2 & 3 which is almost doubled the values. Joint 5 shows the maximum change in the value of its acceleration as compared to every other joint. Low deflection in lines proves the low movement of the joints as that of joint 4. All of the graphs combined give us the link by link and joint by joint detailed information regarding the velocity, acceleration, and forces with torque. All these results show the faster and reliable solution to any problem concerning any other method, especially when done with real-time simulation and modification of the robot.

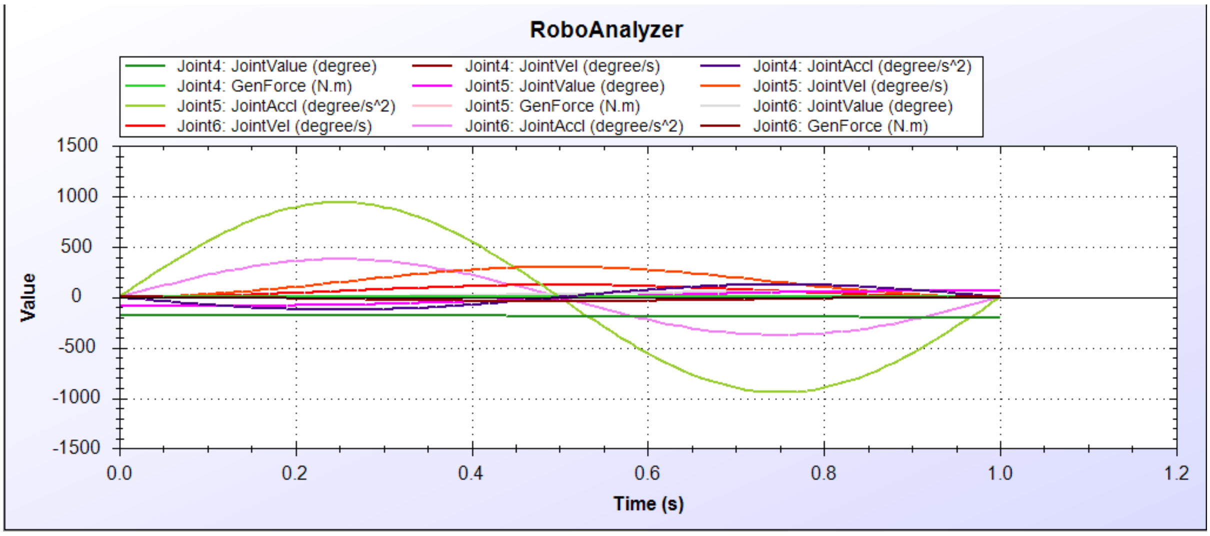


Figure 7. Velocity, acceleration for joints 4, 5, 6.

5. Conclusions

In this work, the RoboAnalyzer software is used to determine the kinematics of the MTAB ARISTO robot i.e. forward kinematics and inverse kinematics. Though theoretical approach consumes a lot of time and can lead to more errors for obtaining the data of each link and joint for individual parameters like force, acceleration, velocity, torque. Existing experiments lead us to only the kinematics solution of the robot, by using the RoboAnalyzer software, we can obtain the values of velocity, acceleration and other corresponding values for links and joints, in addition to the robot end-effector simulation. This robot (MTAB ARISTO) is particularly most suited for operations like pick & place, palletizing, writing, assembly, and welding curvy paths. The major finding from this study is, except for link 6, the end effector distance for all the links ranges under 0.35m. Acceleration for joint 5 is almost double the values from that of joint 1 i.e., the range for joint 1: -400 to 400 and for joint 5: -850 to 850.

6. References

1. Rajeevlochana, C. G., & Saha, S. K. (2011, February). RoboAnalyzer: 3D model based robotic learning software. *In International Conference on Multi Body Dynamics* (pp. 3-13).
2. Gupta, V., Chittawadigi, R. G., & Saha, S. K. (2017). RoboAnalyzer: robot visualization software for robot technicians. *In Proceedings of the Advances in Robotics* (pp. 1-5).

3. A. Jaramillo-Botero, A. Matta-Gomez, J. F. Correa-caicedo and W. Perea-Castro, “ROBOMOSP,” *IEEE Robotics & Automation Magazine*, pp. 62-73, 30 November 2006.
4. Sadanand, O. R., Sairaman, S., Sah, P. B., Udhayakumar, G., Chittawadigi, R. G., & Saha, S. K. (2015, July). Kinematic Analysis of MTAB Robots and its integration with RoboAnalyzer Software. In *Proceedings of the 2015 Conference on Advances In Robotics* (pp. 1-6).
5. Zhang, H., & Zhao, J. (2017). Bio-inspired vision based robot control using featureless estimations of time-to-contact. *Bioinspiration & biomimetics*, *12*(2), 025001.
6. Sampei, M., & Furuta, K. (1988). Robot control in the neighborhood of singular points. *IEEE Journal on Robotics and Automation*, *4*(3), 303-309.
7. Gupta, V., Saha, S. K., & Chaudhary, H. (2019). Optimum design of serial robots. *Journal of Mechanical Design*, *141*(8).
8. WEI, H. Z., Xue, D., JIAO, L. Q., & BAI, W. F. (2012). Error analysis and simulation based on KUKA six-degree robust [J]. *Journal of Changchun University of Technology (Natural Science Edition)*, *3*.
9. Srikanth, A., Ravithej, Y., Sivaraviteja, V., & Sreechand, V. (2013). Kinematic analysis and simulation of 6 DOF of robot for industrial applications. *International Journal Of Engineering And Science*, *3*(8), 01-04.
10. Saha, S. K. (2014). *Introduction to robotics*. Tata McGraw-Hill Education.