ISRES Publishing

The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM)
ISSN: 2602-3199

The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM)

Volume 1, Pages 273-285

ICONTES2017: International Conference on Technology, Engineering and Science

DEVELOPING LINEAR AND NONLINEAR MODELS OF ABB IRB120 INDUSTRIAL ROBOT WITH MAPLESIM MULTIBODY MODELLING SOFTWARE

Onder Tutsoy
Adana Science and Technology University

Ismail Calikusu
Nevşehir Hacı Bektaş Veli University

Sule Colak
Adana Science and Technology University

Orang Vahid
Maplesoft Company

Duygun Erol Barkana
Yeditepe University

Fatma Gongor
Adana Science and Technology University

Abstract: Industrial robots have been extensively used in various industrial applications due to a number of advantages such as accuracy and speed in performing tasks. To achieve complex applications with industrial robots, sophisticated controllers should be developed; henceforth, precise model of the industrial robots must be obtained by using multibody modelling softwares. The purpose of this paper is to create an ABB IRB120 industrial robot representation for simulating and analyzing dynamics and kinematics of the industrial robots by using MapleSim. In addition, this paper presents how linear and nonlinear models of the robot can be obtained and makes available them to public. Therefore, it will be possible to design linear and nonlinear controllers for ABB IRB 120 industrial robot by using the developed models, without requiring any multibody modeling

Keywords: Industrial robots, linear and nonlinear models, MapleSim, modelling, six-DOF ABB IRB120.

Acknowledgement: We would like to thanks Adana Science and Technology University Research Fund under project no MÜHDBF.EEM.2015-3

Introductıon

Industrial robots are able to perform various tasks such as picking, placing and palletizing which humans have traditionally carried out. To achieve these operations efficiently, specific control algorithms should be developed for the industrial robots. However, to analyze and improve designed control algorithms, a model of the physical system is required, where the models either can be created with pen and paper for simple systems or using modelling softwares for more complex systems. Modelling softwares are extensively preferred due to effective and rapid solutions instead of time consuming, tedious modelling with pen and paper [1]. A great number of multibody modeling softwares such as SimMechanics, Robotran, and MapleSim are available.

- Selection and peer-review under responsibility of the Organizing Committee of the conference

*Corresponding author: Onder Tutsoy- E-mail: otutsoy@adanabtu.edu.tr

SimMechanics is a simulation software that provides three-dimensional model of mechanical systems. This software works in the Simulink environment and has extensive component library to represent mechanical systems. In this software, mechanical systems are represented by connected block diagrams that are different from Simulink blocks representing mathematical operations with geometric and kinematic relationship of physical systems directly. In this way, it saves time and effort to obtain the equations of systems. However, SimMechanics requires more time to simplify the equation [2-3]. In addition, it does not provide the linear, nonlinear ordinary differential equations (ODE), which are basically models of the represented physical or mechanical systems.

Another modelling software Robotran is developed at the University of Catholique de Louvain to create model of various multi-body systems. It produces full symbolic generation of the reduced direct-inverse dynamic and kinematics of the systems. It can also generate the model of the multi-body system in the symbolic format and this symbolic format can be utilized with a number of numerical programs such as Matlab [4-5]. However, this software has bugs and also getting technical help from the developers is not generally possible.

MapleSim is a multi-body modelling software developed by Maplesoft Inc. that allows easy modelling of a physical system with its extensive component library. As shown in Figure 1, it is able to model various electrical, electronic, mechanical, hydraulic and magnetic systems. In MapleSim, the symbolic computation technique provides to generate easy and flexible models. In addition, it simplifies the modelling process and reduces the processing time. Symbolic computation technique provides flexibility to develop models quickly and leads to optimal results faster. It enables modeling of multi-body systems from humanoid robots to industrial robots and leads to symbolic inverse and forward kinematics [7]. Therefore, due to these advantages, to obtain linear and nonlinear model of ABB IRB120 industrial robot, MapleSim software is preferred in this paper.

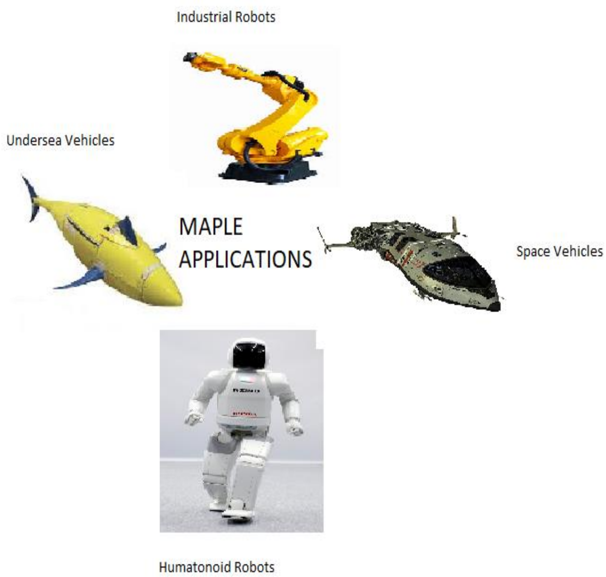

Figure 1. MapleSim applications

In the literature, the mathematical and mechanical modelling of ABB IRB120 industrial robot is performed with various modelling software or manually. However, these robot models do not contain Six-DOF real parameters and generally, three-DOF is used for modelling process. Henceforth, we focus on complete representation of ABB IRB 120 and getting its accurate linear-nonlinear model in this paper.

In the rest of the paper, section 2 introduces overall MapleSim architecture and presents step by step representation of ABB IRB120 industrial robot in MapleSim. Section 3 explains nonlinear model generation process and section 4 states linear model generation process of the represented ABB IRB120 in MapleSim. Finally, the last section presents conclusion and future works.

Modelling Abb Irb120 Industrial Robot

In this part of the paper, initially MapleSim multi-body modelling software is briefly introduced and then a realistic model of ABB IRB 120 industrial robot is performed step by step.

2.1 MapleSim: An Overview

It is possible to model various higher order, complex and coupled mechanical systems by using sophisticated Modelica based MapleSim library. To build a mechanical system in MapleSim, related components of the mechanical systems are taken from the MapleSim library and added to the MapleSim worksheet. For example, to design a simple unstable pendulum; one fixed reference frame, two links, one mass and one joint are dragged from the library into the MapleSim worksheet and then they are connected to each other as in physical world. These are illustrated in Figure 2.

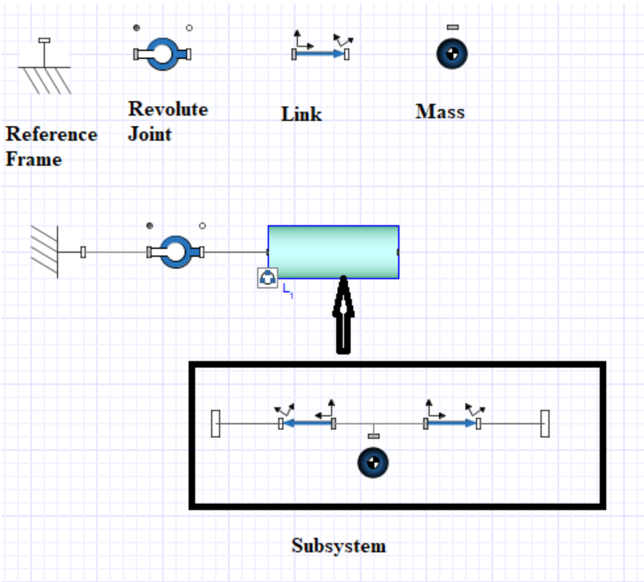

Figure 2.Simple mechanical system representation with MapleSim

Represented systems in MapleSim can be symbolically or numerically analyzed by using Maple software properties. To perform various analysis of mechanical systems in time domain or frequency domain, Maple based custom components are available. For example, nonlinear ODE of the mechanical system can be obtained and in terms of creating sub-systems, the nonlinear model of the system can be linearized around equilibrium or operating points [8]. In order to obtain mathematical model of a mechanical system, MapleSim follows three steps as explained next.

2.1.1 MapleSim Architectures

As shown in Figure 3, MapleSim architecture consists of three stages; representation of a physical system, generation of nonlinear equations of the physical system and creation of the nonlinear ODE based model.

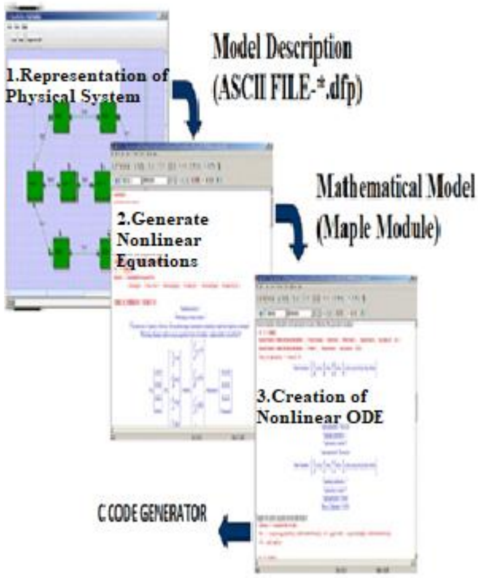

Figure 3.The maplesim architecture

When representing multi-body systems in MapleSim, symbolic values can be assigned for the components of the physical system; so that symbolic nonlinear equations of the system can be generated. This is an important advantage for accurate system representation since numeric based models cause numerical errors and summation of this error leads to significant modelling error, which cannot be eliminated and can be a detrimental drawback for designing efficient controllers. Therefore, model of multi-body mechanical systems should be generated symbolically and numerical values should be assigned when numerical simulation is performed.

It is important to note that the duration of the simulation for a particular system depends on various points. For example, if the generated model contains unnecessary mathematical representations such as insignificant poles and zeros, then numerical simulation takes longer time. To address this problem, MapleSim is able to simplify the generated equations at certain amount. In MapleSim, to represent higher order, complex and coupled systems such as industrial robots, certain steps must be followed. These steps are explained in next section for ABB IRB120 industrial robot.

2.2 Representation of ABB IRB 120 With MapleSim

In this section, six DOF ABB IRB120 industrial robot is represented in MapleSim by taking into account the necessary physical constraints and laws. The corresponding steps are as follows:

1. Firstly, click on MapleSim icon; then, MapleSim worksheet will appear. Save it as ABBIRB120.msim into a designated file. Now, as a first step, add a reference frame, which is called as 'Fixed Frame' in the MapleSim library.

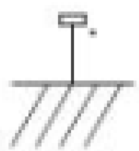

Figure 4. The fixed frame from maplesim library

2. As a second step, the base of the industrial robot should be located at the top of the reference frame. To represent the base as a physical system in MapleSim, consider it as a single link pendulum without a joint. To do that, select two links and one mass from the MapleSim library, and then drag them into the worksheet. In real life, a system like a pendulum has a link and a mass where the mass is located at the center of the link. To represent this system in MapleSim, we need to use two links rather than one link to locate the mass at the centre

of the real link. Now, the links and mass can be connected as shown in Figure 5. After the fixed frame is added, the base of the six DOF industrial robots is placed in workspace.

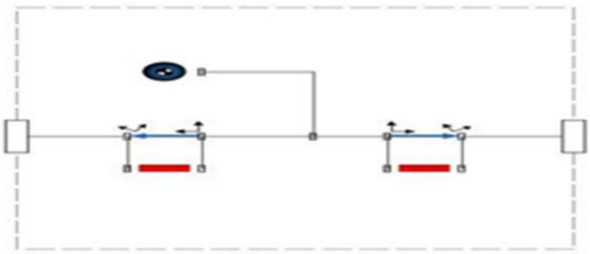

Figure 5.Base link

To simplify physical appearance of the base, a sub-system called as 'Base' can be created.

3. After connection of the base components, length of the links and mass of the base are assigned. To do that, initially direction of the base length in real physical robot is decided. Then, corresponding MapleSim axis is determined by using MapleSim visualization tool. Now, symbolic length and mass variables are written into the dedicated areas. These symbolic length and mass must be introduced into the 'parameters view' section. It is important to note that to assign full length of the base, the two links must have half of the full length where the left link has positive sign whereas the right link has negative sign. Therefore, with respect to the centre of the mass, the overall length of the base is the real base length. The Figure 6 illustrates how the symbolic length of the left link is assigned.

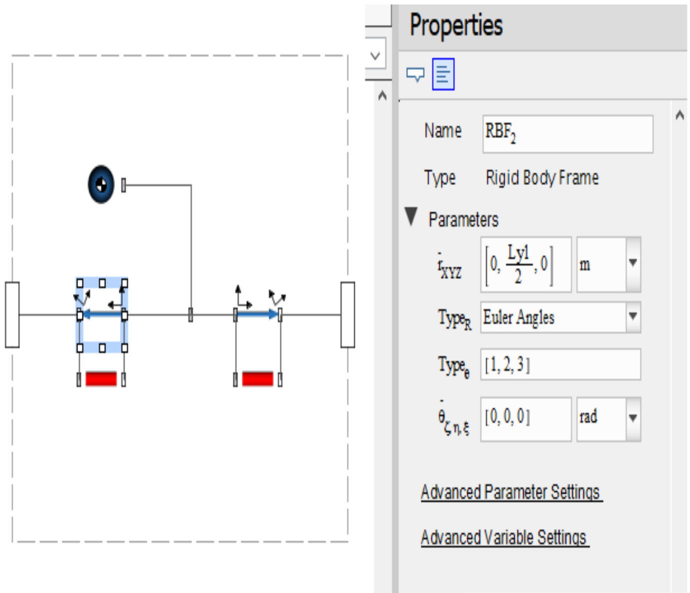

Figure 6.Assigning length parameter of the base for the left link

Similarly, the other length is assigned and mass value is written to the dedicated area. This is shown in Figure 7.

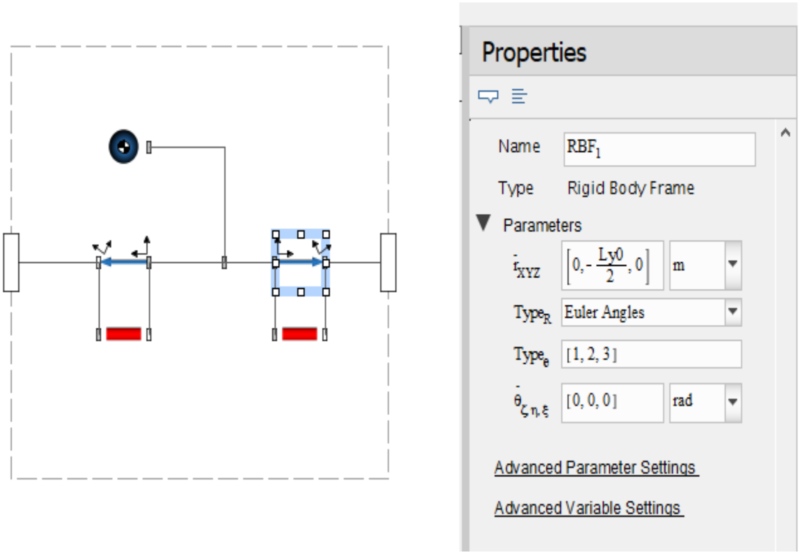

Figure 7. Assigning length parameter of the base for the right link

As stated before, the connection of the links and mass for the base can be located into a sub-system as can be seen from Figure 8.

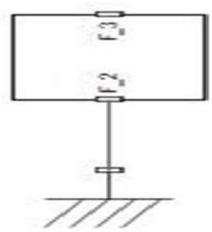

Figure 8.Base link subsystem with reference frame

1. After creating the base of the robot, a joint should be chosen. Since ABB IRB120 has revolute joints with motion constraints, a revolute joint from the MapleSim library is selected. The direction of the movement of the joint is determined based on the movement of the real robot. Since the physical industrial robot has revolute joints for motion, we add a revolute joint on the top of the base. The base of the robot with the first revolute joint is shown in Figure 9.

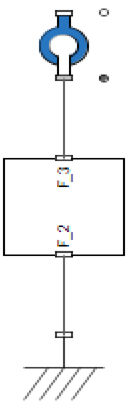

Figure 9. Base of the robot with first revolute joint

2. The representation of the other links, masses and joints follow the same procedure. It is crucial to note that the most important point to take into account when representing a multi-body system such as robot is the

determination of the axes of the lengths and directions of the joints. The overall representation of the ABB IRB120 industrial robot is shown in Figure 10.

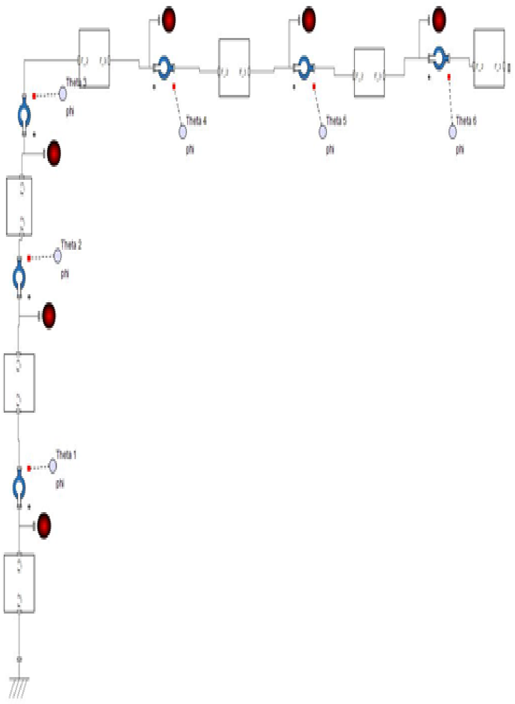

Figure 10.ABB IRB120 industrial robot representation with MapleSim

1. Once the system is represented in MapleSim, 3D view becomes available as shown in Figure 11.

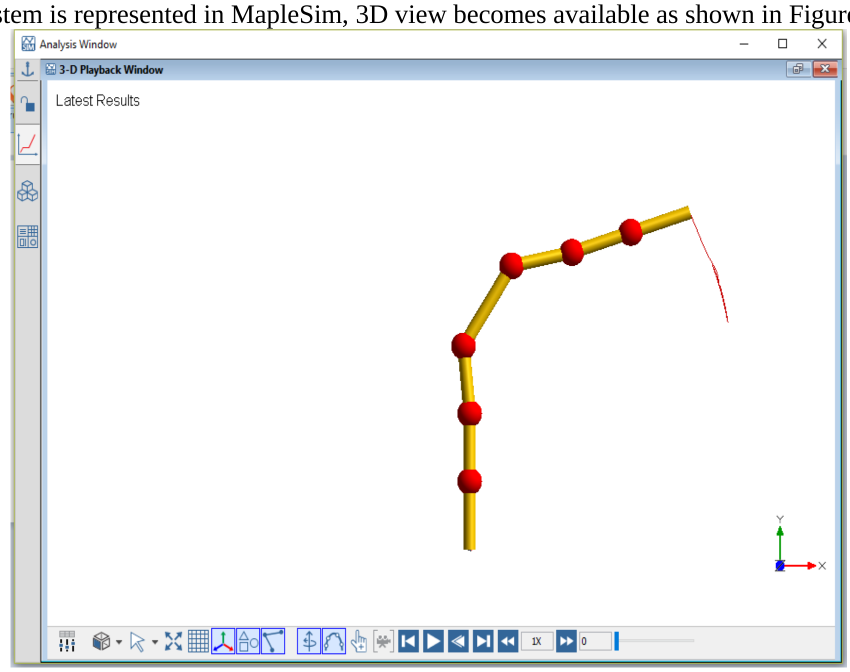

Figure 11.ABB IRB120 3D View

For more realistic view of the robot, CAD components can be added; however, this does not affect the dynamics or nonlinear, linear model of the system.

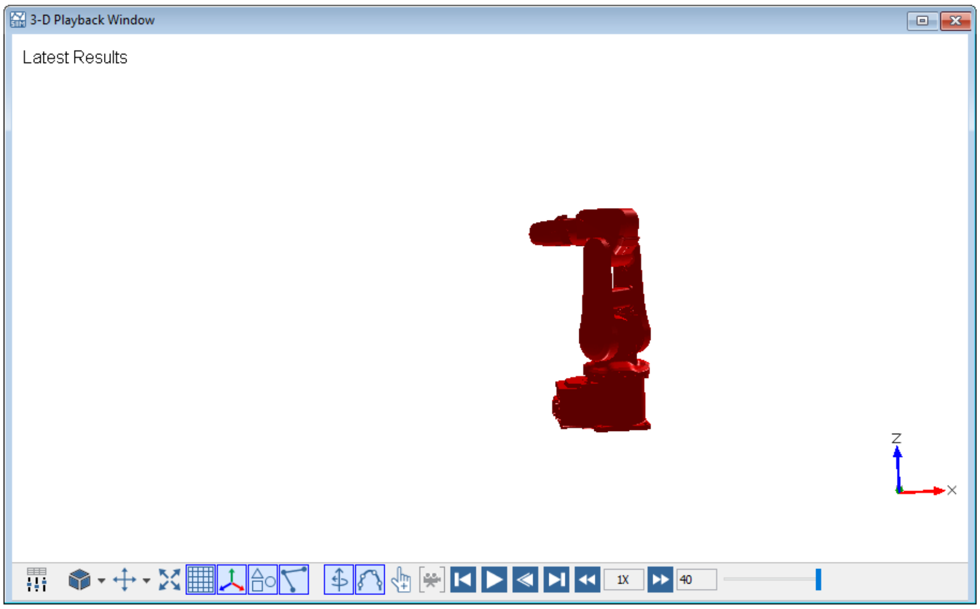

Figure 12.3D view of ABB IRB120 model using CAD models

Since the representation of the robot is available, the next step is determination of nonlinear equations of the robot.

Attaining Nonlinear Equations of the Robot

Having accurate nonlinear equations of a physical system is crucial for designing model based nonlinear controllers. To have nonlinear equations of the robot, symbolic variables must be replaced with their correct values. To have accurate model, real parameters such as lengths and masses of the robot must be substituted with their symbolic values. Table 1 shows the lengths and masses of the ABB IRB 120.

Table 1. Real parameters of ABB IRB120 industrial robot

Parameter Name	**Value**	**Description**
Ly0	0.145	Base length in y direction
Ly1	0.145	Link1 length in y direction
Ly2	0.27	Link2 length in y direction
M0	6,215	Mass of the base
M1	3,060	Mass of the link1
M2	3,908	Mass of the link2
M3	2,940	Mass of the link 3
Lx3	0.134	Link3 length in x direction
Ly3	0.07	Link3 length in y direction
Lx4	0.168	Link 4 length in x direction
M4	1,320	Mass of link 4
M5	0.546	Mass of Link 5
Lx5	0.072	Link 5 length in x direction
Le	0.02	End effector length in x direction
M6	0.0136	End effector Mass

To obtain the nonlinear equations or model, initially the physical representation of the robot is transferred to a worksheet. Various approaches, such as Newtons second law and Lagrange, are available in literature to derive nonlinear ODE of a physical system. Maple uses Lagrange equations, which is based on the difference between kinetic and potential energies, to derive nonlinear ODE. Next section presents a verification example to show that it is possible to have exact nonlinear equation of a simple system.

Verification Example

As an example, nonlinear equation of a simple pendulum, shown in Figure 13, will be derived by using Lagrange method and then this analytically obtained nonlinear ODE will be compared with the one provided by MapleSim.

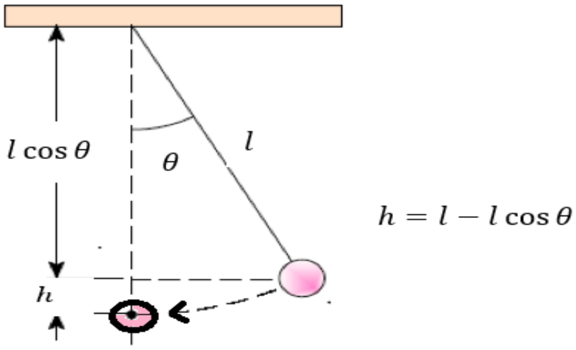

Figure 13.Simple pendulum

Assume that the parameters of the simple pendulum are as follows;
l : length,
m : mass,
g : gravity,
θ : position angle of the pendulum with respect to vertical line.
Firstly, kinetic (T) and potential (U) energies of simple pendulum are expressed.

$$T = \frac{1}{2}mv^2$$
$$= \frac{1}{2}m(l\dot{\theta})^2$$

Eq. 1

where $v = l\dot{\theta}$

$$U = mgh$$
$$= mgl(1 - \cos\theta)$$

Eq. 2

where $h = l - l\cos\theta$

Then, Lagrangian function (L) can be formed as;

$$L = T - U = \frac{1}{2}mL^2\theta^2 - mgl(1 - cos\theta)$$

Eq. 3

The next step is to use Lagrangian formulation defined as;

$$\frac{d}{dt}\frac{dL}{d\dot{\theta}} - \frac{dL}{d\theta} = 0$$

Eq. 4

After simplification process, the nonlinear equation of the simple pendulum is obtained as;

$$\ddot{\theta}(t) = -\frac{g}{l}sin\theta(t)$$

Eq. 5

This result should be compared with the derived by MapleSim.

$$\frac{d^2}{dt^2}\theta_{R1}(t) + \frac{g}{l}\sin(\theta_{R1}(t)) = 0$$

Eq. 6

As can be seen from the equations Eq. 5 and Eq. 6 , the results are identical. Therefore, it can be deduced that MapleSim is capable to represent physical systems accurately.

Forced Nonlinear ODE of ABB IRB 120

Similar to the simple pendulum, MapleSim is able to derive nonlinear equations of more complex systems such as industrial robot in this form;

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + g(\theta)\theta = B * u$$
Eq. 7

where $M(\theta)$ is the mass matrix or inertia matrix, $N(\theta,\dot{\theta})$ is the centrifugal or Coriolis matrix , $g(\theta)$ is the gravitional forces matrix , u is the control signal and B is the control signal coefficient. For the sake of simplicity all t terms are suppressed.

It is not possible to obtain centrifugal and Coriolis matrix together with gravitational forces matrix directly in MapleSim. Thus, extra Maple code must be written to obtain them.

To attain the mass matrix of the nonlinear ODE;

M := multi_body:-xM:

To attain the matrix for gravitional forces;

eqs := multi_body:-vF;
map(coeffs, collect(eqs, [g], 'distributed'), [g]);

To attain the matrix for centrifugal and Coriolis forces;

```
NN := numelems(MB:-vQ);
Mass := MB:-xM;
eqs := MB:-vF;
col_g := map(coeffs, collect(eqs, [g], 'distributed'), [g]);
gTerms := Vector(NN);
for i to NN do
 if numelems([col_g[i]]) > 1
 then gTerms[i] := col_g[i][-1]
 end if
end do;
gTerms*g;
 Vector[column](%id = 18446746608836023358)
qdotTerms := simplify(-g*gTerms+eqs);
```

It might be important to note that to extract these matrices the ABB IRB 120 representation in MapleSim must be converted into a subsystem with each joint has both the input and output as shown in Figure 14.

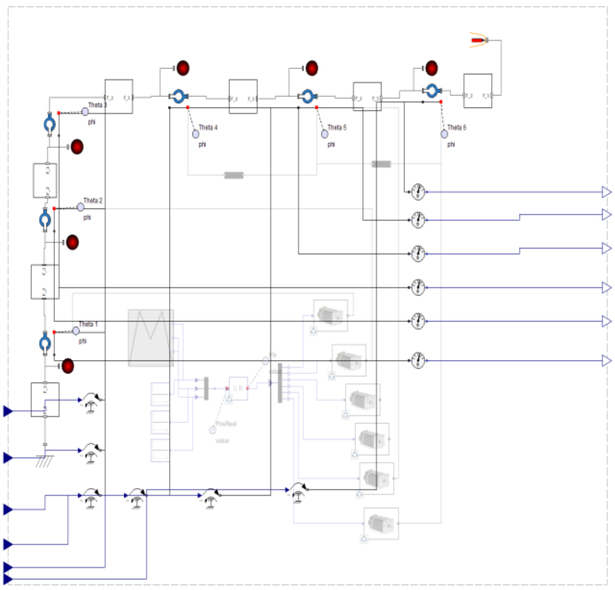

Figure 14. Subsystem of ABB IRB120 with inputs and outputs

Now, it is possible to perform linearization around operating points to obtain the linear model of the system.

Linearization of Nonlinear Systems

Linearization of nonlinear systems is necessary to design linear controllers, which are extensively studied and applied in industry. To understand linearization process, we assume that a general nonlinear model of a multi-body system has n state variables, m input and r output variables;

$$\dot{X}_1 = f_1(\theta_1, \dots \dots \theta_n, u_1, \dots . u_m)$$

$$\vdots$$

$$\dot{X}_n = f_n(\theta_1, \dots \dots \theta_n, u_1, \dots . u_m)$$
$$Y_1 = g_1(\theta_1, \dots \dots \theta_n, u_1, \dots . u_m)$$
$$\vdots$$
$$Y_r = g_r(\theta_1, \dots \dots \theta_n, u_1, \dots . u_m)$$

This nonlinear model can be defined with vector notation as follows;

$$\dot{X} = f(\theta, u)$$
$$Y = g(\theta, u)$$

where function f and g are often assumed smooth enough which means having continuous derivatives to certain order. The linearization process based on the Taylor Series Expansion of a nonlinear function about a specified operating point. Taylor series expansion is given below;

$f(x) \cong f(x_e) + J_f(x_e)(x - x_e)$

Eq. 8

where the $\Delta x = x - x_e = \theta$ is the change in position angle from the operating point. The Jacobian matrix is defined as;

$$J_f(x_e) = \begin{bmatrix} \frac{df_1}{d\theta_1} & \frac{df_1}{d\theta_2} & \dots & \dots & \frac{df_1}{d\theta_n} \\ \frac{df_2}{d\theta_1} & \frac{df_2}{d\theta_2} & \dots & \dots & \frac{df_2}{d\theta_n} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \frac{df_n}{d\theta_1} & \frac{df_n}{d\theta_2} & \dots & \dots & \frac{df_n}{d\theta_n} \end{bmatrix}$$

As systems become more complex, representing them with differential equation becomes cumbersome. Thus, representing the systems with multiple inputs and outputs in state space form is more appropriate. The state space equation has a state equation and an output equation as follows;

$$\dot{\theta} = A\theta + Bu$$
$$Y = C\theta + Du$$

where the first equation is called state equation and the second equation is called output equation.

The MapleSim program automatically performs the above-mentioned operations for multi-body systems. For this purpose, the linearization template shown in Figure 15 can be used.

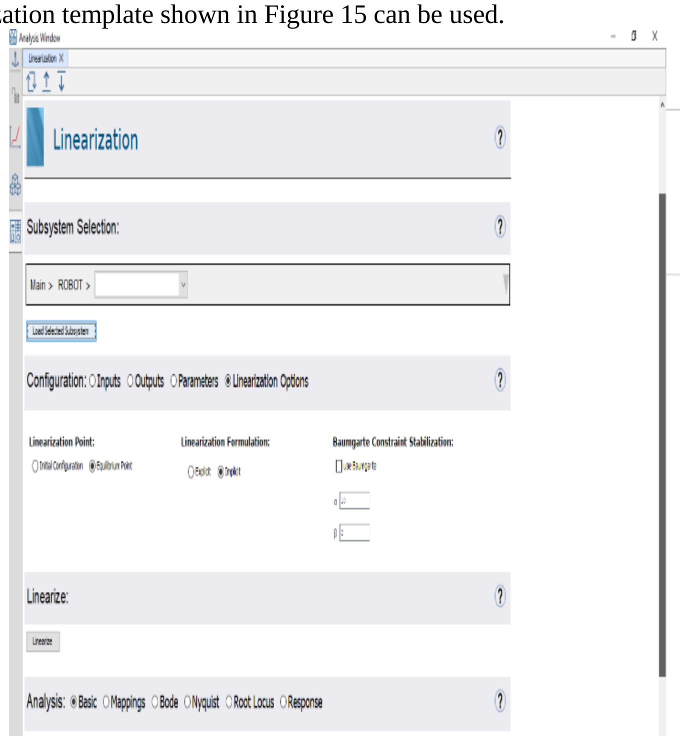

Figure 15.Maple linearization template

After the linearization process, A, B, C, D state space matrices are obtained and given in appendix. After the linearization process, Bode, Nyquist, Root locus plots can be drawn to analyze linearized system and display the effects of different inputs on the outputs of ABB IRB120 robot.

Conclusion

Accurate modeling of industrial robots provides fast solution for testing and developing control algorithms. In this study, modeling of ABB IRB120 industrial robot is realized with MapleSim software, nonlinear equations are generated and linearized by using real robot parameters. In future studies, new model and learning-based control algorithms will be developed for this modelled robot.

References

Rigatos, G. G. (2009). Model-based and model-free control of flexible-link robots: A comparison between representative methods. Applied Mathematical Modelling, 33(10), 3906-3925.

Piotrowski, N. O. R. B. E. R. T., & Barylski, A. D. A. M. (2014). Modelling a 6-DOF manipulator using Matlab software. Archives of Mechanical Technology and Automation, 34(3), 45-55.

Asif, U., & Iqbal, J. (2013). Modeling, Simulation and Motion Cues Visualization of a Six-DOF Motion Platform for Micro-Manipulations. In Advanced Engineering and Computational Methodologies for Intelligent Mechatronics and Robotics (pp. 137-151). IGI Global.

Tutsoy, O., Brown, M., & Wang, H. (2012, September). Reinforcement learning algorithm application and multi-body system design by using MapleSim and Modelica. In Advanced Mechatronic Systems (ICAMechS), 2012 International Conference on (pp. 650-655). IEEE.

Docquier, N., Poncelet, A., & Fisette, P. (2013). ROBOTRAN: a powerful symbolic generator of multibody models. Mechanical Sciences, 4(1), 199-219.

Geddes, K. O., & Fee, G. J. (1992, August). Hybrid symbolic-numeric integration in MAPLE. In Papers from the international symposium on Symbolic and algebraic computation (pp. 36-41). ACM.

Ghozlane, W., & Knani, J. (2016, December). Modelling and simulation using mathematical and CAD model of a robot with six degrees of freedom. In Control Engineering & Information Technology (CEIT), 2016 4th International Conference on (pp. 1-5). IEEE.

Shah, H., Tripathi, S., Lee, L. F., & Krovi, V. (2010). Role of Automated Symbolic Generation of Equations of Motion to Enhance Robotics Education. ASEE Computers in Education Journal.

Švaňa, J. (2016). Modelování s využitím Maple a MapleSim (Doctoral dissertation, Masarykova univerzita, Fakulta informatiky).

Tripathi, S. (2010). Role of symbolic computation in linear and model-based controller development (Doctoral dissertation, Faculty of the Graduate School of The University at Buffalo, State University of New York).

APPENDIX

State Space Matrices

$$A=\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0{,}088385 & 0 & -1{,}2609 & 0 & -0{,}03054 & 0 & 0{,}452842 & 0 & 0 & 0\\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 40{,}15217 & 0 & -29{,}5418 & 0 & 0{,}108109 & 0 & 1{,}133512 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & -44{,}287 & 0 & 108{,}2842 & 0 & -0{,}56953 & 0 & -7{,}27049 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & 0 & -0{,}12654 & 0 & 2{,}01236 & 0 & 1{,}614214 & 0 & 2{,}475777 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0\\
0 & 0 & 11{,}61132 & 0 & -186{,}517 & 0 & 7{,}514058 & 0 & 129{,}9011 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\\
0 & 0 & 0{,}066928 & 0 & -1{,}24236 & 0 & -1{,}58413 & 0 & -2{,}83914 & 0 & 0 & 0
\end{bmatrix}$$

$$B=\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0\\
16{,}52053 & 0{,}075275 & -0{,}35229 & -12{,}2955 & 3{,}515335 & -1{,}98881\\
0 & 0 & 0 & 0 & 0 & 0\\
0{,}075275 & 3{,}888116 & -8{,}51171 & -0{,}11933 & 11{,}05329 & 0{,}073043\\
0 & 0 & 0 & 0 & 0 & 0\\
-0{,}35229 & -8{,}51171 & 25{,}92747 & 0{,}535638 & -58{,}6871 & -0{,}30182\\
0 & 0 & 0 & 0 & 0 & 0\\
-12{,}2955 & -0{,}11933 & 0{,}535638 & 315{,}0428 & -3{,}43332 & -304{,}069\\
0 & 0 & 0 & 0 & 0 & 0\\
3{,}515335 & 11{,}05329 & -58{,}6871 & -3{,}43332 & 785{,}6144 & 0{,}587171\\
0 & 0 & 0 & 0 & 0 & 0\\
-1{,}98881 & 0{,}073043 & -0{,}30182 & -304{,}069 & 0{,}587171 & 337005{,}8
\end{bmatrix}$$

$$C=\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{bmatrix}$$