TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies

Robot Dynamics and Control Based on Exponential Matrices

Diploma thesis

Study program:	N2612 Electrical Engineering and Informatics
Specialization:	Mechatronics

Author:	**Aleksandr Fedorov**
Supervisor:	doc. Mgr. Ing. VáclavZáda, CSc.
Consultant:	Ph.D., Lobanov Vladimir

Liberec 2015

Technical University of Liberec
Faculty of Mechatronics, Informatics and Interdisciplinary Studies
Academic year: **2014/2015**

DIPLOMA THESIS ASSIGNMENT

(PROJECT, ART WORK, ART PERFORMANCE)

First name and surname:	**Aleksandr Fedorov**
Study program:	**N2612 Electrical Engineering and Informatics**
Identification number:	**M13000308**
Specialization:	**Mechatronics**
Topic name:	**Robot Dynamics and Control Based on Exponential Matrices**
Assigning department:	**Institute of Mechatronics and Computer Engineering**

Rules for elaboration:

1. Comparative analysis of methods of robot kinematics and dynamics.
2. Studying the well known methods of robot control.
3. Realization of a dynamical model of an industrial robot using exponential matrices.
4. Development of a technique for designing of robust robot controller using the theory of stability of non-linear systems.
5. Simulation of the robot controller and robot dynamics by Matlab on a chosen robot model.
6. Documentation and description of the created control system of robots.

Scope of graphic works:	**In respect to the documentation needs**
Scope of work report (scope of dissertation):	**c. 40–50 pages**
Form of dissertation elaboration:	**printed/electronical**
Language of dissertation elaboration:	**English**

List of specialized literature:

[1] **Siciliano B., Khatib O. (Eds.). Handbook of Robotics. Berlin, Springer-Verlag, 2008.**

[2] **Angeles J. Fundamentals of Robotics Mechanical Systems. New York, Springer-Verlag, 2003.**

[3] **Murray R. M., Li Z., Sastry S. S. A Mathematical Introduction to Robotic Manipulation. www.cds.caltech.edu/ murray/books/MLS/pdf/mls94-complete.pdf**

[4] **Canudas de Win C., Siciliany B., Bastin G.. Theory of Robot Control. Springer-Verlag, 1996.**

Tutor for dissertation:	**doc. Mgr. Ing. Václav Záda, CSc.** Institute of Mechatronics and Computer Engineering
Dissertation Counsellor:	**Vladimir Lobanov, Ph.D.** Saratov State Technical University, Russia
Date of dissertation assignment:	**10 October 2014**
Date of dissertation submission:	**15 May 2015**

prof. Ing. Václav Kopecký, CSc.
Dean

L.S.

doc. Ing. Milan Kolář, CSc.
Department Manager

Liberec, dated: 10 October 2014

Declaration

I hereby certify that I have been informed the Act 121/2000, the Copyright Act of the Czech Republic, namely § 60 - Schoolwork, applies to my master thesis in full scope.

I acknowledge that the Technical University of Liberec (TUL) does not infringe my copyrights by using my master thesis for TUL's internal purposes.

I am aware of my obligation to inform TUL on having used or licensed to use my master thesis; in such a case, TUL may require compensation of costs spent on creating the work at up to their actual amount.

I have written my master thesis myself using literature listed therein and consulting it with my thesis supervisor and my tutor.

Concurrently I confirm that the printed version of my master thesis is coincident with an electronic version, inserted into the IS STAG.

Date: 15.05.2015

Signature: 

Abstract

In this work was accomplished a review and comparison of the methods which allows make the kinematic and dynamic models. We can distinguish two ways:

- Classic and the most common way of representing a multi-link manipulator. In case of kinematic model it is algorithm Denavit-Hartenberg and the homogeneous transformation matrix, as well as the recursive method based on Newton's equations for dynamic model.
- An alternative way of representing the multi-link manipulator, which is based on the exponential matrices for the kinematic and dynamic model.

I had carried out analysis of manipulator ABB IRB 140. All researching was accomplished on base of this manipulator. Also compiled system description parameters, which required for mathematical model.

Calculations were made using two different methods. On the basis of the results compiled two dynamic models describing the manipulator.

I had done simulation and comparison the obtained characteristics based on the determined models.

Key words

Robot, Dynamics, Control, Kinematics, Models, Denavit-Hartenberg, Newton-Euler, Lagrange-Euler, Exponential Matrices.

CONTENT

Outline

- **Chapter 2**: Fundamental background theory and notation used throughout the thesis are explained in this chapter. It is put importance on the standard convention of how to interpret robot manipulators, as well as the concept of rotation matrices.
- **Chapter 3**: This chapter presents different approaches on dynamic modeling of robot manipulators, and compares the Newton-Euler formulation to the product of exponential formula.
- **Chapter 4**: Based on determined parameters, using the method based on Newton-Euler formulation and method based on the product of exponential formula, we determine the equations which compose the dynamic model.
- **Chapter 5**: This chapter describing the simulation system in the case of open loop and closed loop with PD-controller.
- **Chapter** 6: This chapter execute the comparison of results between classical dynamic model and dynamic model based on product of exponential formula..
- **Chapter 7**: This chapter represent conclusion of this work.

1. Introduction

Robotics is concerned with the study of machines that can replace human beings. The goal of this introductory chapter is to express the motivation behind the thesis, and to give an overview of the contents. The IRB 140 is introduced, as well as the objective and the software that has been used to solve it. An outline and the contributions of the thesis is presented in the end of the chapter.

1.1. History and Motivation

The English term *robot* was derived from the Czech word *robota* that means executive labor, and was first introduced by the Czech playwright Karel Capek in his 1921 play *Rossum's Universal Robots.* Since then the term has been applied to virtually anything that operates with some degree of autonomy, usually under computer control. An official definition of the term, dated to 1980, comes from the Robot Institute of America (RIA) and reflects today status of robotics technology:

A robot is a reprogrammable, multifunctional manipulator designed to move material, parts, tools, or specialized devices through variable programmed motions for the performance of a variety of tasks.

In the early 1980's, robot manipulators were touted as the ultimate solution to automated manufacturing. Predictions were that entire factories of the future would require few, if any, human operators. It turned out that these predictions were a little exaggerated, as the savings in labor costs often did not outweigh the development costs of creating robot systems. Quite simply, people are good at what they do, and installing a robot involves complex systems integration problems. As a result, robotics fell out of favor in the late 1980's.

A resurgence of interest in robotics can be witnessed in the recent years. Deeper understanding of the subject and new technology have made it possible for robots to explore the surface on Mars, locate sunken ships, searching out land mines, and finding victims in collapsed buildings. In an industrial environment the advantages of robots are reduction of manufacturing costs, increase of productivity, improvement of quality standards, and the possibility of eliminating harmful tasks for human operators.

1.2. The description of IRB 140

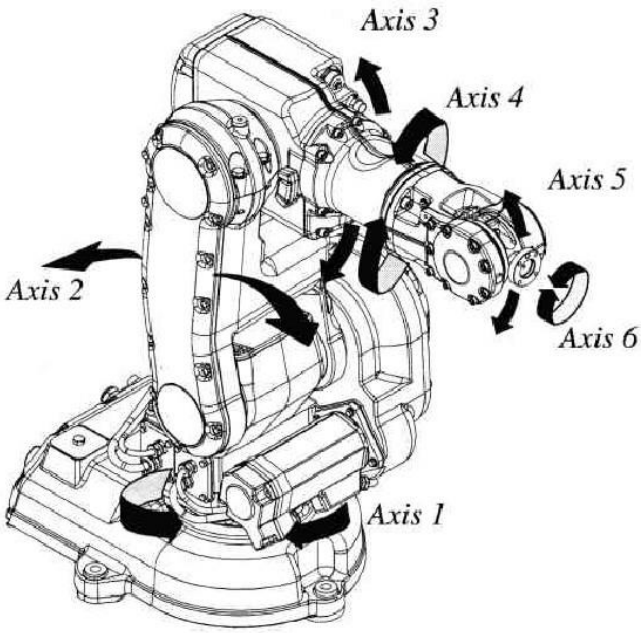


Figure 1.1: The IRB 140 with six degrees of freedom

The IRB 140 is an industrial robot produced by ABB, designed specifically for manufacturing industries. Their website [10] presents various facts about the manipulator, as well as articles, data sheets and movies. The manipulator has a total of six revolute joints that are controlled by AC-motors, hence six degrees of freedom (6 DOF). *Figure 1.1* gives a clear view of the manipulator and its degrees of freedom. The compact and robust design is adapted for flexible use, and the robot can be mounted on the floor, the wall or the roof in any angle. It offers outstanding accuracy and speed, and suits a lot of industrial tasks as for example:

- spray painting,
- packing
- palletizing.

1.3. Objective

The objective of this thesis is to derive the complete dynamic model of the IRB 140 by the product of exponential formula and analyze this method.

For accomplish the task it is necessary to solve following subtask:

- Comparative analysis of methods of robot kinematics and dynamics
- Studying the well-known methods of robot control
- Realization a dynamical model of an industrial robot with using exponential matrices
- Development of a technique for designing of robust robot controller with using the theory of stability of non-linear systems
- Simulation of the robot controller and robot dynamics by Matlab
- Comparison of results

1.4. Software

Mathcad 14

Mathcad [5] is computer software primarily intended for the verification, validation, documentation and re-use of engineering calculations. First introduced in 1986 on DOS, it was the first to introduce live editing of typeset mathematical notation, combined with its automatic computations.

Mathcad, Parametric Technology Corporation's engineering calculation solution, is used by engineers and scientists in various disciplines–most often those of mechanical, chemical, electrical, and civil engineering. Mathcad today includes some of the capabilities of a computer algebra system, but remains oriented towards ease of use and simultaneous documentation of numerical engineering applications.

Mathcad has been used to derive some parameters, which necessary for dynamic model.

Maple 17

Maple [6] is developed by MapleSoft, and is a technical computing software for doing symbolic, numeric and graphical computations. Because of its great efficiency in symbolic computations, Maple has been used to derive the dynamic model for the **IRB** 140.

MatlabR2013bwithSimulink

Matlab [7] is developed by MathWorks, and is a high-level language and numerical computing environment for performing computationally intensive tasks faster than with traditional programming languages. It offers tight integration with other MathWorks products, among them Simulink which is an environment for multidomain simulation and Model-Based Design for dynamic and embedded systems. Matlab and Simulink have been used to simulate the dynamic model for the **IRB** 140, and to present the results graphically.

MicrosoftVisio

Microsoft Office Visio [8] is a diagramming and vector graphics application and is part of the Microsoft Office family. The product was first introduced in 1992, made by the Shapewarecorporation. It was acquired by Microsoft in 2000.

Microsoft Visio has been used to creation drawings describing structure of manipulator.

2. Background theory and notation

This thesis follows the standard convention of how a robot manipulator is interpreted. Fundamental background theory and important notation that are used throughout the thesis are briefly explained in this chapter to facilitate the understanding of the later chapters.

Section 2.1describesthe concept of rotation matrices and kinematics of manipulator. Section 2.2 describes rotational matrices and homogenous transformational matrices, which is the one of part model of robot, also describes their properties and connection with skew symmetric matrices. Section 2.3-2.5 describes mathematical structure of method which based on product based on exponential formula. Section 2.6 describes the main types of joint in robots. Section 2.7 and 2.8 include describing algorithms for creation kinematic model of n-link manipulator. Section 2.9 describes dynamic model structure of manipulator based on Newton-Euler equation and product based on exponential formula.

2.1. Manipulator kinematics

The kinematic of a robot manipulator it is analytic describing of the geometric motion of manipulator relatively to some given absolute coordinate frame without taking force and moments into account, which actuate this motion. Thus the task of kinematics is analytic describing the attitude of manipulator with relation to time and especially determination connection between coordinates of manipulator links and orientation of gripper in orthogonal coordinates.

The manipulator can be consider as open chain, which consist from several rigid links jointed sequentially with the help rotational or translational joints.

Consider two types of tasks

- The **forward kinematics** of a robot determines the configuration of the end-effector (the gripper or tool mounted on the end of the robot) according to given vector of generalized coordinates $q = (q_1, q_2 \dots q_n)^T$
- The **inverse kinematics**of a robot determines the joint angles which achieve desired configuration according to given a desired configuration for the tool frame.

2.2. The rotational matrices

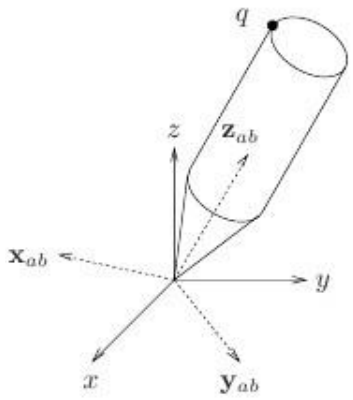


Figure 2.1 Absolute coordinate system and relative coordinate system

In order to perform algebraic manipulations with vectors using coordinates, it is essential that all vectors are expressed in the same coordinate frame. Rotation matrices are used to accomplish this. An n×n rotation matrix specifies the orientation of one frame relative to another frame in the n-dimensional Euclidean space. To specify the coordinate vectors of frame 1 with respect to frame 0 in three dimensions, the 3×3 rotation matrix is written as

$$R_1^0 = [x_{ab} \quad y_{ab} \quad z_{ab}], \tag{2.1}$$

where the columns are the coordinates of the vectors x_{ab}, y_{ab}, z_{ab} expressed in frame XYZ

Below is a matrix of elementary rotations:

$$R_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \propto & -\sin \propto \\ 0 & \sin \propto & \cos \propto \end{bmatrix}, R_{y,\alpha} = \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix}, R_{z,\alpha} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

In a number of cases the mobile coordinate frame can perform the rotation by an angle φ relatively arbitrary axis r, thus in common form rotation matrix is written as

$$R_{r,\varphi} = \begin{bmatrix} r_x^2 \cdot V + c\varphi & r_x \cdot r_y \cdot V - r_z \cdot s\varphi & r_x \cdot r_z \cdot V + r_y \cdot s\varphi \\ r_x \cdot r_y \cdot V + r_z \cdot s\varphi & r_y^2 \cdot V + c\varphi & r_y \cdot r_z \cdot V - r_z \cdot s\varphi \\ r_x \cdot r_z \cdot V - r_y \cdot s\varphi & r_y \cdot r_z \cdot V + r_z \cdot s\varphi & r_z^2 \cdot V + c\varphi \end{bmatrix} \tag{2.3}$$

where $c\varphi = \cos\varphi, s\varphi = \sin\varphi, V = 1 - \cos\varphi$

2.2.1. Properties of the rotation matrices

1. Each column of the rotation matrix is a unit vector in the direction corresponding to the axis of the rotated frame defined by its coordinates relative to the absolute coordinate system.
2. Each row of the rotation matrix is a unit vector in the direction corresponding to the axis of absolute coordinate system defined its coordinates relative to the rotated frame.
3. $R^T = R^{-1}$ and $RR^T = I_3$, where I_3 is a unit matrix with size 3×3
4. detR=1
5. The columns (and therefore the rows) of R are mutually orthogonal

2.2.2. Relation to skew symmetric matrices

An n × n matrix S is said to be skew symmetric if and only if

$$S^T + S = 0$$

which means that every 3×3 skew symmetric matrix has the form

$$S = \begin{bmatrix} 0 & s_3 & s_2 \\ -s_3 & 0 & s_1 \\ -s_2 & -s_1 & 0 \end{bmatrix}$$

Skew symmetric matrices have been found useful in relation to rotationmatrices. Four important properties are given below.

1. For any vectors $a, p \in R^3$

$$S(a)p = a \times p$$

where S is a 3×3 skew symmetric matrix

2. For $R \in SO(3)$ and $a \in R^3$

$$RS(a)R^T = S(Ra)$$

where S is a 3×3 skew symmetric matrix

3. In the general case of angular velocity about an arbitrary and possibly moving axis we have

$$\dot{R}(t) = S(\omega(t))R(t)$$

where $R = R(t) \in SO(3)$for every $t \in R$, S is a 3×3 skew symmetricmatrix,and $\omega(t)$ is the angular velocity of the rotating frame with respect to the fixed frame at time t.

4. For an $n \times n$skew symmetric matrix S and any vector $X \in R^n$

$$X^T SX = 0$$

2.2.3. Homogeneous coordinates and transformation matrix

As 3×3 rotation matrix carries information only about rotation around some axis and does not take translation and scale into account then vector $p = (p_x, p_y, p_z)^T$vector complement fourth coordinateso that the vector take a new form $\hat{p} = (\omega p_x, \omega p_y, \omega p_z, \omega)^T$. Then vector $\hat{p}$ expressed in homogeneous coordinates. Physical coordinates associated with the homogeneous, as follows

$$p_x = \frac{\omega p_x}{\omega}, p_y = \frac{\omega p_y}{\omega}, p_z = \frac{\omega p_z}{\omega}, \quad (2.4)$$

where ω is a forth component of vector of homogeneous coordinates (scale multiplier).

If $\omega = 1$ then homogeneous coordinates of position vector coincide with its physical coordinates.

The homogenous transformation matrix have a size 4×4 and convert vector from one coordinate system to other. The homogeneous matrix in common form is written as:

$$T = \begin{bmatrix} R_{3\times3} & p_{3\times1} \\ f_{1\times3} & 1 \times 1 \end{bmatrix} = \begin{bmatrix} Rotation & Translation \\ Converting\ prospect & Scale \end{bmatrix}$$

$$T = \begin{bmatrix} x_x & y_x & z_x & p_x \\ x_y & y_y & z_y & p_y \\ x_z & y_z & z_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

2.3. Exponential coordinates for rotation

An alternative to the rotation matrix is matrix based on exponential coordinates for rotation. Consider rotation of robot link around fixed axis Figure 2.2.

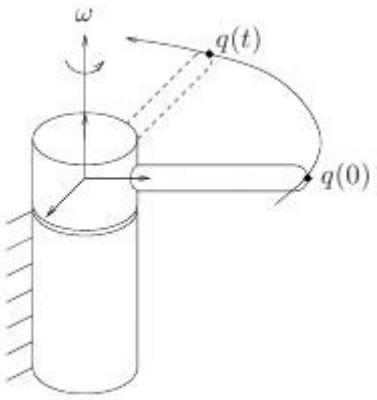


Figure 2.2 Tip point trajectory generated by rotation about the *ω*-axis

Let $\omega \in R^3$ be a unit vector, which specifies the direction of rotation and let$\theta \in R^3$bethe angle of rotation in radians. Then velocity of point q can be written as

$$\dot{q}(t) = \omega \times q(t) = \hat{\omega} q(t). \tag{2.6}$$

This is a time-invariant linear differential equation which may be integrated to give

$$q(t) = e^{\hat{\omega} t} q(0),$$

where q(0) is the initial position of the point and $e^{\hat{\omega} t}$ is the matrix exponential

$$e^{\hat{\omega} t} = I + \hat{\omega} t + \frac{(\hat{\omega} t)^2}{2!} + \frac{(\hat{\omega} t)^3}{3!} + \cdots,$$

$$R = e^{\hat{\omega} t}. \tag{2.7}$$

According to [3] get the finite equation for rotation matrix in common form

$$e^{\hat{\omega}\theta} = I + \hat{\omega} sin\theta + \hat{\omega}^2(1 - cos\theta) = \begin{bmatrix} 1 - v_\theta(\omega_2^2 + \omega_3^2) & \omega_1\omega_2 v_\theta - \omega_3 s_\theta & \omega_1\omega_3 v_\theta + \omega_2 s_\theta \\ \omega_1\omega_2 v_\theta + \omega_3 s_\theta & 1 - v_\theta(\omega_1^2 + \omega_3^2) & \omega_2\omega_3 v_\theta - \omega_1 s_\theta \\ \omega_1\omega_3 v_\theta - \omega_2 s_\theta & \omega_2\omega_3 v_\theta + \omega_1 s_\theta & 1 - v_\theta(\omega_1^2 + \omega_2^2) \end{bmatrix} =$$

$$\boxed{\begin{bmatrix} \omega_1^2 v_\theta + c_\theta & \omega_1\omega_2 v_\theta - \omega_3 s_\theta & \omega_1\omega_3 v_\theta + \omega_2 s_\theta \\ \omega_1\omega_2 v_\theta + \omega_3 s_\theta & \omega_2^2 v_\theta + c_\theta & \omega_2\omega_3 v_\theta - \omega_1 s_\theta \\ \omega_1\omega_3 v_\theta - \omega_2 s_\theta & \omega_2\omega_3 v_\theta + \omega_1 s_\theta & \omega_3^2 v_\theta + c_\theta \end{bmatrix}}, \tag{2.8}$$

where $v_\theta = 1 - cos\theta, s_\theta = sin\theta, c_\theta = cos\theta$

2.4. Exponential coordinates for rigid motion and twists

An alternative to the homogeneous matrix is exponential mapping which allows represent geometric treatment of spatial rigid body motion in elegant and rigorous form. Consider the easy example of robot with one link as shown in Figure 2.3.

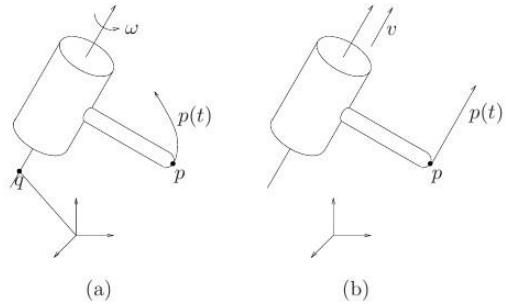


Figure 2.3 (a) Rotation joint and (b) translation joint

a) For rotation link

Velocity of the tip point

$$\dot{p}(t) = \omega \times (p(t) - q). \tag{2.9}$$

Equation can be rewritten with an extra row append to it as

$$\begin{bmatrix} \dot{p} \\ 0 \end{bmatrix} = \begin{bmatrix} \widehat{\omega} & -\omega \times q \\ 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ 1 \end{bmatrix} = \hat{\xi} \begin{bmatrix} p \\ 1 \end{bmatrix} \Longrightarrow \dot{p} = \hat{\xi}\overline{p}$$

where $v = -\omega \times q$

To solution of the differential equation is given by

$$\overline{p}(t) = e^{\hat{\xi}t}\overline{p}(0)$$

where $e^{\hat{\xi}t}$is the 4×4 matrix exponential of the, defined as

$$e^{\hat{\xi}t} = I + \hat{\xi}t + \frac{(\hat{\xi}t)^2}{2!} + \frac{(\hat{\xi}t)^3}{3!} + \cdots$$

The scalar *t* is the total amount of rotation. exp($\hat{\xi}t$) is a mapping from the initial location of a point to its location after rotating *t* radians.

b) In a similar manner can represent the transformation due to translation motion as the exponential of a 4×4 matrix.

The velocity of a point

$$\dot{p}(t) = v. \tag{2.10}$$

In the common form transformation matrix written as

$$e^{\hat{\xi}\theta} = \begin{bmatrix} e^{\widehat{\omega}t} & h\omega\theta \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} e^{\widehat{\omega}t} & \left(I - e^{\widehat{\omega}t}\right)(\omega \times v) + \omega\omega^T v\theta \\ 0 & 1 \end{bmatrix} \omega \neq 0. \tag{2.11}$$

The transformation $g = \exp(\hat{\xi}\theta)$ is slightly different than the rigid transformation. Itsinterpret not as mapping points from one coordinate frame to another, but rather as mapping points from their initial coordinates, $p(0) \in R^3$, to their coordinates after the rigid motion is applied

$$\bar{p}(\theta) = e^{\hat{\xi}\theta}\bar{p}(0)$$

In this equation, both p(0) and p(θ) are specified with respect to a single reference frame. Similarly if $g_{ab}(0)$ represent the initial configuration of a rigid body relative to a frame A, then final configuration still with respect to A, is given by

$$g_{ab}(\theta) = e^{\hat{\xi}\theta}g_{ab}(0). \tag{2.12}$$

2.5. Screws: a geometric description of twists

Consider a rigid body motion which consists of rotation about an axis in space through an angle of θ radians, followed by translation along the same axis by an amount d as shown in Figure 2.4.

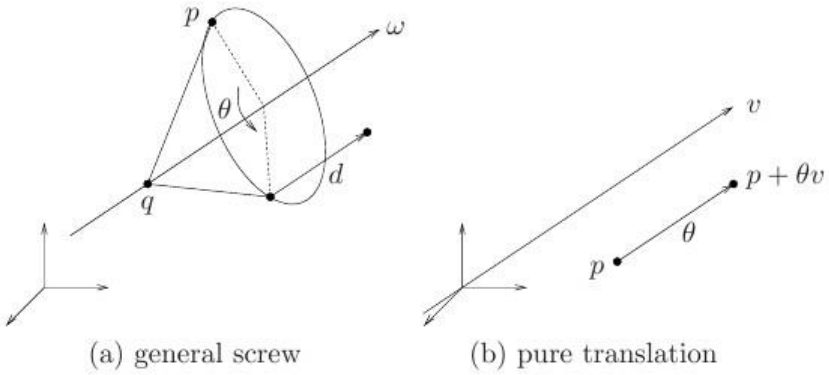


Figure 2.4 Screws motion

This motion called a *screw motion*, since it is reminiscent of the motion of a screw, in so far as a screw rotates and translates about the same axis. Take this analogy into account, we define the *pitch* of the screw to be the ratio of translation to rotation $h = \frac{d}{\theta}$. Represent axis as a directed line through a point; choosing $q \in R^3$ to be a point onhe axis and $\omega \in R^3$ to be a unit vector specifying the direction, the axis is the set of points. If the case of zero rotation, the axis of the screw must be taken as the line through the origin in the direction v , v is a vector of magnitude 1. Below is given geometric description of rotation, as particular case of screw motion.

2.5.1. Geometric description of twist

In order to compute the rigid body transformation associated with a screw, we analyze the motion of a point $p \in R^3$, as shown in Figure 2.5

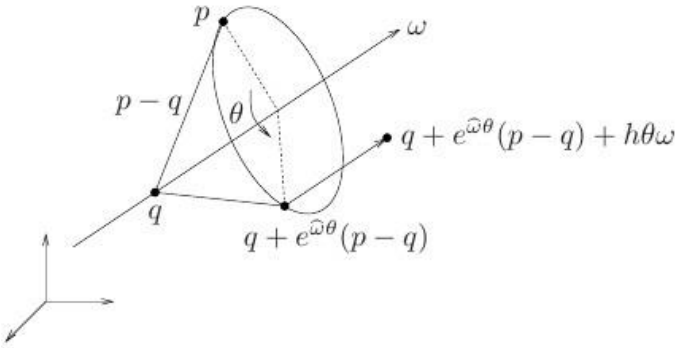


Figure 2.5 Generalized screw motion(with nonzero rotation)

The final location of the point is given

$$gp = q + e^{\hat{\omega}\theta}(p - q) + h\theta\omega$$

or, in homogeneous coordinates,

$$g\begin{bmatrix}p\\1\end{bmatrix}=\begin{bmatrix}e^{\hat{\omega}\theta} & e^{\hat{\omega}t}\left(I-e^{\hat{\omega}t}\right)q+h\theta\omega\\ 0 & 1\end{bmatrix}\begin{bmatrix}p\\1\end{bmatrix}.$$

where

$$\boxed{e^{\hat{\xi}\theta}=\begin{bmatrix}e^{\hat{\omega}\theta} & e^{\hat{\omega}t}\left(I-e^{\hat{\omega}t}\right)q+h\theta\omega\\ 0 & 1\end{bmatrix}} \tag{2.13}$$

Note that equation (2.13) describing displacement of the rigid body have the same form as equation (2.11). If we use the substitute $v=-\omega\times q+h\omega$ in equation (2.11) then we get the same equation for screw motion.

Equation (2.13) is the common form of screw motion. In our case we are interested in the particular case when pitch $h=0$ pure rotation. This case used for computation kinematic map for rotation joint of manipulator.

Geometric explanation fully disclosed in the Chasles theorem: "*Every rigid body motion can be realized by a rotation about an axis combined with a translation parallel to that axis"*.Exponential twists describe relative motion of rigid body. The equation

$$p(\theta)=e^{\hat{\xi}\theta}p(0)$$

describe the finite location of point $p(\theta)$ respect to its initial location $p(0)$, in case on *Figure 2.5*

If a coordinate frame B is attached to a rigid body undergoing a screwmotion, the instantaneous configuration of the coordinate frame B, relative to a fixed frame A, is given by

$$g_{ab}(\theta)=e^{\hat{\xi}\theta}g_{ab}(0) \tag{2.14}$$

This transformation can be interpreted as follows: multiplication by$g_{ab}(\theta)$maps the coordinates of a point relative to the B frame into A'scoordinates, and the exponential map transforms the point to its finallocation (still in A coordinates).

2.6. Kinematic chains

Robot manipulators are composed of links connected by joints to form a kinematic chain, where the joints are revolute or prismatic. A revolute joint is like a hinge and allows relative rotation between two links, while a prismatic joint allows a linear relative motion between two links. Both types of joints have a single degree of freedom, thus each joint*i* can be represented by a single joint variable q_i. Figure 2.6 shows a symbolic representation of robot joints in 2D and 3D.

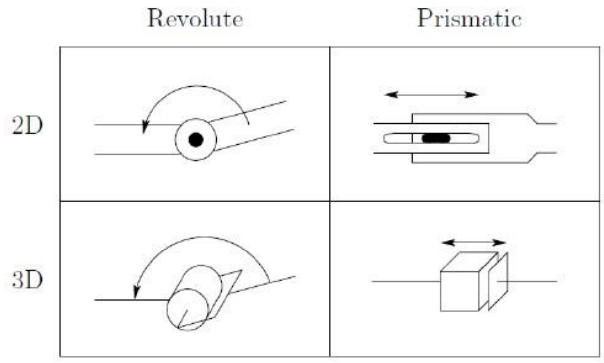


Figure 2.6 Symbolic representation of robot joints

A configuration of a manipulator is a complete specification of every point on the manipulator. Assuming a manipulator with rigid links and a fixed base,that means the configuration is entirely given by q, the vector of joint variables. In case of joints with more degrees of freedom, like a ball or a spherical wrist, these joints can always be thought of as a succession of joints with a single degree of freedom.

A coordinate frame is rigidly attached to each link, and an inertial frame is attached to the robots base. Links, joints and frames are defined as summarized below.

- Links are numbered from 0 to n where link 0 is the base.
- Joints are numbered from 1 ton where joint *i* connects link $i-1$ to link
- When joint i is actuated, link i moves. The base cannot be actuated.
- Frames are numbered from 0 ton where frame i is attached to link i.
- Frames are attached such that axis z_i of frame i is the axis of actuationfor joint $j+1$.
- The joint variable q_i is associated with joint i.

2.7. Denavit-Hartenberg algorithm

For describing rotation joints and translation joints between adjacent links Denavit and Hartenberg offer in 1955 algorithm based on the matrix method for determine coordinate systems. The idea of DH algorithm is in creature a homogeneous transformation matrix which have a size 4×4. This makes it possible to consistently convert the coordinates of the gripper from reference systems associated with the last link tothe basic reference frame which is an inertial coordinate system for the dynamical system.

Each of the coordinate system forms based on the follow rules:

1) z_i-axis is direct along axis of*i-th* joint
2) x_i-axis is perpendicularto thez_{i-1}-axis and direct against it
3) y_i-axis is supplement the x_i, z_i axes to right-hand coordinate system

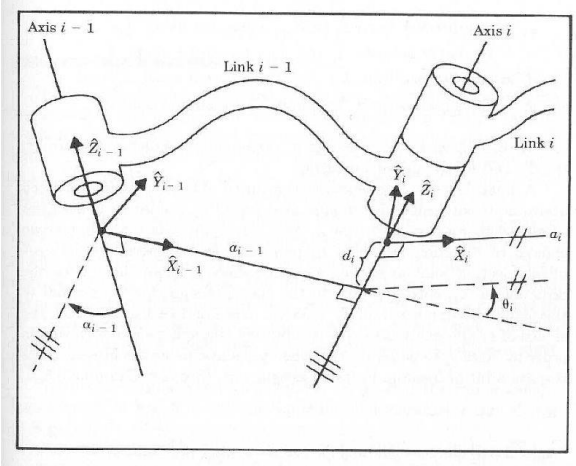


Figure 2.7 Denavit-Hartenberg coordinate system

DH-parameters of rigid links depends from fourth geometric parameters which associated with each link. These four parameters fully described any rotation or translation motion.

- d: offset along previous z to the common normal
- θ: angle about previous z, from old x to new x
- r: length of the common normal(aka a, but if using this notation, do not confuse with α). Assuming a revolute joint, this is the radius about previous z
- α: angle about common normal, from old z axis to new z axis

2.7.1. Forward kinematic equation

The homogeneous matrix T_0^i which determine location of the i-th coordinate system relative to base coordinate system is a multiplication of series of the homogeneous transformation matrices A_{i-1}^i, have the form

$$T_0^i = A_0^1 A_1^2 \dots A_{i-1}^i = \prod_{i=1}^{i} A_{i-1}^i = \begin{bmatrix} x_i & y_i & z_i & p_i \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_0^i & p_0^i \\ 0 & 1 \end{bmatrix}, \quad (2.15)$$

i=1,2,…, n

where$[x_i \quad y_i \quad z_i]$ is a matrix which determine orientation of i-th coordinate system (coupled with i-th link) relative to the base coordinate system. This is the top left sub matrix, have the size 3×3. $p_i$is a vector which connected the beginning of the base coordinate system with beginning i-th coordinate system. It is the top right sub matrix, have the size 3×1. Particularly if i = 6 we will get matrix $T = A_0^6$ which determine location and orientation of the gripper relative the base coordinate system.

2.8. Algorithm for N-link manipulator, based on product of exponential formula.

In the common form the procedure for solving the forward kinematic task for manipulator with open-chain structure and n-DOF looks as follows. Let S is a coordinate system of base of manipulator, T is a coordinate system of a last link.

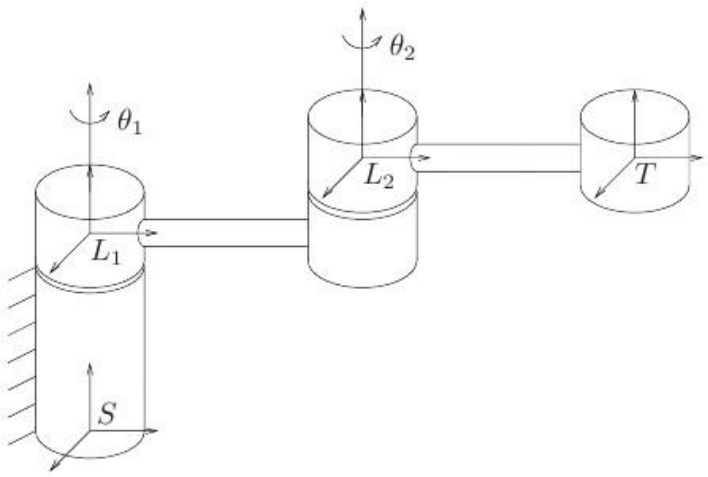


Figure 2.8 Manipulator with 2 DOF

- It is necessary to determine the basic configuration of the manipulator, corresponding to $\theta = 0$, where $g_{st}(0)$ describes a transformation matrix between the T and S, when the manipulator is in the basic configuration.
- For each joint, it is necessary to record the twists ξ_i which corresponds to a screw motion for each i-th joint, given that other angles for the joints $\theta_j = 0$.

  $\xi_i = \begin{bmatrix} -\omega_i \times q_i \\ \omega_i \end{bmatrix}$ – revolute joint

  $\xi_i = \begin{bmatrix} v_i \\ 0 \end{bmatrix}$ – prismatic joint
- Combining the individual joint motions, we can get the solution for forward kinematic task.

$$\boxed{g_{st}(\theta) = e^{\widehat{\xi_1}\theta_1} e^{\widehat{\xi_2}\theta_2} \dots e^{\widehat{\xi_n}\theta_n} g_{st}(0),} \quad (2.18)$$

The ξ_i must be numbered sequentially starting from the base, but $g_{st}(\theta)$gives the configuration of the tool frame independently of the orderin which the rotations and translations are actually performed. Equation (2.18) is called the product of exponentials formula for the manipulator forward kinematics.

2.9. Dynamics of manipulator

Robot manipulators can be described mathematically in different ways. The problem of kinematics is to describe the motion of the manipulator without consideration of forces and torques causing the motion. These equations determine the position and orientation of the end effector given the values for the joint variables (forward kinematics), and as the opposite the values of the joint variables given the position and orientation of the end effector (inverse kinematics).

Dynamics section as part of robotics is a mathematical description of the correlation of forces and moments acting on the arm, in the form of the equations of dynamics. Also equations needed to simulate the movement of the manipulator using a computer, in choosing of control laws, as well as in the evaluation of the quality and design of the kinematic scheme and construction of robot. For compiling dynamic equation which is a mathematical model usually

used the known laws of Newtonian and Lagrangian mechanics. Also exist an alternative method of calculating the elements of the equation, constituting a model based on the product of exponential formula. The result of the application of these laws is the equation that is the same for all representation methods:

$$\boxed{M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau,} \tag{2.19}$$

- q - generalized coordinates (n×1))
- τ – vector of actuator toques (n×1);
- M– inertia matrix;
- C– Coriolis matrix;
- g– gravity vector;

2.9.1. Newton – Euler versus product of exponential formula

The efficiency of the Newton-Euler formulation and product of exponential formula is an interesting topic. Actually there is no clear answer to the question of which method is better than the other. The main goal is to derive the dynamic model as fast as possible, and how well this goal is satisfied for each method depends on several factors. The number of link and joints in the kinematic chain, the topology of the chain (e.g. serial or parallel), the position and orientation of the coordinate frames, and whether a recursive procedure is used or not, are factors that will influence the computation time.

The Newton-Euler formulation is usually the preferred choice for manipulators with many degrees of freedom. The reason is the recursive structure which the Newton-Euler formulation is based on. If the frames are attached in a convenient way, the recursions will be greatly simplified. The recursive approach is in general faster than treating the manipulator as a whole system. It should also be mentioned that for the case of parallel manipulators, the Newton-Euler formulation gives an advantage for dynamic computations and control.

Also exist an alternative methods of realization, one of them the method based on product of exponential formula which consider manipulator as a whole system. In the [3] consider the method of calculation a dynamic model which structure is similar to the Euler – Lagrange formulation and allegedly the author : “If the forward kinematics are specified using the product of exponential formula, then it is possible ti get more explicit formulas for the inertia and Coriolis matrices.”

The selection of algorithm is a matter of personal preference and the key factor for selection this or that algorithm is that each algorithm can provide a different representation of the same mechanism.

2.9.2. Equation of Newton-Euler formula

The basis of the Newton-Euler formulation is three important mechanic laws:

- Every action has an equal and opposite reaction. Thus, if link 1 applies a force f and torque τ to link 2, then link 2 applies a force — f and torque—τ to link 1.
- The rate of change of the linear momentum equals the total force applied to the link.
- The rate of change of the angular momentum equals the total torque applied to the link.

Applying the second law to the linear motion of a link gives the relationship

$$\frac{d(mv)}{dt} = f, \tag{2.20}$$

where *m* is the mass of the link, *v* is the velocity of the center of mass with respect to an inertia! frame, and f is the sum of external forces applied to the link. Since the mass is constant as a function of time for robot manipulators, Equation (2.20) can be simplified to

$$f = ma, \tag{2.21}$$

where *a* is the acceleration of the center of mass. The third law gives the relationship

$$\frac{d(I_0\omega_0)}{dt} = \tau_0, \tag{2.22}$$

where $I_0$is the moment of inertia of the link, $\omega_0$is the angular velocity of the link, and τ_0 is the sum of torques applied on the link. All three variables are expressed in an inertial frame whose origin is at the center of mass. Note that I_0 is not necessarily a constant function of time, but this can be taken care of by rewriting Equation (2.22) to be valid for a frame rigidly attached to the the link instead of an inertial frame. A similarity transformation of I0 is given by

$$I = R^{-1}I_0R \tag{2.23}$$

which gives

$$I_0 = RIR^T \tag{2.24}$$

where *R* is the rotation matrix that transforms coordinates from the link attached frame to the inertial frame. Equation (2.22) together with the Equation (2.24) and facts

$$\omega_0 = R\omega, \qquad \tau_0 = R\tau. \tag{2.25}$$

yields

$$\frac{d(I_0\omega_0)}{dt} = \frac{d(RIR^TR\omega)}{dt} = \frac{d(RI\omega)}{dt} = \dot{R}I\omega + RI\dot{\omega}, \tag{2.26}$$

and the equation for the rate of change of the angular momentum with respect to the link attached frame is

$$\tau = R^T\tau_0 = R^T(\dot{R}I\omega + RI\dot{\omega}) = R^T\dot{R}I\omega + I\dot{\omega}, \tag{2.27}$$

The rotation matrix in Equation (2.27) can be cancelled out by taking advantage of the properties showed in subsection 2.2.2. The final torque expression becomes

$$\tau = R^T\dot{R}I\omega + I\dot{\omega} = R^TS(\omega_0)RI\omega + I\dot{\omega} = S(R^T\omega_0)I\omega + I\dot{\omega} = S(\omega)I\omega + I\dot{\omega} = \omega \times I\omega + I\dot{\omega}. \tag{2.28}$$

This concludes the general case of the derivation with the force balance and moment balance summarized respectively as

$$\boxed{f = ma,} \tag{2.29}$$

$$\boxed{\tau = \omega \times I\omega + I\dot{\omega},} \tag{2.30}$$

2.9.3. Equations of an n-link manipulator

To begin with, several vectors need to be introduced. Note that all these vectors are expressed in frame *i*.

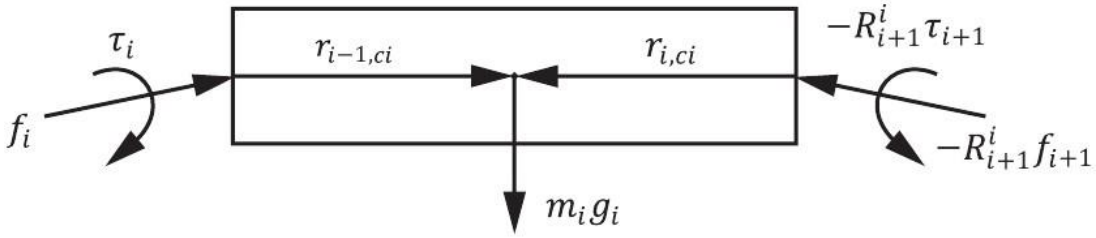


Figure 2.9 Forces and torques acting on a random link

$a_{c,i}$ - acceleration of the center of mass of link *i*

$a_{e,i}$ - acceleration of the end of link *i*(origin of frame *i*+ 1)

ω_i - angular velocity of frame *i*with respect to frame 0

α_i - angular acceleration of frame *i*with respect to frame 0

z_i - axis of actuation of frame *i*with respect to frame 0

g_i - acceleration due to gravity

f_i - force exerted by link *i* — 1 on link *i*

τ_i - torque exerted by link *i* — 1 on link *i*

R^i_{i-1} - rotation matrix from frame *i*to frame *i*+ 1

m_i - the mass of link *i*

I_i - inertia tensor of link *i*about a frame parallel to frame *I*whose origin is at the center of mass of link *i*

$r_{i-1,ci}$ - vector from the origin of frame *i* — 1 to the center of mass of link *i*

$r_{i-1,i}$ - vector from the origin of frame *i* — 1 to the origin of frame *i*

$r_{i,ci}$ - vector from the origin of frame *i*to the center of mass of link

When all vectors in Figure 2.9 are expressed in frame *i*, the force balance equation based on (2.29) can be stated as

$$\sum_{link} f = ma \tag{2.31}$$

$$f_i - R_{i+1}^i f_{i+1} + m_i g_i = m_i a_{c,i}, \tag{2.32}$$

$$f_i = R_{i+1}^i f_{i+1} + m_i a_{c,i} - m_i g_i \tag{2.33}$$

Next, the moment balance equation for the link will be computed, and it is important to note two things:

1) the moment exerted by a force f about a point is given by $f \times r$, where r is the radial vector from the point where the force is applied to the point where the moment is computed.
2) the vector $m_i g_i$does not appear in the moment balance since it is applied directly at the center of mass. The moment balance equation based on (2.30) becomes

$$\sum_{link} \tau = \omega \times (I\omega) + I\dot{\omega} \tag{2.34}$$

$$\tau_i - R_{i+1}^i \tau_{i+1} + f_i \times r_{i-1,ci} - \left(R_{i+1}^i f_{i+1}\right) \times r_{i,ci} = \omega_i \times (I_i \omega_i) + I_i \alpha_i \tag{2.35}$$

$$\tau_i = R_{i+1}^i \tau_{i+1} - f_i \times r_{i-1,ci} + \left(R_{i+1}^i f_{i+1}\right) \times r_{i,ci} + \omega_i \times (I_i \omega_i) + I_i \alpha_i. \tag{2.36}$$

The force balance equation is actually a part of the moment balance equation. Solving Equation (2.36) for decreasing *i* and substituting (2.33) is the ultimate goal of the formulation, but the solution needs to be expressed only by $q, \dot{q}, \ddot{q}$and constant parameters to achieve the general matrix form (2.19). That means it is necessary to find a relation between $q, \dot{q}, \ddot{q}$and $a_{c,i}, \omega_i$and α_i. This can be obtained by a recursive procedure of increasing *i*.

Since the force and moment equations are expressed with respect to the link attached frame, this also applies to$a_{c,i}, \omega_i$and $\alpha_{i,c}$. However, as a starting point $\omega_i$and $\alpha_i$need to be expressed in the inertial frame, and the superscript (0) will be used to denote that. This gives

$$\omega_i^{(0)} = \omega_{i-1}^{(0)} + z_{i-1}\dot{q}_i, \tag{2.37}$$

because of the fact that the angular velocity of frame *i* equals that of frame *i* -1 plus the added rotation from joint *i*. Using rotation matrices this leads to

$$\omega_i = (R_i^{i-1})^T \omega_{i-1} + b_i \dot{q}_i \tag{2.38}$$

where

$$b_i = (R_i^0)^T R_{i-1}^0 z_0 \tag{2.39}$$

is the rotation of joint *i* expressed in frame *i*.

For the angular acceleration it is important to note that

$$\alpha_i = (R_i^0)^T \dot{\omega}_i^{(0)} \tag{2.40}$$

which means $\alpha_i \neq \dot{\omega}_\imath$! By using Newtons Second Law in a rotating frame, the time derivative of Equation (2.36) becomes

$$\dot{\omega}_i^{(0)} = \omega_{i-1}^{(0)} + z_{i-1}\dot{q}_i + \omega_i^{(0)} \times z_{i-1}\dot{q}_i, \tag{2.41}$$

and expressed in frame *i* it directly becomes

$$\alpha_i = (R_i^{i-1})^T \alpha_{i-1} + b_i q_\imath + \ddot{\omega_\imath} \times b_\imath \dot{q}_\imath \tag{2.42}$$

Now it only remains to find an expression for $a_{c,i}$. First, the linear velocity of the center of mass of link *i* is expressed as

$$v_{c,i}^{(0)} = v_{e,i-1}^{(0)} + \omega_i^{(0)} \times r_{i-1,ci}^{(0)} \tag{2.43}$$

and note that $r_{i-1,ci}^{(0)}$is constant in frame *i*. Thus

$$a_{c,i}^{(0)} = a_{e,i-1}^{(0)} \times r_{i-1,ci}^{(0)} + \omega_i^{(0)} \times \left(\omega_i^{(0)} \times r_{i-1,ci}^{(0)}\right) \tag{2.44}$$

Multiplying with rotation matrices and using the fact that

$$R(a \times b) = (Ra) \times (Rb) \tag{2.45}$$

the final expression for the acceleration of the center of mass of link *i*, expressed in frame *i*, becomes

$$a_{c,i} = (R_i^{i-1})^T a_{e,i-1} + \dot{\omega}_i \times r_{i-1,ci} + \omega_i \times \left(\omega_i \times r_{i-1,ci}\right) \tag{2.46}$$

To find the acceleration of the end of the link, $r_{i-1,ci}$ is replaced by $r_{i-1,i}$

$$a_{e,i} = (R_i^{i-1})^T a_{e,i-1} + \dot{\omega}_i \times r_{i-1,i} + \omega_i \times \left(\omega_i \times r_{i-1,i}\right) \tag{2.47}$$

This completes the recursive formulation, and the Newton-Euler formulation of an n-link manipulator can be stated as follows.

1. Forward recursion: Start with the initial conditions

$$\omega_0 = \alpha_0 = a_{c,0} = a_{e,0} = 0 \tag{2.48}$$

and solve Equations (2.38), (2.39), (2.40), (2.42). (2.46) and (2.47) (in that order) to compute ω_i, α_i,$a_{c,i}$, $a_{e,i}$ for increasing *i* from 1 to *n.*

2. Backward recursion: Start with the terminal conditions

$$f_{n+1} = \tau_{n+1} = 0$$

and solve Equations (2.34) and (2.36) (in that order) for decreasing *i* from *n* to 1.

2.9.4. Robot dynamic model based on product of exponential matrix

In the case when forward kinematics are specified using the product of exponential formula, then it is possible to get more explicit formulas for the inertia and Coriolis matrices."

In order to obtain the inertial matrix, it is necessary to determine the following parameters:

- Link inertia matrixM_i

$$M_i = \begin{bmatrix} m_i \mathrm{I} & 0 \\ 0 & I_i \end{bmatrix} \tag{2.50}$$

- Adjoint transformation$A_{ij} \in R^{6\times 6}$

- $$A_{ij} = \begin{cases} Ad^{-1}_{\left(e^{\xi_{j+1}\theta_{j+1}} \ldots e^{\xi_i \theta_i}\right)} & i > j \\ I & i = j \\ 0 & i < j \end{cases} \tag{2.51}$$

Adjoint transformation followed from the equation which describe the space velocity of rigid body [3].

In the general case $g_{ab}(t) \in SE(3)$ is a matrix describing the trajectory of rigid body with coordinate frame B relative to coordinate A.

$$g_{ab}(t) = \begin{bmatrix} R_{ab}(t) & p_{ab}(t) \\ 0 & 1 \end{bmatrix} \tag{2.52}$$

In the [3] cites the equations which describes space velocity of rigid body

$$\omega^s_{ab} = R_{ab}\omega^b_{ab}$$

$$v^s_{ab} = p_{ab} \times \left(R_{ab}\omega^b_{ab}\right) + R_{ab}v^b_{ab}$$

where v^s_{ab}-space velocity of point , ω^s_{ab}-angle velocity in space, v^b_{ab}- velocity of the coordinate system origin relative to the space coordinate system, in respect to current position of coordinate system of body. ω^b_{ab}- angle velocity coordinate system also in respect to current position. Rewrite in the matrix form:

$$V^s_{ab} = \begin{bmatrix} v^s_{ab} \\ \omega^s_{ab} \end{bmatrix} = \begin{bmatrix} R_{ab} & \hat{p}_{ab}R_{ab} \\ 0 & R_{ab} \end{bmatrix} \begin{bmatrix} v^b_{ab} \\ \omega^b_{ab} \end{bmatrix} \tag{2.53}$$

The 6×6, matrix which transforms twists from one coordinate frame to another is referred to as the *adjoint transformation* associated with g, written as Ad_g. Thus, given $g \in SE(3)$which maps one coordinate system to another,

$$Ad_g = \begin{bmatrix} R_{ab} & \hat{p}_{ab}R_{ab} \\ 0 & R_{ab} \end{bmatrix} \tag{2.54}$$

This matrix is invertible:

$$Ad_g{}^{-1} = \begin{bmatrix} R^T & -(R^T p)\hat{}R^T \\ 0 & R^T \end{bmatrix} = \begin{bmatrix} R^T & -R^T\hat{p} \\ 0 & R^T \end{bmatrix} = Ad_{g^{-1}} \tag{2.55}$$

Equation (2.55) allows determine the elements of adjoint transformation matrix (2.51). Used equation (2.51) jth column of the body Jacobian for the ith link is given by $Ad_{g_{sl_i}^{-1}} A_{ij}\xi_j$:

$$J_i(\theta) = Ad_{g^{-1}_{sl_i^{(0)}}} \begin{bmatrix} A_{i1}\xi_1 \dots A_{ij}\xi_i & 0 \dots 0 \end{bmatrix} \tag{2.56}$$

Combine $Ad_{g^{-1}_{sl_i^{(0)}}}$ with the link inertia matrix by defining the transformed inertia matrix for the link [3]

$$M_i' = Ad^T_{g^{-1}_{sl_i^{(0)}}} M_i Ad_{g^{-1}_{sl_i^{(0)}}} \tag{2.57}$$

Using the equations (2.51), (2.55), (2.57) it can be get equations for determining inertia matrix and Coriolis matrix which necessary for composition dynamic equation

$$\boxed{M_{ij}(\theta) = \sum_{l=\max(i,j)}^{n} \xi_i^T A_{li}^T M_l' A_{lj} \xi_j} \tag{2.58}$$

$$\boxed{C_{ij}(\theta) = \frac{1}{2}\sum_{k=1}^{n} \left(\frac{\partial M_{ij}}{\partial \theta_k} + \frac{\partial M_{ik}}{\partial \theta_j} - \frac{\partial M_{kj}}{\partial \theta_i}\right) \dot{\theta}_k} \tag{2.59}$$

As shown in equations (2.58), (2.59) all of the dynamic attributes of the manipulator can be determined directly from the joint twists ξ_i, the linkframes $g_{sl_i^{(0)}}$, and the link inertia matrices M_i. The matrices A_{ij} are the only expressions which depend on current configuration of the manipulator.

2.10. Feedback Controllers

A system can be controlled in open loop or closed loop. With an open-loop controller, the input is computed without observing the output that it is controlling. Complex systems will not be possible to control in open loop, because the controller will never know if the output has achieved the desired goal. However, by adding feedback controllers, it might be possible to stabilize the system in closed loop.

A feedback controller observes the output and calculates the error between this output and a reference value. Then the input is computed based on this error such that the output approaches the reference value. To achieve a desired behavior of the output, controllers can take one or more of three standard control elements. These elements are

- **P - proportional term**: The input is proportional to the error between the reference value and the current output. Kp is the proportional gain.
- **I - integral term**: Integrates the error over time and multiplies with the integral gain Ki. The term eliminates steady state error.
- **D - derivative term**: Determines the slope of the error over time and multiplies with the derivative gain Kd. The term has as a damping effect.

3. System Description and dynamic parameter estimation

ABB has produced the industrial robot manipulator named IRB 140. Their website [10] presents facts about the manipulator, as well as data sheet, articles and movies about abilities of manipulator.

This chapter is presenting all information about the IRB 140 which is needed to derive the dynamic model. The manipulator comes with a product manual, a product specification [10], and a data sheet (Attachment A1). The manual is not of much interest in this thesis, as it focuses solely on safety, installation and maintenance. What is interesting is the data sheet, which is basically a summary of the product specification, presenting some facts about the structure and performance of the manipulator. The relevant information given in the data sheets are summarized in Section 3.1.

Out of consideration for trade secrets in ABB, the data sheets present a very limited amount of information. Section 3.2 states these limitations and how they lead to simplified dynamic parameter estimation.

In Section 3.3, a symbolic representation shows how the joints and links can be represented as a serial kinematic chain, and how frames are attached to the links. This representation follows all guidelines described in the previous chapters, and can be said to lay the foundation for the whole dynamic model.

3.1. Information from data sheets

The manipulator has a total of six revolute joints that are controlled by AC-motors, hence six degrees of freedom (6 DOF). Thetotal mass including the base and without a payload is 98 kg, and the mass of the payload alone must not exceed 6 kg. Someapplicable link dimensions are given in Figure 3.1 (lengths in millimeters).

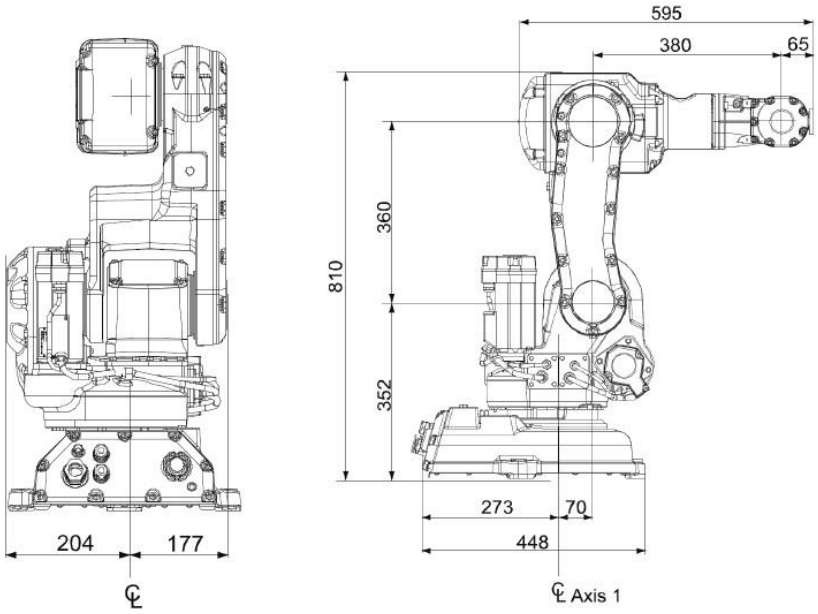


Figure 3.1 View of the manipulator from the back and side

3.2. Limitations

It is not possible to derive an accurate dynamic model for the IRB 140 with the limited information available in the data sheets. The dynamic parameters for the links are not given and explained in Section 3.1, these parameters are indeed a demanding task to estimate. The masses of the links could have been identified by dismantling the manipulator and weigh them one by one, but this would have been a comprehensive task by itself. Besides, this useless, if through experiments on estimating the inertia parameters and centers of mass would not be performed.

Researching dynamic parameter of the IRB 140 is an interesting and challenging task.It can be use identification methods like for example CAD modeling because on the website is a CAD-model of ABB IRB 140. But ABB does not give the characteristic about material of manipulator which is necessary for estimation with the help of CAD-system. Consequently, the dynamic parameters in the model have been estimated quite roughly. The estimation is based on intuitive guesses, with the purpose of creating a simple model which still represents the IRB 140 as good as possible

3.3. Kinematic model

3.3.1. Algorithm of Denavit-Hartenberg

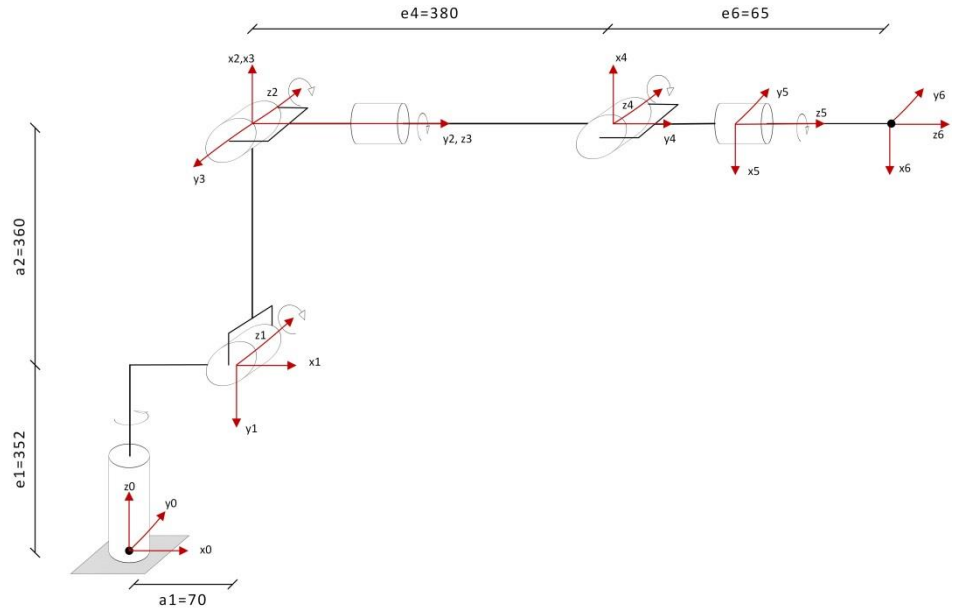


Figure 3.2 Schematic representation of manipulator TRB 140

The IRB 140 can be interpreted in such a way that the first three degrees of freedom make up an elbow manipulator, and the last three degrees of freedom is a spherical wrist attached to the end of the arm. This spherical wrist alone is built up by three single degree of freedom revolute joints, where the rotation axes intersect in the wrist center point. Thus the two links in between will have zero length and zero mass.

Examining the manipulator closer, it is discovered that some freedom is given to the choice of how to model joint 4. Actually, modeling the last three joints as a spherical wrist is not the desired choice, because the two links in between (link 4 and 5) do not have zero length and

mass. To compensate for this, it is found convenient to interpret the manipulator such that joint 3 and 4 has their center point in common, and joint 5 and 6 has their center point in common. In that case it is link 3 and link 5 which is modeled with zero length and mass. Figure 3.2 shows a symbolic representation of the manipulator by this interpretation, including how the frames have been attached to the links

The coordinate systems oriented according to Denavit-Hartenberg algorithm it allows get the product of basic rotation matrices around t the z-axis for each joint

$$R_{z,\theta} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.1}$$

where θ is the rotation angle. According to Figure 4.2, substituting q instead of θ and multiply matrix of basic rotation on addition matrix which turn the coordinate system(around axes x, y, z) according to the joint position. The result becomes

$$A_{i-1}^{i} = \begin{bmatrix} R_{i-1}^{i} & p_i \\ 0 & 1 \end{bmatrix} \tag{3.2}$$

$$A_0^1 = \begin{bmatrix} \cos(q1) & -\sin(q1) & 0 & 0 \\ \sin(q1) & \cos(q1) & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & a_1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & e_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos(q1) & 0 & \sin(q1) & a_1\cos(q1) \\ \sin(q1) & 0 & -\cos(q1) & a_1\sin(q1) \\ 0 & 1 & 0 & e_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_0^1 = \begin{bmatrix} \cos(q1) & 0 & \sin(q1) & a_1\cos(q1) \\ \sin(q1) & 0 & -\cos(q1) & a_1\sin(q1) \\ 0 & 1 & 0 & e_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3}$$

$$A_1^2 = \begin{bmatrix} \cos(q2) & -\sin(q2) & 0 & a_2\cos(q2) \\ \sin(q2) & \cos(q2) & 0 & a_2\sin(q2) \\ 0 & 0 & 1 & e_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

$$A_2^3 = \begin{bmatrix} \cos(q3) & 0 & \sin(q3) & 0 \\ \sin(q3) & 0 & -\cos(q3) & 0 \\ 0 & 1 & 0 & e_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.5}$$

$$A_3^4 = \begin{bmatrix} \cos(q4) & 0 & \sin(q4) & 0 \\ \sin(q4) & 0 & -\cos(q4) & 0 \\ 0 & 1 & 0 & e_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.6}$$

$$A_4^5 = \begin{bmatrix} \cos(q5) & 0 & \sin(q5) & 0 \\ \sin(q5) & 0 & -\cos(q5) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.7}$$

$$A_5^6 = \begin{bmatrix} \cos(q6) & -\sin(q6) & 0 & 0 \\ \sin(q6) & \cos(q6) & 0 & 0 \\ 0 & 0 & 1 & e_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.8)$$

Multiplying the received matrix, we obtain the solution for the forward kinematic task.

$$T_0^6 = A_0^1 A_1^2 A_2^3 A_3^4 A_4^5 A_5^6 \quad (3.9)$$

But for dynamic model we need only rotation matrices $R_0^2, R_0^3, R_0^4, R_0^5, R_0^6$ which we get from homogeneous matrices

$R_0^2 = R_0^1 R_1^2$,(3.10) $R_0^3 = R_0^2 R_2^3$,(3.11) $R_0^4 = R_0^3 R_3^4$,(3.12)

$R_0^5 = R_0^4 R_4^5$,(3.13) $R_0^6 = R_0^5 R_5^6$,(3.14)

3.3.2. Algorithm based on product of exponential fotmula

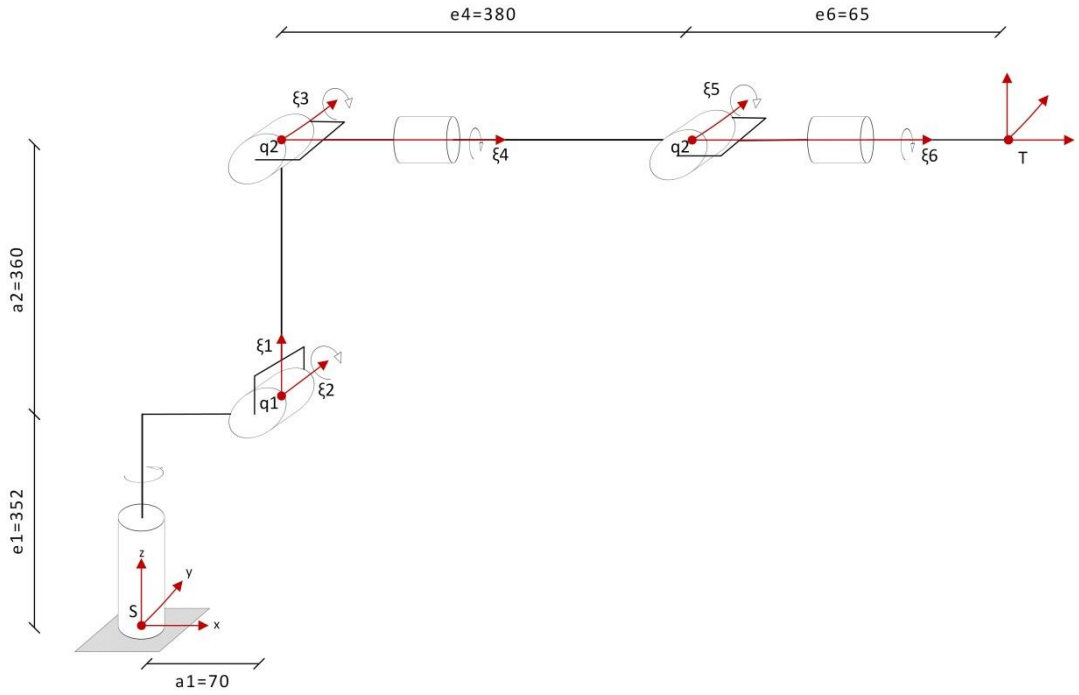


Figure 3.3 Schematic representation of manipulator IRB 140

In section 3.3.1 were describing the kinematic characteristics of manipulator. With the help of Denavit-Hartenberg algorithm with tacking kinematic characteristic into account were calculate homogeneous matrix T_0^6 which determine the position of six link of manipulator.

In this section also using the kinematic characteristics of manipulator described earlier. Determine the homogeneous matrix using alternative method based on product of exponential matrices.

- Determine the base configuration on manipulator

$$g_{ST}(0) = \begin{bmatrix} 1 & 0 & 0 & a_1 + e_4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & e_1 + e_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.15)$$

- For each link determine axis of rotation this parameter will be characterized by the vectorω. Also determine position of joint in space this parameter will be characterized by vector q.

$\omega_1 = \begin{bmatrix}1\\0\\0\end{bmatrix}$,(3.16)	$\omega_2 = \begin{bmatrix}0\\1\\0\end{bmatrix}$,(3.17)	$\omega_3 = \begin{bmatrix}0\\0\\1\end{bmatrix}$,(3.18)
$q_1 = \begin{bmatrix}a_1\\0\\e_1\end{bmatrix}$,(3.19)	$q_2 = \begin{bmatrix}a_1\\0\\e_1+e_2\end{bmatrix}$,(3.20)	$q_3 = T = \begin{bmatrix}a_1+e_1\\0\\e_1+e_2\end{bmatrix}$.(3.21)

- Find the twists (2.16) for each link, this six-dimensional vector characterized the revolute joint

$$\xi_1 = \begin{bmatrix}0\\-a_1\\0\\0\\0\\1\end{bmatrix},(3.22) \qquad \xi_2 = \begin{bmatrix}-e_1\\0\\a_1\\0\\1\\0\end{bmatrix},(3.23) \qquad \xi_3 = \begin{bmatrix}-e_1-e_2\\0\\a_1\\0\\1\\0\end{bmatrix},(3.24)$$

$$\xi_4 = \begin{bmatrix}0\\e_1+e_2\\a_1\\1\\0\\0\end{bmatrix},(3.25) \qquad \xi_5 = \begin{bmatrix}-e_1-e_2\\0\\e_4+a_1\\0\\1\\0\end{bmatrix},(3.26) \qquad \xi_6 = \begin{bmatrix}0\\e_1+e_2\\0\\1\\0\\0\end{bmatrix},(3.27)$$

- Find the transformation matrix (2.13) describing the joint motion for each joint.

$$e^{\xi_1\theta_1} = \begin{bmatrix}\cos(\theta1) & -\sin(\theta1) & 0 & a_1(1-\cos(\theta1))\\ \sin(\theta1) & \cos(\theta1) & 0 & -a_1\sin(\theta1)\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \quad (3.28)$$

$$e^{\xi_2\theta_2} = \begin{bmatrix}\cos(\theta2) & 0 & \sin(\theta2) & a_1(1-\cos(\theta2)) - e_1\sin(\theta2)\\ 0 & 1 & 0 & 0\\ -\sin(\theta2) & 0 & \cos(\theta2) & a_1\sin(\theta2) + e_1(1-\cos(\theta2))\\ 0 & 0 & 0 & 1\end{bmatrix} \quad (3.29)$$

$$e^{\xi_3\theta_3} = \begin{bmatrix}\cos(\theta3) & 0 & \sin(\theta3) & a_1(1-\cos(\theta3)) - \sin(\theta3)(e_1+e_2)\\ 0 & 1 & 0 & 0\\ -\sin(\theta3) & 0 & \cos(\theta3) & a_1\sin(\theta3) + e_1(1-\cos(\theta3))\\ 0 & 0 & 0 & 1\end{bmatrix} \quad (3.30)$$

$$e^{\xi_4\theta_4} = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos(\theta4) & -\sin(\theta4) & \sin(\theta4)(e_1+e_2)\\ 0 & \sin(\theta4) & \cos(\theta4) & (1-\cos(\theta4))(e_1+e_2)\\ 0 & 0 & 0 & 1\end{bmatrix} \quad (3.31)$$

$$e^{\xi_5\theta_5} = \begin{bmatrix}\cos(\theta5) & 0 & \sin(\theta5) & (1-\cos(\theta5))(a_1+e_4) - \sin(\theta5)(e_1+e_2)\\ 0 & 1 & 0 & 0\\ -\sin(\theta5) & 0 & \cos(\theta5) & \sin(\theta5)\,(a_1+e_4) + (1-\cos(\theta5))(e_1+e_2)\\ 0 & 0 & 0 & 1\end{bmatrix} \quad (3.32)$$

$$e^{\xi_6\theta_6} = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos(\theta6) & -\sin(\theta6) & \sin(\theta6)(e_1+e_2)\\ 0 & \sin(\theta6) & \cos(\theta6) & (1-\cos(\theta6))(e_1+e_2)\\ 0 & 0 & 0 & 1\end{bmatrix} \quad (3.33)$$

- According to the Equation (2.18) we get the solution for Forward kinematic task

$$g_{st}(\theta) = e^{\hat{\xi}_1\theta_1}e^{\hat{\xi}_2\theta_2}e^{\hat{\xi}_3\theta_3}e^{\hat{\xi}_4\theta_4}e^{\hat{\xi}_5\theta_5}e^{\hat{\xi}_6\theta_6}g_{st}(0), \tag{3.34}$$

3.4. Parameter estimation

This section describes how the dynamic parameters are estimated. It is mentioned in Section 3.2 that the parameters are estimated quite roughly. Still they should be close enough to the real unknown parameters that simulations show a behavior that is somewhat in accordance to the behavior of a perfect model.

The centers of mass of the four links have been estimated by studying the manipulator thoroughly, assuming the links have uniform mass density. Figure 3.4 shows the estimated centers of mass with colored dots. Link 1 has a red dot, link 2 has a green dot, link 4 has a blue dot, and link 6 has a yellow dot. Note that viewing from the back in Figure 4.3(a), link 4 and 6 have their centers of mass along the same line perpendicular to the paper.

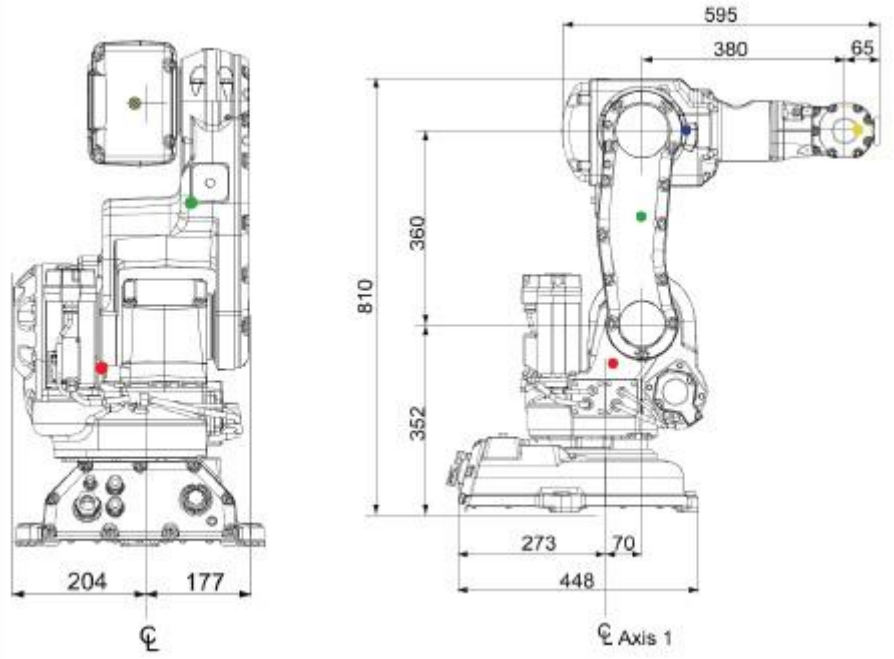


Figure 3.4 Location of centers of mass

Vectors between the origins of the frames are defined precisely by the dimensions in Figure 3.4. Vectors from the origins of the frames to the centers of mass are calculated by first computing the scale of the figure, and then multiplying the scale with the lengths measured by a ruler. The clever way of attaching frames to the links in the Newton-Euler formulation make all length vectors independent of the configuration of the manipulator. The results are given below (lengths in meters).

$$r_{0,c1} = [0.014 \;\; -0.264 \;\; 0.067]^T, \tag{3.35}$$

$$r_{1,c2} = [0.201 \;\; 0 \;\; -0.070]^T, \tag{3.36}$$

$$r_{2,c3} = [0 \;\; 0 \;\; 0]^T, \tag{3.37}$$

$$r_{3,c4} = [0\ \ 0.080\ \ 0]^T, \tag{3.38}$$

$$r_{4,c5} = [0\ \ 0\ \ 0]^T, \tag{3.39}$$

$$r_{5,c6} = [0\ \ 0\ \ 0.029]^T, \tag{3.40}$$

$$r_{0,1} = [0.070\ \ -0.352\ \ 0]^T, \tag{3.41}$$

$$r_{1,2} = [0.360\ \ 0\ \ 0]^T, \tag{3.42}$$

$$r_{2,3} = [0\ \ 0\ \ 0]^T \tag{3.43}$$

$$r_{3,4} = [0\ \ 0.380\ \ 0]^T, \tag{3.44}$$

$$r_{4,5} = [0\ \ 0\ \ 0]^T, \tag{3.45}$$

$$r_{5,6} = [0\ \ 0\ \ 0.065]^T. \tag{3.46}$$

Estimating the inertia parameters are definitely the most difficult task. The irregular shapes of the links makes it highly complicated to come up with realistic parameters without performing some kind of identification. As a fair simplification the links are modeled as cylindrical links with uniform mass density, where the center of mass of each link is the geometric center of the cylinder. Figure 3.5 shows an example of how this simplification can be applied on link 2

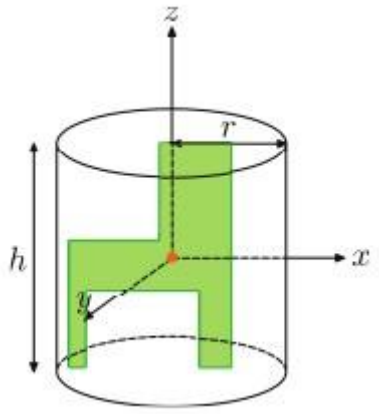


Figure 3.5 Example of the link 2 as cylinder

The green figure illustrates link 2 viewed from the back, and the orange dot is the center of mass.

3.5. The inertia tensor

The inertia tensor of such a cylinder can be determine with the help of follow equations

- The rotation axis is along to the cylinder axis:

$$J = \frac{1}{2}mr^2. \tag{3.47}$$

- The rotation axis is perpendicular to the cylinder axis and goes through its center of mass:

$$J = \frac{1}{12}mh^2 + \frac{1}{4}mr^2 \tag{3.48}$$

where m is the mass, r is the radius and h is the height of the cylinder. The cross products are identically zero such that the inertia tensor becomes a diagonal matrix in its principal axis form.

Determining the the mass, radius and height of the cylinders is kind of a constrained task, where the constraints are that the total mass must be 98 kg (including the base), and that the radius and height of the cylinders match the dimensions of the manipulator given in Figure 3.4. Like the actual links do, the cylinders will also overlap each other since the centers of mass are not geometrically right in between two frames, and it was assumed uniform mass density.

It is fair to believe that the mass density of every link is approximately equal. The links are constructed of a shell of metal with components such as motors, gearboxes, cables and belts on the inside. In addition, large proportions of the total volume is just air in between these components. By a trial-and-error approach, the masses, radii and heights was eventually found to match the physical shape of the manipulator using a mutual mass density of 1500 кг/м^3. The parameter values are given in Table 3.1, where the missing mass of 23 kg is allocated the manipulator base. To make a comparison, the mass density of steel is 7850 кг/м^{3}according to [9]. That is for massive steel, such that assuming a mass density of the links of about the fifth the mass density for steel seems satisfying.

Table 3.1 – Parameters of cylinders

Звено	Масса, кг	Радиус, м	Высота, м
1	27	0.191	0.363
2	22	0.151	0.515
3	-	-	-
4	25	0.115	0.583
5	-	-	-
6	1	0.044	0.107

Note that the orientation of the attached frame determines the coordination of the principal moments of inertia. Since in this work represents two methods below is giving the inertia tensors for each of case

3.5.1. The inertia tensor for algorithm of Denavit-Hartenberg

$$I_{DH,1} = \begin{bmatrix} \frac{1}{12}m_1h_1^2 + \frac{1}{4}m_1r_1^2 & 0 & 0 \\ 0 & \frac{1}{2}m_1r_1^2 & 0 \\ 0 & 0 & \frac{1}{12}m_1h_1^2 + \frac{1}{4}m_1r_1^2 \end{bmatrix} \quad (3.49)$$

$$I_{DH,2} = \begin{bmatrix} \frac{1}{2}m_2r_2^2 & 0 & 0 \\ 0 & \frac{1}{12}m_2h_2^2 + \frac{1}{4}m_2r_2^2 & 0 \\ 0 & 0 & \frac{1}{12}m_2h_2^2 + \frac{1}{4}m_2r_2^2 \end{bmatrix}, \quad (3.50)$$

$$I_{DH,3} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (3.51)$$

$$I_{DH,4} = \begin{bmatrix} \frac{1}{12}m_4h_4^2 + \frac{1}{4}m_4r_4^2 & 0 & 0 \\ 0 & \frac{1}{2}m_4r_4^2 & 0 \\ 0 & 0 & \frac{1}{12}m_4h_4^2 + \frac{1}{4}m_4r_4^2 \end{bmatrix} \quad (3.52)$$

$$I_{DH,5} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (3.53)$$

$$I_{DH,6} = \begin{bmatrix} \frac{1}{12}m_6h_6^2 + \frac{1}{4}m_6r_6^2 & 0 & 0 \\ 0 & \frac{1}{12}m_6h_6^2 + \frac{1}{4}m_6r_6^2 & 0 \\ 0 & 0 & \frac{1}{2}m_6r_6^2 \end{bmatrix} \quad (3.54)$$

3.5.2. The inertia tensors for algorithm based on product of exponential formula

$$I_{exp,1} = \begin{bmatrix} \frac{1}{12}m_1h_1^2 + \frac{1}{4}m_1r_1^2 & 0 & 0 \\ 0 & \frac{1}{12}m_1h_1^2 + \frac{1}{4}m_1r_1^2 & 0 \\ 0 & 0 & \frac{1}{2}m_1r_1^2 \end{bmatrix} \quad (3.55)$$

$$I_{exp,2} = \begin{bmatrix} \frac{1}{12}m_2h_2^2 + \frac{1}{4}m_2r_2^2 & 0 & 0 \\ 0 & \frac{1}{12}m_2h_2^2 + \frac{1}{4}m_2r_2^2 & 0 \\ 0 & 0 & \frac{1}{2}m_2r_2^2 \end{bmatrix} \quad (3.56)$$

$$I_{exp,3} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (3.57)$$

$$I_{exp,4} = \begin{bmatrix} \frac{1}{2}m_4r_4^2 & 0 & 0 \\ 0 & \frac{1}{12}m_4h_4^2 + \frac{1}{4}m_4r_4^2 & 0 \\ 0 & 0 & \frac{1}{12}m_4h_4^2 + \frac{1}{4}m_4r_4^2 \end{bmatrix} \quad (3.58)$$

$$I_{exp,5} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (3.59)$$

$$I_{exp,6} = \begin{bmatrix} \frac{1}{2}m_6r_6^2 & 0 & 0 \\ 0 & \frac{1}{12}m_6h_6^2 + \frac{1}{4}m_6r_6^2 & 0 \\ 0 & 0 & \frac{1}{12}m_6h_6^2 + \frac{1}{4}m_6r_6^2 \end{bmatrix} \quad (3.60)$$

4. The dynamic model

In the Chapter 2 were described two methods of determination the dynamic model

- The recursive method based on Newton-Euler formulation
- The method based on product of exponential formula

In the Chapter 3, a system description of the IRB 140 was presented. In this Chapter is presented calculation of dynamic model based on methods described earlier.

Since the manipulator has a six degrees of freedom, even the simplified system is quite complex. Actually, the final torque equations in the model are so huge that they do not even suit to be shown in this text. Therefore, to let this chapter be clear and easy to follow, all equations are kept in their symbolic form. Attachment A2 shows how the model has been computed in Maple by adjusting the framework and Appendix A3.

4.1. Method based on Newton-Euler formulation

4.1.1. Froward recursion

The forward recursion describes the linear and angular motion of the links, starting with link 1 and ending with link 6. The algorithm is described in Section 2.9.3, and it is just a matter of substituting in the general equations for an n-link manipulator.

As a part of the forward recursion it is necessary to compute $b_i$the axis of rotation for each joint *i* expressed in frame *i*.. The rotation axis in frame 0 is given directly as axis z

$$z_0 = [0\ \ 0\ \ 1]^T \tag{4.1}$$

and then the rotation axes for the joints are computed by Equation (2.39) as

$$b_1 = (R_1^0)^T z_0 = [0\ \ 1\ \ 0]^T, \tag{4.2}$$

$$b_2 = (R_2^1)^T R_1^0 z_0 = [0\ \ 0\ \ 1]^T, \tag{4.3}$$

$$b_3 = (R_3^2)^T R_2^0 z_0 = [0\ \ 1\ \ 0]^T, \tag{4.3}$$

$$b_4 = (R_4^3)^T R_3^0 z_0 = [0\ \ 1\ \ 0]^T, \tag{4.4}$$

$$b_5 = (R_5^4)^T R_5^0 z_0 = [0\ \ 1\ \ 0]^T, \tag{4.5}$$

$$b_6 = (R_6^5)^T R_6^0 z_0 = [0\ \ 0\ \ 1]^T. \tag{4.6}$$

Due to the coupled kinematics, these rotation axes will normally be functions of q just like the rotation matrices. They will depend on how the coordinate frames are defined, and therefore directly influence the efficiency of the Newton-Euler formulation. By inspecting how the frames are defined in Figure 3.2, it can be seen that when looking from frame *i* into frame *i-1*, the angular velocity $\omega_i$does not depend on $q_i$itself, but completely on the axis of rotation. Consequently the rotation axes $b_i$are not depending on q.

Link 1

The initial conditions are

$$\omega_0 = \alpha_0 = a_{c,0} = a_{e,0} = 0. \tag{4.8}$$

Angular velocity and acceleration are calculated from Equation (2.38) and (2.40) respectively, and becomes

$$\omega_1 = b_1 \dot{q}_1, \tag{4.9}$$

$$\alpha_1 = b_1\ddot{q}_1 + \omega_1 \times b_1\dot{q}_1. \quad (4.10)$$

Acceleration of the end of the link and the center of the link are calculated from Equation (2.46) and (2.47) respectively, and becomes

$$a_{e,1} = \dot{\omega}_1 \times r_{0,1} + \omega_1 \times (\omega_1 \times r_{0,1}), \quad (4.11)$$

$$a_{c,1} = \dot{\omega}_1 \times r_{0,c1} + \omega_1 \times (\omega_1 \times r_{0,c1}), \quad (4.12)$$

Using the same equations as for the link 1, we can get the angular velocity and acceleration, and acceleration of the end of the link and the center of mass for each link.

Link 2

$$\omega_2 = (R_2^1)^T \omega_1 + b_2\dot{q}_2, \quad (4.13)$$

$$\alpha_2 = (R_2^1)^T \alpha_1 + b_2\ddot{q}_2 + \omega_2 \times b_2\dot{q}_2, \quad (4.14)$$

$$a_{e,2} = (R_2^1)^T a_{e,1} + \dot{\omega}_2 \times r_{1,2} + \omega_2 \times (\omega_2 \times r_{1,2}), \quad (4.15)$$

$$a_{c,2} = (R_2^1)^T a_{e,1} + \dot{\omega}_2 \times r_{1,c2} + \omega_2 \times (\omega_2 \times r_{1,c2}), \quad (4.16)$$

Link 3

$$\omega_3 = (R_3^2)^T \omega_2 + b_3\dot{q}_3, \quad (4.17)$$

$$\alpha_3 = (R_3^2)^T \alpha_2 + b_3\ddot{q}_3 + \omega_3 \times b_3\dot{q}_3, \quad (4.18)$$

$$a_{e,3} = (R_3^2)^T a_{e,2}, \quad (4.19)$$

$$a_{c,3} = (R_3^2)^T a_{e,2}, \quad (4.20)$$

Link 4`

$$\omega_4 = (R_4^3)^T \omega_3 + b_4\dot{q}_4, \quad (4.21)$$

$$\alpha_4 = (R_4^3)^T \alpha_3 + b_4\ddot{q}_4 + \omega_4 \times b_4\dot{q}_4, \quad (4.22)$$

$$a_{e,4} = (R_4^3)^T a_{e,3} + \dot{\omega}_4 \times r_{3,4} + \omega_4 \times (\omega_4 \times r_{3,4}), \quad (4.23)$$

$$a_{c,4} = (R_4^3)^T a_{e,3} + \dot{\omega}_4 \times r_{3,c4} + \omega_4 \times (\omega_4 \times r_{3,c4}), \quad (4.24)$$

Link 5

$$\omega_5 = (R_5^4)^T \omega_4 + b_5\dot{q}_5, \quad (4.25)$$

$$\alpha_5 = (R_5^4)^T \alpha_2 + b_5\ddot{q}_5 + \omega_5 \times b_5\dot{q}_5, \quad (4.26)$$

$$a_{e,5} = (R_5^4)^T a_{e,4}, \quad (4.27)$$

$$a_{c,5} = (R_5^4)^T a_{e,4}, \quad (4.28)$$

Link 6

$$\omega_6 = (R_6^5)^T \omega_5 + b_6\dot{q}_6, \quad (4.29)$$

$$\alpha_6 = (R_6^5)^T \alpha_5 + b_6\ddot{q}_6 + \omega_6 \times b_6\dot{q}_6, \quad (4.30)$$

$$a_{c,4} = (R_6^5)^T a_{e,5} + \dot{\omega}_6 \times r_{5,c6} + \omega_6 \times (\omega_6 \times r_{5,c6}), \quad (4.31)$$

Note that there is no need to compute $a_{e,6}$because $a_{e,i}$is only used to compute $a_{e,i+1}$ (and there is no link 7).

4.1.2. Backward recursion

The backward recursion calculates the forces and joint torques acting on the links, starting with link 6 and ending with link 1. Determining the joint torques is the ultimate goal of the Newton-Euler formulation, because the torques are the externally applied input to the model. As for the forward recursion, the algorithm is described in Section 2.9.3 and it is just a matter of substituting in the general equations for an n-link manipulator. Note that the force equation includes the gravity vector. This gravity vector differs for each link, but can easily be calculated with the use of rotation matrices as shown in the recursions below

Link 6

The terminal conditions are

$$f_7 = \tau_7 = 0. \tag{4.32}$$

The gravity vector becomes

$$g_6 = (R_6^0)^T g_0, \tag{4.33}$$

Where g_0 is the gravity vector in the inertial frame defined as

$$g_0 = [0 \;\; 0 \;\; -g]^T. \tag{4.34}$$

The force and joint torque exerted on the link are calculated from Equation (2.33) and (2.36) respectively, and becomes

$$f_6 = m_6 a_{c,6} - m_6 g_6, \tag{4.35}$$

$$\tau_6 = -f_6 \times r_{5c,6} + \omega_6 \times (I_6 \omega_6) + I_6 \alpha_6. \tag{4.36}$$

Using the same equations as for the link 6, we can get the gravity vector, and force and joint torque for each link

Link 5

$$g_5 = (R_5^0)^T g_0, \tag{4.37}$$

$$f_5 = R_6^5 f_6, \tag{4.38}$$

$$\tau_6 = R_6^5 \tau_6 + \omega_5 \times (I_5 \omega_5) + I_5 \alpha_5, \tag{4.39}$$

Link 4

$$g_4 = (R_4^0)^T g_0, \tag{4.40}$$

$$f_4 = R_5^4 f_5 + m_4 a_{c,4} - m_4 g_4, \tag{4.41}$$

$$\tau_4 = R_5^4 \tau_5 - f_4 \times r_{3,c4} + R_5^4 f_5 \times r_{4,c4} + \omega_4 \times (I_4 \omega_4) + I_4 \alpha_4, \tag{4.42}$$

Link 3

$$g_3 = (R_3^0)^T g_0, \tag{4.43}$$

$$f_3 = R_4^3 f_4, \tag{4.44}$$

$$\tau_3 = R_4^3 \tau_4 + \omega_3 \times (I_3 \omega_3) + I_3 \alpha_3, \tag{4.45}$$

Link 2

$$g_2 = (R_2^0)^T g_0, \tag{4.46}$$

$$f_2 = R_3^2 f_3 + m_2 a_{c,2} - m_2 g_2, \tag{4.47}$$

$$\tau_2 = R_3^2 \tau_3 - f_2 \times r_{1,c2} + R_3^2 f_3 \times r_{2,c2} + \omega_2 \times (I_2 \omega_2) + I_2 \alpha_2, \tag{4.48}$$

Link 1

$$g_1 = (R_1^0)^T g_0, \tag{4.49}$$

$$f_1 = R_2^1 f_2 + m_1 a_{c,1} - m_1 g_1, \tag{4.50}$$

$$\tau_1 = R_2^1 \tau_2 - f_1 \times r_{0,c1} + R_2^1 f_2 \times r_{1,c1} + \omega_1 \times (I_1 \omega_1) + I_1 \alpha_1. \tag{4.51}$$

4.1.3. Comments

The results in this chapter are interesting and verifies why the Newton-Euler formulation often is the preferred choice for manipulators with many degrees of freedom. The recursive algorithm is easy to implement and consequently there are small chances of doing any mistakes in the derivation. Strange behavior of the model can mostly be connected to the preparations such as the set-up of the kinematic chain and the frames, rotation matrices, vector definitions and inertia tensors.

Note that even though link 3 and 5 have zero length and mass, they still have to be considered in the recursions. The Newton-Euler formulation is based on a kinematic chain with only single degree-of-freedom joints, such that *n* degrees of freedom always lead to *n* steps in each recursion. However, some terms in the expressions for link 3 and 5 are canceled out.

One interesting insight in the Newton-Euler formulation comes from the final joint torque vectors in the backward recursion. All joints in the kinematic chain are single degree-of-freedom joints, such that the torques applied are scalars about the rotation axes computed in Equations (4.2)-(4.7). The other two elements of the torque vectors can be explained as follows. When applying torque to any of the joints, this will also generate torque components about the other axes of the joints due to the coupled kinematics in the system. These torque components are not included in the dynamic model because they do not induce motion (not affecting q), but still it is valuable information about the physics of the manipulator. If the joints in the manipulator are not constructed to physically resist these torque quantities, the joints will break.

Although utilizing the Newton-Euler formulation appears to be quite easy, the complexity of the resulting model should be emphasized. The basic idea behind recursion is that the solution to a problem depends on solutions to smaller instances on the same problem. The backward recursion of link 1 depends on the backward recursion of link 2, which depends of the backwards recursion of link 3, and so on. All in all the backward recursion of link 1 is directly dependent on all 11 steps back to the forward recursion of link 1. Thus it should not be a surprise that calculating $\tau 1$ from Equation (4.51) results in a huge vector.

4.2. The dynamic model based on product of exponential formula

In the case when forward kinematics are specified using the product of exponential formula, then it is possible to get more explicit formulas for the inertia and Coriolis matrices in case of n-link manipulator.

In order to obtain the components of the dynamic equation, it is necessary determine the following parameters

- Inertia matrixM_i(2.50)for each of links

$$M_i = \begin{bmatrix} m_i \mathrm{I} & 0 \\ 0 & I_i \end{bmatrix},$$

where$m_i$isamass of link, I_i is a inertia tensor of i-th link. The inertia tensors for links have been determined in section 3.5.2 in matrix form (3.55) - (3.60) .Hence, using the known parameters we can determine the inertia matrix for each of links.

$$M_1 = \begin{bmatrix} m_1 \mathrm{I} & 0 \\ 0 & I_1 \end{bmatrix}, \tag{4.52}$$

$$M_2 = \begin{bmatrix} m_2 \mathrm{I} & 0 \\ 0 & I_2 \end{bmatrix}, \tag{4.53}$$

$$M_3 = \begin{bmatrix} m_3 \mathrm{I} & 0 \\ 0 & I_3 \end{bmatrix} = \mathrm{Z}, \tag{4.54}$$

$$M_4 = \begin{bmatrix} m_4 \mathrm{I} & 0 \\ 0 & I_4 \end{bmatrix}, \tag{4.55}$$

$$M_5 = \begin{bmatrix} m_5 \mathrm{I} & 0 \\ 0 & I_5 \end{bmatrix} = \mathrm{Z}, \tag{4.56}$$

$$M_6 = \begin{bmatrix} m_6 \mathrm{I} & 0 \\ 0 & I_6 \end{bmatrix}, \tag{4.57}$$

where Z is a zero matrix.

- Adjoint transformation matrix $A_{ij} \in R^{6\times6}$

For the beginning determine nonzero elements of matrix which need to be calculate. According to (2.51) we get the 6×6 matrix:

$$A = \begin{bmatrix} \mathrm{I} & 0 & 0 & 0 & 0 & 0 \\ A_{21} & \mathrm{I} & 0 & 0 & 0 & 0 \\ A_{31} & A_{32} & \mathrm{I} & 0 & 0 & 0 \\ A_{41} & A_{42} & A_{43} & \mathrm{I} & 0 & 0 \\ A_{51} & A_{52} & A_{53} & A_{54} & \mathrm{I} & 0 \\ A_{61} & A_{62} & A_{63} & A_{64} & A_{65} & \mathrm{I} \end{bmatrix},$$

Each element of matrix calculate according to (). And results write as

$$A_{21} = Ad^{-1}_{(e^{\xi_2\theta_2}e^{\xi_2\theta_2})} = \begin{bmatrix} R_{22}^T & -{}^{T}_{21}\hat{p}_{22} \\ 0 & R_{22}^T \end{bmatrix}, \tag{4.59}$$

$$A_{31} = Ad^{-1}_{(e^{\xi_2\theta_2}e^{\xi_3\theta_3})} = \begin{bmatrix} R_{23}^T & -R_{23}^T\hat{p}_{23} \\ 0 & R_{23}^T \end{bmatrix}, \tag{4.60}$$

$$A_{32} = Ad^{-1}_{(e^{\xi_3\theta_3}e^{\xi_3\theta_3})} = \begin{bmatrix} R_{33}^T & -R_{33}^T\hat{p}_{33} \\ 0 & R_{33}^T \end{bmatrix}, \tag{4.61}$$

$$A_{41} = Ad^{-1}_{(e^{\xi_2\theta_2}e^{\xi_4\theta_4})} = \begin{bmatrix} R_{24}^T & -R_{24}^T\hat{p}_{24} \\ 0 & R_{24}^T \end{bmatrix}, \quad (4.62)$$

$$A_{42} = Ad^{-1}_{(e^{\xi_3\theta_3}e^{\xi_4\theta_4})} = \begin{bmatrix} R_{34}^T & -R_{34}^T\hat{p}_{34} \\ 0 & R_{34}^T \end{bmatrix}, \quad (4.63)$$

$$A_{43} = Ad^{-1}_{(e^{\xi_4\theta_4}e^{\xi_4\theta_4})} = \begin{bmatrix} R_{44}^T & -R_{44}^T\hat{p}_{44} \\ 0 & R_{44}^T \end{bmatrix}, \quad (4.64)$$

$$A_{51} = Ad^{-1}_{(e^{\xi_2\theta_2}e^{\xi_5\theta_5})} = \begin{bmatrix} R_{25}^T & -R_{25}^T\hat{p}_{25} \\ 0 & R_{25}^T \end{bmatrix}, \quad (4.65)$$

$$A_{52} = Ad^{-1}_{(e^{\xi_3\theta_3}e^{\xi_5\theta_5})} = \begin{bmatrix} R_{35}^T & -R_{35}^T\hat{p}_{35} \\ 0 & R_{35}^T \end{bmatrix}, \quad (4.66)$$

$$A_{53} = Ad^{-1}_{(e^{\xi_4\theta_4}e^{\xi_5\theta_5})} = \begin{bmatrix} R_{45}^T & -R_{45}^T\hat{p}_{45} \\ 0 & R_{45}^T \end{bmatrix}, \quad (4.67)$$

$$A_{54} = Ad^{-1}_{(e^{\xi_5\theta_5}e^{\xi_5\theta_5})} = \begin{bmatrix} R_{55}^T & -R_{55}^T\hat{p}_{55} \\ 0 & R_{55}^T \end{bmatrix}, \quad (4.68)$$

$$A_{61} = Ad^{-1}_{(e^{\xi_2\theta_2}e^{\xi_6\theta_6})} = \begin{bmatrix} R_{26}^T & -R_{26}^T\hat{p}_{26} \\ 0 & R_{26}^T \end{bmatrix}, \quad (4.69)$$

$$A_{62} = Ad^{-1}_{(e^{\xi_3\theta_3}e^{\xi_6\theta_6})} = \begin{bmatrix} R_{36}^T & -R_{36}^T\hat{p}_{36} \\ 0 & R_{36}^T \end{bmatrix}, \quad (4.70)$$

$$A_{63} = Ad^{-1}_{(e^{\xi_4\theta_4}e^{\xi_6\theta_6})} = \begin{bmatrix} R_{46}^T & -R_{46}^T\hat{p}_{46} \\ 0 & R_{46}^T \end{bmatrix}, \quad (4.71)$$

$$A_{64} = Ad^{-1}_{(e^{\xi_5\theta_5}e^{\xi_6\theta_6})} = \begin{bmatrix} R_{56}^T & -R_{56}^T\hat{p}_{56} \\ 0 & R_{56}^T \end{bmatrix}, \quad (4.72)$$

$$A_{65} = Ad^{-1}_{(e^{\xi_6\theta_6}e^{\xi_6\theta_6})} = \begin{bmatrix} R_{66}^T & -R_{66}^T\hat{p}_{66} \\ 0 & R_{66}^T \end{bmatrix}. \quad (4.73)$$

Then we need determine the transformed inertia matrix M_i' which describe theinertia moments of each link relative to the base coordinate frame of manipulator. In Section 2.9.4 was given Equation (2.56) allows determine the Jacoby matrix, from this equation we need use the inverse adjoint matrix of i-thlink $Ad^T_{g^{-1}_{sl_i^{(0)}}}$. Equation (2.57) allows determine the transformed inertia matrix.

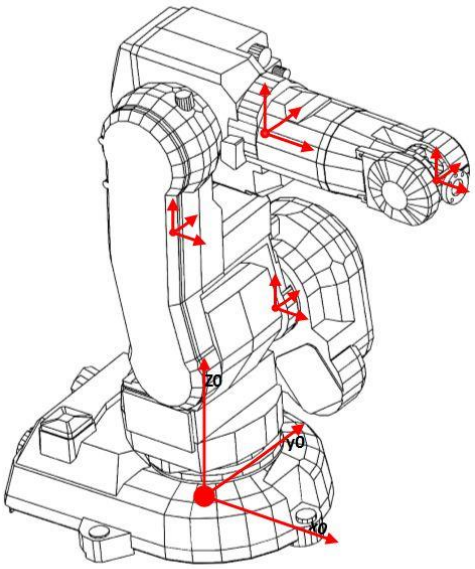


Figure 4.1 the coordinate system location of each links

Assuming that the link frames are initially aligned with the base frame and are located at the centers of mass of the links Figure 4.1, the transformed link inertia matrices have the form

$$M_i^{'} = Ad^{T}_{g^{-1}_{sl_i^{(0)}}} M_i Ad_{g^{-1}_{sl_i^{(0)}}} = \begin{bmatrix} \mathrm{I} & 0 \\ -\hat{p}_i & \mathrm{I} \end{bmatrix} \begin{bmatrix} m_i\mathrm{I} & 0 \\ 0 & I_i \end{bmatrix} \begin{bmatrix} \mathrm{I} & \hat{p}_i \\ 0 & \mathrm{I} \end{bmatrix} = \begin{bmatrix} m_i\mathrm{I} & m_i\hat{p}_i \\ -m_i\hat{p}_i & I_i \end{bmatrix}$$

wherep_i is the location of the origin of the i-th link frame relative to the base frame S.

Below is given the results in symbolic form:

$$M_1^{'} = \begin{bmatrix} m_1\mathrm{I} & m_1\hat{p}_1 \\ -m_1\hat{p}_1 & I_1 \end{bmatrix}, \tag{4.74}$$

$$M_2^{'} = \begin{bmatrix} m_2\mathrm{I} & m_2\hat{p}_2 \\ -m_2\hat{p}_2 & I_2 \end{bmatrix}, \tag{4.75}$$

$$M_3^{'} = \begin{bmatrix} m_3\mathrm{I} & m_3\hat{p}_3 \\ -m_3\hat{p}_3 & I_3 \end{bmatrix}, \tag{4.76}$$

$$M_4^{'} = \begin{bmatrix} m_4\mathrm{I} & m_4\hat{p}_4 \\ -m_4\hat{p}_4 & I_4 \end{bmatrix}, \tag{4.77}$$

$$M_i^{'} = \begin{bmatrix} m_5\mathrm{I} & m_5\hat{p}_5 \\ -m_5\hat{p}_5 & I_5 \end{bmatrix}, \tag{4.78}$$

$$M_i^{'} = \begin{bmatrix} m_6\mathrm{I} & m_6\hat{p}_6 \\ -m_6\hat{p}_6 & I_6 \end{bmatrix}, \tag{4.79}$$

Using all determined parameters we find the inertia matrix and Coriolis matrix for manipulator with 6 DOF and open-chain kinematic map based on equations (2.58) and (2.59)

$$M_{ij}(\theta) = \sum_{l=\max(i,j)}^{n} \xi_i^T A_{li}^T M_l^{'} A_{lj} \xi_j$$

$$C_{ij}(\theta) = \frac{1}{2}\sum_{k=1}^{n}\left(\frac{\partial M_{ij}}{\partial \theta_k} + \frac{\partial M_{ik}}{\partial \theta_j} - \frac{\partial M_{kj}}{\partial \theta_i}\right)\dot{\theta}_k$$

4.2.1. Inertia matrix

$$M(\theta) = \begin{bmatrix} M_{11} & M_{12} & M_{13} & M_{14} & M_{15} & M_{16} \\ M_{21} & M_{22} & M_{23} & M_{24} & M_{25} & M_{26} \\ M_{31} & M_{32} & M_{33} & M_{34} & M_{35} & M_{36} \\ M_{41} & M_{42} & M_{43} & M_{44} & M_{45} & M_{46} \\ M_{51} & M_{52} & M_{53} & M_{54} & M_{55} & M_{56} \\ M_{61} & M_{62} & M_{63} & M_{64} & M_{65} & M_{66} \end{bmatrix} \tag{4.80}$$

$$M_{11}(\theta) = \xi_1^T A_{11}^T M_1' A_{11}\xi_1 + \xi_1^T A_{21}^T M_2' A_{21}\xi_1 + \xi_1^T A_{31}^T M_3' A_{31}\xi_1 + \xi_1^T A_{41}^T M_4' A_{41}\xi_1 + \xi_1^T A_{51}^T M_5' A_{51}\xi_1 + \xi_1^T A_{61}^T M_6' A_{61}\xi_1$$

$$M_{12}(\theta) = \xi_1^T A_{21}^T M_2' A_{22}\xi_2 + \xi_1^T A_{31}^T M_3' A_{32}\xi_2 + \xi_1^T A_{41}^T M_4' A_{42}\xi_2 + \xi_1^T A_{51}^T M_5' A_{52}\xi_2 + \xi_1^T A_{61}^T M_6' A_{62}\xi_2$$

$$M_{13}(\theta) = \xi_1^T A_{31}^T M_3' A_{33}\xi_3 + \xi_1^T A_{41}^T M_4' A_{43}\xi_3 + \xi_1^T A_{51}^T M_5' A_{53}\xi_3 + \xi_1^T A_{61}^T M_6' A_{63}\xi_3$$

$$M_{14}(\theta) = \xi_1^T A_{41}^T M_4' A_{44}\xi_4 + \xi_1^T A_{51}^T M_5' A_{54}\xi_4 + \xi_1^T A_{61}^T M_6' A_{64}\xi_4$$

$$M_{15}(\theta) = \xi_1^T A_{51}^T M_5' A_{55}\xi_5 + \xi_1^T A_{61}^T M_6' A_{65}\xi_5$$

$$M_{16}(\theta) = \xi_1^T A_{61}^T M_6' A_{66}\xi_6$$

$$M_{21}(\theta) = \xi_2^T A_{22}^T M_2' A_{21}\xi_1 + \xi_2^T A_{32}^T M_3' A_{31}\xi_1 + \xi_2^T A_{42}^T M_4' A_{41}\xi_1 + \xi_2^T A_{52}^T M_5' A_{51}\xi_1 + \xi_2^T A_{62}^T M_6' A_{61}\xi_1$$

$$M_{22}(\theta) = \xi_2^T A_{22}^T M_2' A_{22}\xi_2 + \xi_2^T A_{32}^T M_3' A_{32}\xi_2 + \xi_2^T A_{42}^T M_4' A_{42}\xi_2 + \xi_2^T A_{52}^T M_5' A_{52}\xi_2 + \xi_2^T A_{62}^T M_6' A_{62}\xi_2$$

$$M_{23}(\theta) = \xi_2^T A_{32}^T M_3' A_{33}\xi_3 + \xi_2^T A_{42}^T M_4' A_{43}\xi_3 + \xi_2^T A_{52}^T M_5' A_{53}\xi_3 + \xi_2^T A_{62}^T M_6' A_{63}\xi_3$$

$$M_{24}(\theta) = \xi_2^T A_{42}^T M_4' A_{44}\xi_4 + \xi_2^T A_{52}^T M_5' A_{54}\xi_4 + \xi_2^T A_{62}^T M_6' A_{64}\xi_4$$

$$M_{25}(\theta) = \xi_2^T A_{52}^T M_5' A_{55}\xi_5 + \xi_2^T A_{62}^T M_6' A_{65}\xi_5$$

$$M_{26}(\theta) = \xi_2^T A_{62}^T M_6' A_{66}\xi_6$$

$$M_{31}(\theta) = \xi_3^T A_{33}^T M_3' A_{31}\xi_1 + \xi_3^T A_{43}^T M_4' A_{41}\xi_1 + \xi_3^T A_{53}^T M_5' A_{51}\xi_1 + \xi_3^T A_{63}^T M_6' A_{61}\xi_1$$

$$M_{32}(\theta) = \xi_3^T A_{33}^T M_3' A_{32}\xi_2 + \xi_3^T A_{43}^T M_4' A_{42}\xi_2 + \xi_3^T A_{53}^T M_5' A_{52}\xi_2 + \xi_3^T A_{63}^T M_6^{'} A_{62}\xi_2$$

$$M_{33}(\theta) = \xi_3^T A_{33}^T M_3' A_{33}\xi_3 + \xi_3^T A_{43}^T M_4' A_{43}\xi_3 + \xi_3^T A_{53}^T M_5' A_{53}\xi_3 + \xi_3^T A_{63}^T M_6' A_{63}\xi_3$$

$$M_{34}(\theta) = \xi_3^T A_{43}^T M_4' A_{44}\xi_4 + \xi_3^T A_{53}^T M_5' A_{54}\xi_4 + \xi_3^T A_{63}^T M_6' A_{64}\xi_4$$

$$M_{35}(\theta) = \xi_3^T A_{53}^T M_5' A_{55}\xi_5 + \xi_3^T A_{63}^T M_6' A_{65}\xi_5$$

$$M_{36}(\theta) = \xi_3^T A_{63}^T M_6' A_{66}\xi_6$$

$$M_{41}(\theta) = \xi_4^T A_{44}^T M_4' A_{41}\xi_1 + \xi_4^T A_{54}^T M_5' A_{51}\xi_1 + \xi_4^T A_{64}^T M_6' A_{61}\xi_1$$

$$M_{42}(\theta) = \xi_4^T A_{44}^T M_4' A_{42}\xi_2 + \xi_4^T A_{54}^T M_5' A_{52}\xi_2 + \xi_4^T A_{64}^T M_6' A_{62}\xi_2$$

$$M_{43}(\theta) = \xi_4^T A_{44}^T M_4' A_{43}\xi_3 + \xi_4^T A_{54}^T M_5' A_{53}\xi_3 + \xi_4^T A_{64}^T M_6' A_{63}\xi_3$$

$$M_{44}(\theta) = \xi_4^T A_{44}^T M_4' A_{44}\xi_4 + \xi_4^T A_{54}^T M_5' A_{54}\xi_4 + \xi_4^T A_{64}^T M_6' A_{64}\xi_4$$

$$M_{45}(\theta) = \xi_4^T A_{54}^T M_5' A_{55} \xi_5 + \xi_4^T A_{64}^T M_6' A_{65} \xi_5$$
$$M_{46}(\theta) = \xi_4^T A_{64}^T M_6' A_{66} \xi_6$$

$$M_{51}(\theta) = \xi_5^T A_{55}^T M_5' A_{51} \xi_1 + \xi_5^T A_{65}^T M_6' A_{61} \xi_1$$
$$M_{52}(\theta) = \xi_5^T A_{55}^T M_5' A_{52} \xi_2 + \xi_5^T A_{65}^T M_6' A_{62} \xi_2$$
$$M_{53}(\theta) = \xi_5^T A_{55}^T M_5' A_{53} \xi_3 + \xi_5^T A_{65}^T M_6' A_{63} \xi_3$$
$$M_{54}(\theta) = \xi_5^T A_{55}^T M_5' A_{54} \xi_4 + \xi_5^T A_{65}^T M_6' A_{64} \xi_4$$
$$M_{55}(\theta) = \xi_5^T A_{55}^T M_5' A_{55} \xi_5 + \xi_5^T A_{65}^T M_6' A_{65} \xi_5$$
$$M_{56}(\theta) = \xi_5^T A_{65}^T M_6' A_{66} \xi_6$$

$$M_{61}(\theta) = \xi_6^T A_{66}^T M_6' A_{61} \xi_1$$
$$M_{62}(\theta) = \xi_6^T A_{66}^T M_6' A_{62} \xi_2$$
$$M_{63}(\theta) = \xi_6^T A_{66}^T M_6' A_{63} \xi_3$$
$$M_{64}(\theta) = \xi_6^T A_{66}^T M_6' A_{64} \xi_4$$
$$M_{65}(\theta) = \xi_6^T A_{66}^T M_6' A_{65} \xi_5$$
$$M_{66}(\theta) = \xi_6^T A_{66}^T M_6' A_{66} \xi_6$$

4.2.2. Coriolis matrix

$$C(\theta) = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66} \end{bmatrix} \tag{4.81}$$

$$C_{11}(\theta) = \frac{1}{2}\left[\left(\frac{\partial M_{11}}{\partial \theta_1} + \frac{\partial M_{11}}{\partial \theta_1} - \frac{\partial M_{11}}{\partial \theta_1}\right)\dot{\theta}_1 + \left(\frac{\partial M_{11}}{\partial \theta_2} + \frac{\partial M_{12}}{\partial \theta_1} - \frac{\partial M_{21}}{\partial \theta_1}\right)\dot{\theta}_2 + \left(\frac{\partial M_{11}}{\partial \theta_3} + \frac{\partial M_{13}}{\partial \theta_1} - \frac{\partial M_{31}}{\partial \theta_1}\right)\dot{\theta}_3 \right.$$
$$\left. + \left(\frac{\partial M_{11}}{\partial \theta_4} + \frac{\partial M_{14}}{\partial \theta_1} - \frac{\partial M_{41}}{\partial \theta_1}\right)\dot{\theta}_4 + \left(\frac{\partial M_{11}}{\partial \theta_5} + \frac{\partial M_{15}}{\partial \theta_1} - \frac{\partial M_{51}}{\partial \theta_1}\right)\dot{\theta}_5 + \left(\frac{\partial M_{11}}{\partial \theta_6} + \frac{\partial M_{16}}{\partial \theta_1} - \frac{\partial M_{61}}{\partial \theta_1}\right)\dot{\theta}_6\right]$$

$$C_{12}(\theta) = \frac{1}{2}\left[\left(\frac{\partial M_{12}}{\partial \theta_1} + \frac{\partial M_{11}}{\partial \theta_2} - \frac{\partial M_{12}}{\partial \theta_1}\right)\dot{\theta}_1 + \left(\frac{\partial M_{12}}{\partial \theta_2} + \frac{\partial M_{12}}{\partial \theta_2} - \frac{\partial M_{21}}{\partial \theta_1}\right)\dot{\theta}_2 + \left(\frac{\partial M_{13}}{\partial \theta_3} + \frac{\partial M_{13}}{\partial \theta_2} - \frac{\partial M_{32}}{\partial \theta_1}\right)\dot{\theta}_3 \right.$$
$$\left. + \left(\frac{\partial M_{14}}{\partial \theta_4} + \frac{\partial M_{14}}{\partial \theta_2} - \frac{\partial M_{42}}{\partial \theta_1}\right)\dot{\theta}_4 + \left(\frac{\partial M_{15}}{\partial \theta_5} + \frac{\partial M_{15}}{\partial \theta_2} - \frac{\partial M_{52}}{\partial \theta_1}\right)\dot{\theta}_5 + \left(\frac{\partial M_{16}}{\partial \theta_6} + \frac{\partial M_{16}}{\partial \theta_2} - \frac{\partial M_{62}}{\partial \theta_1}\right)\dot{\theta}_6\right]$$

$$C_{13}(\theta) = \frac{1}{2}\left[\left(\frac{\partial M_{13}}{\partial \theta_1} + \frac{\partial M_{11}}{\partial \theta_3} - \frac{\partial M_{13}}{\partial \theta_1}\right)\dot{\theta}_1 + \left(\frac{\partial M_{13}}{\partial \theta_2} + \frac{\partial M_{12}}{\partial \theta_3} - \frac{\partial M_{23}}{\partial \theta_1}\right)\dot{\theta}_2 + \left(\frac{\partial M_{13}}{\partial \theta_3} + \frac{\partial M_{13}}{\partial \theta_3} - \frac{\partial M_{33}}{\partial \theta_1}\right)\dot{\theta}_3 \right.$$
$$\left. + \left(\frac{\partial M_{13}}{\partial \theta_4} + \frac{\partial M_{14}}{\partial \theta_3} - \frac{\partial M_{43}}{\partial \theta_1}\right)\dot{\theta}_4 + \left(\frac{\partial M_{13}}{\partial \theta_5} + \frac{\partial M_{15}}{\partial \theta_3} - \frac{\partial M_{53}}{\partial \theta_1}\right)\dot{\theta}_5 + \left(\frac{\partial M_{13}}{\partial \theta_6} + \frac{\partial M_{16}}{\partial \theta_3} - \frac{\partial M_{63}}{\partial \theta_1}\right)\dot{\theta}_6\right]$$

$$C_{14}(\theta) = \frac{1}{2}\left[\left(\frac{\partial M_{14}}{\partial \theta_1} + \frac{\partial M_{11}}{\partial \theta_4} - \frac{\partial M_{14}}{\partial \theta_1}\right)\dot{\theta}_1 + \left(\frac{\partial M_{14}}{\partial \theta_2} + \frac{\partial M_{12}}{\partial \theta_4} - \frac{\partial M_{24}}{\partial \theta_1}\right)\dot{\theta}_2 + \left(\frac{\partial M_{14}}{\partial \theta_3} + \frac{\partial M_{13}}{\partial \theta_4} - \frac{\partial M_{34}}{\partial \theta_1}\right)\dot{\theta}_3 \right.$$
$$\left. + \left(\frac{\partial M_{14}}{\partial \theta_4} + \frac{\partial M_{14}}{\partial \theta_4} - \frac{\partial M_{44}}{\partial \theta_1}\right)\dot{\theta}_4 + \left(\frac{\partial M_{14}}{\partial \theta_5} + \frac{\partial M_{15}}{\partial \theta_4} - \frac{\partial M_{54}}{\partial \theta_1}\right)\dot{\theta}_5 + \left(\frac{\partial M_{14}}{\partial \theta_6} + \frac{\partial M_{16}}{\partial \theta_4} - \frac{\partial M_{64}}{\partial \theta_1}\right)\dot{\theta}_6\right]$$

$$C_{15}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{15}}{\partial\theta_1}+\frac{\partial M_{11}}{\partial\theta_5}-\frac{\partial M_{15}}{\partial\theta_1}\right)\dot{\theta}_1+\left(\frac{\partial M_{15}}{\partial\theta_2}+\frac{\partial M_{12}}{\partial\theta_5}-\frac{\partial M_{25}}{\partial\theta_1}\right)\dot{\theta}_2+\left(\frac{\partial M_{15}}{\partial\theta_3}+\frac{\partial M_{13}}{\partial\theta_5}-\frac{\partial M_{35}}{\partial\theta_1}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{15}}{\partial\theta_4}+\frac{\partial M_{14}}{\partial\theta_5}-\frac{\partial M_{45}}{\partial\theta_1}\right)\dot{\theta}_4+\left(\frac{\partial M_{15}}{\partial\theta_5}+\frac{\partial M_{15}}{\partial\theta_5}-\frac{\partial M_{55}}{\partial\theta_1}\right)\dot{\theta}_5+\left(\frac{\partial M_{15}}{\partial\theta_6}+\frac{\partial M_{16}}{\partial\theta_5}-\frac{\partial M_{65}}{\partial\theta_1}\right)\dot{\theta}_6\right]$$

$$C_{16}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{16}}{\partial\theta_1}+\frac{\partial M_{11}}{\partial\theta_6}-\frac{\partial M_{16}}{\partial\theta_1}\right)\dot{\theta}_1+\left(\frac{\partial M_{16}}{\partial\theta_2}+\frac{\partial M_{12}}{\partial\theta_6}-\frac{\partial M_{26}}{\partial\theta_1}\right)\dot{\theta}_2+\left(\frac{\partial M_{16}}{\partial\theta_3}+\frac{\partial M_{13}}{\partial\theta_6}-\frac{\partial M_{36}}{\partial\theta_1}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{16}}{\partial\theta_4}+\frac{\partial M_{14}}{\partial\theta_6}-\frac{\partial M_{46}}{\partial\theta_1}\right)\dot{\theta}_4+\left(\frac{\partial M_{16}}{\partial\theta_5}+\frac{\partial M_{15}}{\partial\theta_6}-\frac{\partial M_{56}}{\partial\theta_1}\right)\dot{\theta}_5+\left(\frac{\partial M_{16}}{\partial\theta_6}+\frac{\partial M_{16}}{\partial\theta_6}-\frac{\partial M_{66}}{\partial\theta_1}\right)\dot{\theta}_6\right]$$

$$C_{21}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{21}}{\partial\theta_1}+\frac{\partial M_{21}}{\partial\theta_1}-\frac{\partial M_{11}}{\partial\theta_2}\right)\dot{\theta}_1+\left(\frac{\partial M_{21}}{\partial\theta_2}+\frac{\partial M_{22}}{\partial\theta_1}-\frac{\partial M_{21}}{\partial\theta_2}\right)\dot{\theta}_2+\left(\frac{\partial M_{21}}{\partial\theta_3}+\frac{\partial M_{23}}{\partial\theta_1}-\frac{\partial M_{31}}{\partial\theta_2}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{21}}{\partial\theta_4}+\frac{\partial M_{24}}{\partial\theta_1}-\frac{\partial M_{41}}{\partial\theta_2}\right)\dot{\theta}_4+\left(\frac{\partial M_{21}}{\partial\theta_5}+\frac{\partial M_{25}}{\partial\theta_1}-\frac{\partial M_{51}}{\partial\theta_2}\right)\dot{\theta}_5+\left(\frac{\partial M_{21}}{\partial\theta_6}+\frac{\partial M_{26}}{\partial\theta_1}-\frac{\partial M_{61}}{\partial\theta_2}\right)\dot{\theta}_6\right]$$

$$C_{22}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{22}}{\partial\theta_1}+\frac{\partial M_{21}}{\partial\theta_2}-\frac{\partial M_{12}}{\partial\theta_2}\right)\dot{\theta}_1+\left(\frac{\partial M_{22}}{\partial\theta_2}+\frac{\partial M_{22}}{\partial\theta_2}-\frac{\partial M_{22}}{\partial\theta_2}\right)\dot{\theta}_2+\left(\frac{\partial M_{22}}{\partial\theta_3}+\frac{\partial M_{23}}{\partial\theta_2}-\frac{\partial M_{32}}{\partial\theta_2}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{22}}{\partial\theta_4}+\frac{\partial M_{24}}{\partial\theta_2}-\frac{\partial M_{42}}{\partial\theta_2}\right)\dot{\theta}_4+\left(\frac{\partial M_{22}}{\partial\theta_5}+\frac{\partial M_{25}}{\partial\theta_2}-\frac{\partial M_{52}}{\partial\theta_2}\right)\dot{\theta}_5+\left(\frac{\partial M_{22}}{\partial\theta_6}+\frac{\partial M_{26}}{\partial\theta_2}-\frac{\partial M_{62}}{\partial\theta_2}\right)\dot{\theta}_6\right]$$

$$C_{23}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{23}}{\partial\theta_1}+\frac{\partial M_{21}}{\partial\theta_3}-\frac{\partial M_{13}}{\partial\theta_2}\right)\dot{\theta}_1+\left(\frac{\partial M_{23}}{\partial\theta_2}+\frac{\partial M_{22}}{\partial\theta_3}-\frac{\partial M_{23}}{\partial\theta_2}\right)\dot{\theta}_2+\left(\frac{\partial M_{23}}{\partial\theta_3}+\frac{\partial M_{23}}{\partial\theta_3}-\frac{\partial M_{33}}{\partial\theta_2}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{23}}{\partial\theta_4}+\frac{\partial M_{24}}{\partial\theta_3}-\frac{\partial M_{43}}{\partial\theta_2}\right)\dot{\theta}_4+\left(\frac{\partial M_{23}}{\partial\theta_5}+\frac{\partial M_{25}}{\partial\theta_3}-\frac{\partial M_{53}}{\partial\theta_2}\right)\dot{\theta}_5+\left(\frac{\partial M_{23}}{\partial\theta_6}+\frac{\partial M_{26}}{\partial\theta_3}-\frac{\partial M_{63}}{\partial\theta_2}\right)\dot{\theta}_6\right]$$

$$C_{24}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{24}}{\partial\theta_1}+\frac{\partial M_{21}}{\partial\theta_4}-\frac{\partial M_{14}}{\partial\theta_2}\right)\dot{\theta}_1+\left(\frac{\partial M_{24}}{\partial\theta_2}+\frac{\partial M_{22}}{\partial\theta_4}-\frac{\partial M_{24}}{\partial\theta_2}\right)\dot{\theta}_2+\left(\frac{\partial M_{24}}{\partial\theta_3}+\frac{\partial M_{23}}{\partial\theta_4}-\frac{\partial M_{34}}{\partial\theta_2}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{24}}{\partial\theta_4}+\frac{\partial M_{24}}{\partial\theta_4}-\frac{\partial M_{44}}{\partial\theta_2}\right)\dot{\theta}_4+\left(\frac{\partial M_{24}}{\partial\theta_5}+\frac{\partial M_{25}}{\partial\theta_4}-\frac{\partial M_{54}}{\partial\theta_2}\right)\dot{\theta}_5+\left(\frac{\partial M_{24}}{\partial\theta_6}+\frac{\partial M_{26}}{\partial\theta_4}-\frac{\partial M_{64}}{\partial\theta_2}\right)\dot{\theta}_6\right]$$

$$C_{25}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{25}}{\partial\theta_1}+\frac{\partial M_{21}}{\partial\theta_5}-\frac{\partial M_{15}}{\partial\theta_2}\right)\dot{\theta}_1+\left(\frac{\partial M_{25}}{\partial\theta_2}+\frac{\partial M_{22}}{\partial\theta_5}-\frac{\partial M_{25}}{\partial\theta_2}\right)\dot{\theta}_2+\left(\frac{\partial M_{25}}{\partial\theta_3}+\frac{\partial M_{23}}{\partial\theta_5}-\frac{\partial M_{35}}{\partial\theta_2}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{25}}{\partial\theta_4}+\frac{\partial M_{24}}{\partial\theta_5}-\frac{\partial M_{45}}{\partial\theta_2}\right)\dot{\theta}_4+\left(\frac{\partial M_{25}}{\partial\theta_5}+\frac{\partial M_{25}}{\partial\theta_5}-\frac{\partial M_{55}}{\partial\theta_2}\right)\dot{\theta}_5+\left(\frac{\partial M_{25}}{\partial\theta_6}+\frac{\partial M_{26}}{\partial\theta_5}-\frac{\partial M_{65}}{\partial\theta_2}\right)\dot{\theta}_6\right]$$

$$C_{26}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{26}}{\partial\theta_1}+\frac{\partial M_{21}}{\partial\theta_6}-\frac{\partial M_{16}}{\partial\theta_2}\right)\dot{\theta}_1+\left(\frac{\partial M_{26}}{\partial\theta_2}+\frac{\partial M_{22}}{\partial\theta_6}-\frac{\partial M_{26}}{\partial\theta_2}\right)\dot{\theta}_2+\left(\frac{\partial M_{26}}{\partial\theta_3}+\frac{\partial M_{23}}{\partial\theta_6}-\frac{\partial M_{36}}{\partial\theta_2}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{26}}{\partial\theta_4}+\frac{\partial M_{24}}{\partial\theta_6}-\frac{\partial M_{46}}{\partial\theta_2}\right)\dot{\theta}_4+\left(\frac{\partial M_{26}}{\partial\theta_5}+\frac{\partial M_{25}}{\partial\theta_6}-\frac{\partial M_{56}}{\partial\theta_2}\right)\dot{\theta}_5+\left(\frac{\partial M_{26}}{\partial\theta_6}+\frac{\partial M_{26}}{\partial\theta_6}-\frac{\partial M_{66}}{\partial\theta_2}\right)\dot{\theta}_6\right]$$

$$C_{31}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{31}}{\partial\theta_1}+\frac{\partial M_{31}}{\partial\theta_1}-\frac{\partial M_{11}}{\partial\theta_3}\right)\dot{\theta}_1+\left(\frac{\partial M_{31}}{\partial\theta_2}+\frac{\partial M_{32}}{\partial\theta_1}-\frac{\partial M_{21}}{\partial\theta_3}\right)\dot{\theta}_2+\left(\frac{\partial M_{31}}{\partial\theta_3}+\frac{\partial M_{33}}{\partial\theta_1}-\frac{\partial M_{31}}{\partial\theta_3}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{31}}{\partial\theta_4}+\frac{\partial M_{34}}{\partial\theta_1}-\frac{\partial M_{41}}{\partial\theta_3}\right)\dot{\theta}_4+\left(\frac{\partial M_{31}}{\partial\theta_5}+\frac{\partial M_{35}}{\partial\theta_1}-\frac{\partial M_{51}}{\partial\theta_3}\right)\dot{\theta}_5+\left(\frac{\partial M_{31}}{\partial\theta_6}+\frac{\partial M_{36}}{\partial\theta_1}-\frac{\partial M_{61}}{\partial\theta_3}\right)\dot{\theta}_6\right]$$

$$C_{32}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{32}}{\partial\theta_1}+\frac{\partial M_{31}}{\partial\theta_2}-\frac{\partial M_{12}}{\partial\theta_3}\right)\dot{\theta}_1+\left(\frac{\partial M_{32}}{\partial\theta_2}+\frac{\partial M_{32}}{\partial\theta_2}-\frac{\partial M_{22}}{\partial\theta_3}\right)\dot{\theta}_2+\left(\frac{\partial M_{32}}{\partial\theta_3}+\frac{\partial M_{33}}{\partial\theta_2}-\frac{\partial M_{32}}{\partial\theta_3}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{32}}{\partial\theta_4}+\frac{\partial M_{34}}{\partial\theta_2}-\frac{\partial M_{42}}{\partial\theta_3}\right)\dot{\theta}_4+\left(\frac{\partial M_{32}}{\partial\theta_5}+\frac{\partial M_{35}}{\partial\theta_2}-\frac{\partial M_{52}}{\partial\theta_3}\right)\dot{\theta}_5+\left(\frac{\partial M_{32}}{\partial\theta_6}+\frac{\partial M_{36}}{\partial\theta_2}-\frac{\partial M_{62}}{\partial\theta_3}\right)\dot{\theta}_6\right]$$

$$C_{33}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{33}}{\partial\theta_1}+\frac{\partial M_{31}}{\partial\theta_3}-\frac{\partial M_{13}}{\partial\theta_3}\right)\dot{\theta}_1+\left(\frac{\partial M_{33}}{\partial\theta_2}+\frac{\partial M_{32}}{\partial\theta_3}-\frac{\partial M_{23}}{\partial\theta_3}\right)\dot{\theta}_2+\left(\frac{\partial M_{33}}{\partial\theta_3}+\frac{\partial M_{33}}{\partial\theta_3}-\frac{\partial M_{33}}{\partial\theta_3}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{33}}{\partial\theta_4}+\frac{\partial M_{34}}{\partial\theta_3}-\frac{\partial M_{43}}{\partial\theta_3}\right)\dot{\theta}_4+\left(\frac{\partial M_{33}}{\partial\theta_5}+\frac{\partial M_{35}}{\partial\theta_3}-\frac{\partial M_{53}}{\partial\theta_3}\right)\dot{\theta}_5+\left(\frac{\partial M_{33}}{\partial\theta_6}+\frac{\partial M_{36}}{\partial\theta_3}-\frac{\partial M_{63}}{\partial\theta_3}\right)\dot{\theta}_3\right]$$

$$C_{34}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{34}}{\partial\theta_1}+\frac{\partial M_{31}}{\partial\theta_4}-\frac{\partial M_{14}}{\partial\theta_3}\right)\dot{\theta}_1+\left(\frac{\partial M_{34}}{\partial\theta_2}+\frac{\partial M_{32}}{\partial\theta_4}-\frac{\partial M_{24}}{\partial\theta_3}\right)\dot{\theta}_2+\left(\frac{\partial M_{34}}{\partial\theta_3}+\frac{\partial M_{33}}{\partial\theta_4}-\frac{\partial M_{34}}{\partial\theta_3}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{34}}{\partial\theta_4}+\frac{\partial M_{34}}{\partial\theta_4}-\frac{\partial M_{44}}{\partial\theta_3}\right)\dot{\theta}_4+\left(\frac{\partial M_{34}}{\partial\theta_5}+\frac{\partial M_{35}}{\partial\theta_4}-\frac{\partial M_{54}}{\partial\theta_3}\right)\dot{\theta}_5+\left(\frac{\partial M_{34}}{\partial\theta_6}+\frac{\partial M_{36}}{\partial\theta_4}-\frac{\partial M_{64}}{\partial\theta_3}\right)\dot{\theta}_6\right]$$

$$C_{35}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{35}}{\partial\theta_1}+\frac{\partial M_{31}}{\partial\theta_5}-\frac{\partial M_{15}}{\partial\theta_3}\right)\dot{\theta}_1+\left(\frac{\partial M_{35}}{\partial\theta_2}+\frac{\partial M_{32}}{\partial\theta_5}-\frac{\partial M_{25}}{\partial\theta_3}\right)\dot{\theta}_2+\left(\frac{\partial M_{35}}{\partial\theta_3}+\frac{\partial M_{33}}{\partial\theta_5}-\frac{\partial M_{35}}{\partial\theta_3}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{35}}{\partial\theta_4}+\frac{\partial M_{34}}{\partial\theta_5}-\frac{\partial M_{45}}{\partial\theta_3}\right)\dot{\theta}_4+\left(\frac{\partial M_{35}}{\partial\theta_5}+\frac{\partial M_{35}}{\partial\theta_5}-\frac{\partial M_{55}}{\partial\theta_3}\right)\dot{\theta}_5+\left(\frac{\partial M_{35}}{\partial\theta_6}+\frac{\partial M_{36}}{\partial\theta_5}-\frac{\partial M_{65}}{\partial\theta_3}\right)\dot{\theta}_6\right]$$

$$C_{36}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{36}}{\partial\theta_1}+\frac{\partial M_{31}}{\partial\theta_6}-\frac{\partial M_{16}}{\partial\theta_3}\right)\dot{\theta}_1+\left(\frac{\partial M_{36}}{\partial\theta_2}+\frac{\partial M_{32}}{\partial\theta_6}-\frac{\partial M_{26}}{\partial\theta_3}\right)\dot{\theta}_2+\left(\frac{\partial M_{36}}{\partial\theta_3}+\frac{\partial M_{33}}{\partial\theta_6}-\frac{\partial M_{36}}{\partial\theta_3}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{36}}{\partial\theta_4}+\frac{\partial M_{34}}{\partial\theta_6}-\frac{\partial M_{46}}{\partial\theta_3}\right)\dot{\theta}_4+\left(\frac{\partial M_{36}}{\partial\theta_5}+\frac{\partial M_{35}}{\partial\theta_6}-\frac{\partial M_{56}}{\partial\theta_3}\right)\dot{\theta}_5+\left(\frac{\partial M_{36}}{\partial\theta_6}+\frac{\partial M_{36}}{\partial\theta_6}-\frac{\partial M_{66}}{\partial\theta_3}\right)\dot{\theta}_6\right]$$

$$C_{41}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{41}}{\partial\theta_1}+\frac{\partial M_{41}}{\partial\theta_1}-\frac{\partial M_{11}}{\partial\theta_4}\right)\dot{\theta}_1+\left(\frac{\partial M_{41}}{\partial\theta_2}+\frac{\partial M_{42}}{\partial\theta_1}-\frac{\partial M_{21}}{\partial\theta_4}\right)\dot{\theta}_2+\left(\frac{\partial M_{41}}{\partial\theta_3}+\frac{\partial M_{43}}{\partial\theta_1}-\frac{\partial M_{31}}{\partial\theta_4}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{41}}{\partial\theta_4}+\frac{\partial M_{44}}{\partial\theta_1}-\frac{\partial M_{41}}{\partial\theta_4}\right)\dot{\theta}_4+\left(\frac{\partial M_{41}}{\partial\theta_5}+\frac{\partial M_{45}}{\partial\theta_1}-\frac{\partial M_{51}}{\partial\theta_4}\right)\dot{\theta}_5+\left(\frac{\partial M_{41}}{\partial\theta_6}+\frac{\partial M_{46}}{\partial\theta_1}-\frac{\partial M_{61}}{\partial\theta_4}\right)\dot{\theta}_6\right]$$

$$C_{42}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{42}}{\partial\theta_1}+\frac{\partial M_{41}}{\partial\theta_2}-\frac{\partial M_{12}}{\partial\theta_4}\right)\dot{\theta}_1+\left(\frac{\partial M_{42}}{\partial\theta_2}+\frac{\partial M_{42}}{\partial\theta_2}-\frac{\partial M_{22}}{\partial\theta_4}\right)\dot{\theta}_2+\left(\frac{\partial M_{42}}{\partial\theta_3}+\frac{\partial M_{43}}{\partial\theta_2}-\frac{\partial M_{32}}{\partial\theta_4}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{42}}{\partial\theta_4}+\frac{\partial M_{44}}{\partial\theta_2}-\frac{\partial M_{42}}{\partial\theta_4}\right)\dot{\theta}_4+\left(\frac{\partial M_{42}}{\partial\theta_5}+\frac{\partial M_{45}}{\partial\theta_2}-\frac{\partial M_{52}}{\partial\theta_4}\right)\dot{\theta}_5+\left(\frac{\partial M_{42}}{\partial\theta_6}+\frac{\partial M_{46}}{\partial\theta_2}-\frac{\partial M_{62}}{\partial\theta_4}\right)\dot{\theta}_6\right]$$

$$C_{43}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{43}}{\partial\theta_1}+\frac{\partial M_{41}}{\partial\theta_3}-\frac{\partial M_{13}}{\partial\theta_4}\right)\dot{\theta}_1+\left(\frac{\partial M_{43}}{\partial\theta_2}+\frac{\partial M_{42}}{\partial\theta_3}-\frac{\partial M_{23}}{\partial\theta_4}\right)\dot{\theta}_2+\left(\frac{\partial M_{43}}{\partial\theta_3}+\frac{\partial M_{43}}{\partial\theta_3}-\frac{\partial M_{33}}{\partial\theta_4}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{43}}{\partial\theta_4}+\frac{\partial M_{44}}{\partial\theta_3}-\frac{\partial M_{43}}{\partial\theta_4}\right)\dot{\theta}_4+\left(\frac{\partial M_{43}}{\partial\theta_5}+\frac{\partial M_{45}}{\partial\theta_3}-\frac{\partial M_{53}}{\partial\theta_4}\right)\dot{\theta}_5+\left(\frac{\partial M_{43}}{\partial\theta_6}+\frac{\partial M_{46}}{\partial\theta_3}-\frac{\partial M_{63}}{\partial\theta_4}\right)\dot{\theta}_6\right]$$

$$C_{44}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{44}}{\partial\theta_1}+\frac{\partial M_{41}}{\partial\theta_4}-\frac{\partial M_{14}}{\partial\theta_4}\right)\dot{\theta}_1+\left(\frac{\partial M_{44}}{\partial\theta_2}+\frac{\partial M_{42}}{\partial\theta_4}-\frac{\partial M_{24}}{\partial\theta_4}\right)\dot{\theta}_2+\left(\frac{\partial M_{44}}{\partial\theta_3}+\frac{\partial M_{43}}{\partial\theta_4}-\frac{\partial M_{34}}{\partial\theta_4}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{44}}{\partial\theta_4}+\frac{\partial M_{44}}{\partial\theta_4}-\frac{\partial M_{44}}{\partial\theta_4}\right)\dot{\theta}_4+\left(\frac{\partial M_{44}}{\partial\theta_5}+\frac{\partial M_{45}}{\partial\theta_4}-\frac{\partial M_{54}}{\partial\theta_4}\right)\dot{\theta}_5+\left(\frac{\partial M_{44}}{\partial\theta_6}+\frac{\partial M_{46}}{\partial\theta_4}-\frac{\partial M_{64}}{\partial\theta_4}\right)\dot{\theta}_6\right]$$

$$C_{45}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{45}}{\partial\theta_1}+\frac{\partial M_{41}}{\partial\theta_5}-\frac{\partial M_{15}}{\partial\theta_4}\right)\dot{\theta}_1+\left(\frac{\partial M_{45}}{\partial\theta_2}+\frac{\partial M_{42}}{\partial\theta_5}-\frac{\partial M_{25}}{\partial\theta_4}\right)\dot{\theta}_2+\left(\frac{\partial M_{45}}{\partial\theta_3}+\frac{\partial M_{43}}{\partial\theta_5}-\frac{\partial M_{35}}{\partial\theta_4}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{45}}{\partial\theta_4}+\frac{\partial M_{44}}{\partial\theta_5}-\frac{\partial M_{45}}{\partial\theta_4}\right)\dot{\theta}_4+\left(\frac{\partial M_{45}}{\partial\theta_5}+\frac{\partial M_{45}}{\partial\theta_5}-\frac{\partial M_{55}}{\partial\theta_4}\right)\dot{\theta}_5+\left(\frac{\partial M_{45}}{\partial\theta_6}+\frac{\partial M_{46}}{\partial\theta_5}-\frac{\partial M_{65}}{\partial\theta_4}\right)\dot{\theta}_6\right]$$

$$C_{46}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{46}}{\partial\theta_1}+\frac{\partial M_{41}}{\partial\theta_6}-\frac{\partial M_{16}}{\partial\theta_4}\right)\dot{\theta}_1+\left(\frac{\partial M_{46}}{\partial\theta_2}+\frac{\partial M_{42}}{\partial\theta_6}-\frac{\partial M_{26}}{\partial\theta_4}\right)\dot{\theta}_2+\left(\frac{\partial M_{46}}{\partial\theta_3}+\frac{\partial M_{43}}{\partial\theta_6}-\frac{\partial M_{36}}{\partial\theta_4}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{46}}{\partial\theta_4}+\frac{\partial M_{44}}{\partial\theta_6}-\frac{\partial M_{46}}{\partial\theta_4}\right)\dot{\theta}_4+\left(\frac{\partial M_{46}}{\partial\theta_5}+\frac{\partial M_{45}}{\partial\theta_6}-\frac{\partial M_{56}}{\partial\theta_4}\right)\dot{\theta}_5+\left(\frac{\partial M_{46}}{\partial\theta_6}+\frac{\partial M_{46}}{\partial\theta_6}-\frac{\partial M_{66}}{\partial\theta_4}\right)\dot{\theta}_6\right]$$

$$C_{51}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{51}}{\partial\theta_1}+\frac{\partial M_{51}}{\partial\theta_1}-\frac{\partial M_{11}}{\partial\theta_5}\right)\dot{\theta}_1+\left(\frac{\partial M_{51}}{\partial\theta_2}+\frac{\partial M_{52}}{\partial\theta_1}-\frac{\partial M_{21}}{\partial\theta_5}\right)\dot{\theta}_2+\left(\frac{\partial M_{51}}{\partial\theta_3}+\frac{\partial M_{53}}{\partial\theta_1}-\frac{\partial M_{31}}{\partial\theta_5}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{51}}{\partial\theta_4}+\frac{\partial M_{54}}{\partial\theta_1}-\frac{\partial M_{41}}{\partial\theta_5}\right)\dot{\theta}_4+\left(\frac{\partial M_{51}}{\partial\theta_5}+\frac{\partial M_{55}}{\partial\theta_1}-\frac{\partial M_{51}}{\partial\theta_5}\right)\dot{\theta}_5+\left(\frac{\partial M_{51}}{\partial\theta_6}+\frac{\partial M_{56}}{\partial\theta_1}-\frac{\partial M_{61}}{\partial\theta_5}\right)\dot{\theta}_6\right]$$

$$C_{52}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{52}}{\partial\theta_1}+\frac{\partial M_{51}}{\partial\theta_2}-\frac{\partial M_{12}}{\partial\theta_5}\right)\dot{\theta}_1+\left(\frac{\partial M_{52}}{\partial\theta_2}+\frac{\partial M_{52}}{\partial\theta_2}-\frac{\partial M_{22}}{\partial\theta_5}\right)\dot{\theta}_2+\left(\frac{\partial M_{52}}{\partial\theta_3}+\frac{\partial M_{53}}{\partial\theta_2}-\frac{\partial M_{32}}{\partial\theta_5}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{52}}{\partial\theta_4}+\frac{\partial M_{54}}{\partial\theta_2}-\frac{\partial M_{42}}{\partial\theta_5}\right)\dot{\theta}_4+\left(\frac{\partial M_{52}}{\partial\theta_5}+\frac{\partial M_{55}}{\partial\theta_2}-\frac{\partial M_{52}}{\partial\theta_5}\right)\dot{\theta}_5+\left(\frac{\partial M_{52}}{\partial\theta_6}+\frac{\partial M_{56}}{\partial\theta_2}-\frac{\partial M_{62}}{\partial\theta_5}\right)\dot{\theta}_6\right]$$

$$C_{53}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{53}}{\partial\theta_1}+\frac{\partial M_{51}}{\partial\theta_3}-\frac{\partial M_{13}}{\partial\theta_5}\right)\dot{\theta}_1+\left(\frac{\partial M_{53}}{\partial\theta_2}+\frac{\partial M_{52}}{\partial\theta_3}-\frac{\partial M_{23}}{\partial\theta_5}\right)\dot{\theta}_2+\left(\frac{\partial M_{53}}{\partial\theta_3}+\frac{\partial M_{53}}{\partial\theta_3}-\frac{\partial M_{33}}{\partial\theta_5}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{53}}{\partial\theta_4}+\frac{\partial M_{54}}{\partial\theta_3}-\frac{\partial M_{43}}{\partial\theta_5}\right)\dot{\theta}_4+\left(\frac{\partial M_{53}}{\partial\theta_5}+\frac{\partial M_{55}}{\partial\theta_3}-\frac{\partial M_{53}}{\partial\theta_5}\right)\dot{\theta}_5+\left(\frac{\partial M_{53}}{\partial\theta_6}+\frac{\partial M_{56}}{\partial\theta_3}-\frac{\partial M_{63}}{\partial\theta_5}\right)\dot{\theta}_6\right]$$

$$C_{54}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{54}}{\partial\theta_1}+\frac{\partial M_{51}}{\partial\theta_4}-\frac{\partial M_{14}}{\partial\theta_5}\right)\dot{\theta}_1+\left(\frac{\partial M_{54}}{\partial\theta_2}+\frac{\partial M_{52}}{\partial\theta_4}-\frac{\partial M_{24}}{\partial\theta_5}\right)\dot{\theta}_2+\left(\frac{\partial M_{54}}{\partial\theta_3}+\frac{\partial M_{53}}{\partial\theta_4}-\frac{\partial M_{34}}{\partial\theta_5}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{54}}{\partial\theta_4}+\frac{\partial M_{54}}{\partial\theta_4}-\frac{\partial M_{44}}{\partial\theta_5}\right)\dot{\theta}_4+\left(\frac{\partial M_{54}}{\partial\theta_5}+\frac{\partial M_{55}}{\partial\theta_4}-\frac{\partial M_{54}}{\partial\theta_5}\right)\dot{\theta}_5+\left(\frac{\partial M_{54}}{\partial\theta_6}+\frac{\partial M_{56}}{\partial\theta_4}-\frac{\partial M_{64}}{\partial\theta_5}\right)\dot{\theta}_6\right]$$

$$C_{55}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{55}}{\partial\theta_1}+\frac{\partial M_{51}}{\partial\theta_5}-\frac{\partial M_{15}}{\partial\theta_5}\right)\dot{\theta}_1+\left(\frac{\partial M_{55}}{\partial\theta_2}+\frac{\partial M_{52}}{\partial\theta_5}-\frac{\partial M_{25}}{\partial\theta_5}\right)\dot{\theta}_2+\left(\frac{\partial M_{55}}{\partial\theta_3}+\frac{\partial M_{52}}{\partial\theta_5}-\frac{\partial M_{25}}{\partial\theta_5}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{55}}{\partial\theta_4}+\frac{\partial M_{54}}{\partial\theta_5}-\frac{\partial M_{45}}{\partial\theta_5}\right)\dot{\theta}_4+\left(\frac{\partial M_{55}}{\partial\theta_5}+\frac{\partial M_{55}}{\partial\theta_5}-\frac{\partial M_{55}}{\partial\theta_5}\right)\dot{\theta}_5+\left(\frac{\partial M_{55}}{\partial\theta_6}+\frac{\partial M_{56}}{\partial\theta_5}-\frac{\partial M_{65}}{\partial\theta_5}\right)\dot{\theta}_6\right]$$

$$C_{56}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{56}}{\partial\theta_1}+\frac{\partial M_{51}}{\partial\theta_6}-\frac{\partial M_{16}}{\partial\theta_5}\right)\dot{\theta}_1+\left(\frac{\partial M_{56}}{\partial\theta_2}+\frac{\partial M_{52}}{\partial\theta_6}-\frac{\partial M_{26}}{\partial\theta_5}\right)\dot{\theta}_2+\left(\frac{\partial M_{56}}{\partial\theta_3}+\frac{\partial M_{53}}{\partial\theta_6}-\frac{\partial M_{36}}{\partial\theta_5}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{56}}{\partial\theta_4}+\frac{\partial M_{54}}{\partial\theta_6}-\frac{\partial M_{46}}{\partial\theta_5}\right)\dot{\theta}_4+\left(\frac{\partial M_{56}}{\partial\theta_5}+\frac{\partial M_{55}}{\partial\theta_6}-\frac{\partial M_{56}}{\partial\theta_5}\right)\dot{\theta}_5+\left(\frac{\partial M_{56}}{\partial\theta_6}+\frac{\partial M_{56}}{\partial\theta_6}-\frac{\partial M_{66}}{\partial\theta_5}\right)\dot{\theta}_6\right]$$

$$C_{61}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{61}}{\partial\theta_1}+\frac{\partial M_{61}}{\partial\theta_1}-\frac{\partial M_{11}}{\partial\theta_6}\right)\dot{\theta}_1+\left(\frac{\partial M_{61}}{\partial\theta_2}+\frac{\partial M_{62}}{\partial\theta_1}-\frac{\partial M_{21}}{\partial\theta_6}\right)\dot{\theta}_2+\left(\frac{\partial M_{61}}{\partial\theta_3}+\frac{\partial M_{63}}{\partial\theta_1}-\frac{\partial M_{31}}{\partial\theta_6}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{61}}{\partial\theta_4}+\frac{\partial M_{64}}{\partial\theta_1}-\frac{\partial M_{41}}{\partial\theta_6}\right)\dot{\theta}_4+\left(\frac{\partial M_{61}}{\partial\theta_5}+\frac{\partial M_{65}}{\partial\theta_1}-\frac{\partial M_{51}}{\partial\theta_6}\right)\dot{\theta}_5+\left(\frac{\partial M_{61}}{\partial\theta_6}+\frac{\partial M_{66}}{\partial\theta_1}-\frac{\partial M_{61}}{\partial\theta_6}\right)\dot{\theta}_6\right]$$

$$C_{62}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{62}}{\partial\theta_1}+\frac{\partial M_{61}}{\partial\theta_2}-\frac{\partial M_{12}}{\partial\theta_6}\right)\dot{\theta}_1+\left(\frac{\partial M_{62}}{\partial\theta_2}+\frac{\partial M_{62}}{\partial\theta_2}-\frac{\partial M_{22}}{\partial\theta_6}\right)\dot{\theta}_2+\left(\frac{\partial M_{62}}{\partial\theta_3}+\frac{\partial M_{63}}{\partial\theta_2}-\frac{\partial M_{32}}{\partial\theta_6}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{62}}{\partial\theta_4}+\frac{\partial M_{64}}{\partial\theta_2}-\frac{\partial M_{42}}{\partial\theta_6}\right)\dot{\theta}_4+\left(\frac{\partial M_{62}}{\partial\theta_5}+\frac{\partial M_{65}}{\partial\theta_2}-\frac{\partial M_{52}}{\partial\theta_6}\right)\dot{\theta}_5+\left(\frac{\partial M_{62}}{\partial\theta_6}+\frac{\partial M_{66}}{\partial\theta_2}-\frac{\partial M_{62}}{\partial\theta_6}\right)\dot{\theta}_6\right]$$

$$C_{63}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{63}}{\partial\theta_1}+\frac{\partial M_{61}}{\partial\theta_3}-\frac{\partial M_{13}}{\partial\theta_6}\right)\dot{\theta}_1+\left(\frac{\partial M_{63}}{\partial\theta_2}+\frac{\partial M_{62}}{\partial\theta_3}-\frac{\partial M_{23}}{\partial\theta_6}\right)\dot{\theta}_2+\left(\frac{\partial M_{63}}{\partial\theta_3}+\frac{\partial M_{63}}{\partial\theta_3}-\frac{\partial M_{33}}{\partial\theta_6}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{63}}{\partial\theta_4}+\frac{\partial M_{64}}{\partial\theta_3}-\frac{\partial M_{43}}{\partial\theta_6}\right)\dot{\theta}_4+\left(\frac{\partial M_{63}}{\partial\theta_5}+\frac{\partial M_{65}}{\partial\theta_3}-\frac{\partial M_{53}}{\partial\theta_6}\right)\dot{\theta}_5+\left(\frac{\partial M_{63}}{\partial\theta_6}+\frac{\partial M_{66}}{\partial\theta_3}-\frac{\partial M_{63}}{\partial\theta_6}\right)\dot{\theta}_6\right]$$

$$C_{64}(\theta)=\frac{1}{2}\left[\left(\frac{\partial M_{64}}{\partial\theta_1}+\frac{\partial M_{61}}{\partial\theta_4}-\frac{\partial M_{14}}{\partial\theta_6}\right)\dot{\theta}_1+\left(\frac{\partial M_{64}}{\partial\theta_2}+\frac{\partial M_{62}}{\partial\theta_4}-\frac{\partial M_{24}}{\partial\theta_6}\right)\dot{\theta}_2+\left(\frac{\partial M_{64}}{\partial\theta_3}+\frac{\partial M_{63}}{\partial\theta_4}-\frac{\partial M_{34}}{\partial\theta_6}\right)\dot{\theta}_3\right.$$
$$\left.+\left(\frac{\partial M_{64}}{\partial\theta_4}+\frac{\partial M_{64}}{\partial\theta_4}-\frac{\partial M_{44}}{\partial\theta_6}\right)\dot{\theta}_4+\left(\frac{\partial M_{64}}{\partial\theta_5}+\frac{\partial M_{65}}{\partial\theta_4}-\frac{\partial M_{54}}{\partial\theta_6}\right)\dot{\theta}_5+\left(\frac{\partial M_{64}}{\partial\theta_6}+\frac{\partial M_{66}}{\partial\theta_4}-\frac{\partial M_{64}}{\partial\theta_6}\right)\dot{\theta}_6\right]$$

$$C_{65}(\theta) = \frac{1}{2}\left[\left(\frac{\partial M_{65}}{\partial \theta_1} + \frac{\partial M_{61}}{\partial \theta_5} - \frac{\partial M_{15}}{\partial \theta_6}\right)\dot{\theta}_1 + \left(\frac{\partial M_{65}}{\partial \theta_2} + \frac{\partial M_{62}}{\partial \theta_5} - \frac{\partial M_{25}}{\partial \theta_6}\right)\dot{\theta}_2 + \left(\frac{\partial M_{65}}{\partial \theta_3} + \frac{\partial M_{63}}{\partial \theta_5} - \frac{\partial M_{35}}{\partial \theta_6}\right)\dot{\theta}_3 + \left(\frac{\partial M_{65}}{\partial \theta_4} + \frac{\partial M_{64}}{\partial \theta_5} - \frac{\partial M_{45}}{\partial \theta_6}\right)\dot{\theta}_4 + \left(\frac{\partial M_{65}}{\partial \theta_5} + \frac{\partial M_{65}}{\partial \theta_5} - \frac{\partial M_{55}}{\partial \theta_6}\right)\dot{\theta}_5 + \left(\frac{\partial M_{65}}{\partial \theta_6} + \frac{\partial M_{66}}{\partial \theta_5} - \frac{\partial M_{65}}{\partial \theta_6}\right)\dot{\theta}_6\right]$$

$$C_{66}(\theta) = \frac{1}{2}\left[\left(\frac{\partial M_{66}}{\partial \theta_1} + \frac{\partial M_{61}}{\partial \theta_6} - \frac{\partial M_{16}}{\partial \theta_6}\right)\dot{\theta}_1 + \left(\frac{\partial M_{66}}{\partial \theta_2} + \frac{\partial M_{62}}{\partial \theta_6} - \frac{\partial M_{26}}{\partial \theta_6}\right)\dot{\theta}_2 + \left(\frac{\partial M_{66}}{\partial \theta_3} + \frac{\partial M_{63}}{\partial \theta_6} - \frac{\partial M_{36}}{\partial \theta_6}\right)\dot{\theta}_3 + \left(\frac{\partial M_{66}}{\partial \theta_4} + \frac{\partial M_{64}}{\partial \theta_6} - \frac{\partial M_{46}}{\partial \theta_6}\right)\dot{\theta}_4 + \left(\frac{\partial M_{66}}{\partial \theta_5} + \frac{\partial M_{65}}{\partial \theta_6} - \frac{\partial M_{56}}{\partial \theta_6}\right)\dot{\theta}_5 + \left(\frac{\partial M_{66}}{\partial \theta_6} + \frac{\partial M_{66}}{\partial \theta_6} - \frac{\partial M_{66}}{\partial \theta_6}\right)\dot{\theta}_6\right]$$

5. Simulations

This chapter deals with simulations of the dynamic model in open loop and closed loop. Note that the main goal is to perform simulations and comparative analyze of the model, not to optimize a control system for a specific job task.

The simulation structure in Simulink and the connection to Matlab is described in Section5.1. In Section 5.2 the second order model is reduced to an equivalent first order model which is needed to perform simulations in Simulink.

The open loop case is presented in Section 5.3. First the model is driven with desired torque to check for open loop stability, and then energy properties are investigated. The closed loop case in Section 5.5 presents a mathematical proof of global asymptotic stability with PD control of a system model in the form (2.19).

5.1. Simulation structure

For realization the dynamic model was used one of the Simulink tool so called *Level-2 Matlab S-Function.* This is a block with multiple input and output ports where input 1 is the state vector, input 2 is the applied torque vector, and the output is the vector of state derivatives. For each time step in the simulation the updated vector of state derivatives is computed from the new inputs. The contents of the *Level-2 Matlab S-Function* block is the dynamic model in reduced form (see Section 5.2).

For each time step in the simulation, the state vector is sent from Simulink to the Matlab interface through a *To Workspace* block. That makes it possible to present the results graphically, and use the states to compute kinetic and potential energy.

5.2. Reduced system order

As described in Section 3.1, the dynamic model can be written on matrix form as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = u \tag{5.1}$$

To simulate the system in Simulink it is necessary to express it in the first-order nonlinear form

$$\dot{x} = f(x,u) \tag{5.2}$$

where $\dot{x}$ is the state vector and u is the torque vector.

The first step is to rearrange the terms $\ddot{q}$ in (5.1) to get

$$\ddot{q} = M^{-1}(-C\dot{q} - g + u) \tag{5.3}$$

where it is assumed that the inertia matrix M is invertible. The inertia matrix is the main factor of the kinetic energy expression $\frac{1}{2}\dot{q}^T M(q)\dot{q}$. Positive definiteness of M is seen directly by the fact that the kinetic energy is always nonnegative, and is zero if and only if all the joint velocities are zero. Thus, M is invertible and Equation (5.3) is valid.

The second step is to reduce the system from 6 second-order equations to 12 first-order equations. Defining

$$x_1 = q_1 \qquad x_2 = \dot{x}_1 = \dot{q}_1$$
$$x_3 = q_2 \qquad x_4 = \dot{x}_3 = \dot{q}_2$$

$$x_{11} = q_6 \qquad\qquad x_{12} = \dot{x}_{11} = \dot{q}_6$$

the dynamic system can be expressed in the form (5.2) as

$$\dot{x}_1 = x_2 \tag{5.4}$$

$$\dot{x}_2 = f_2(x, u) \tag{5.5}$$

$$\dot{x}_3 = x_4 \tag{5.6}$$

$$\dot{x}_4 = f_4(x, u) \tag{5.7}$$

$$\dot{x}_5 = x_6 \tag{5.8}$$

$$\dot{x}_6 = f_6(x, u) \tag{5.9}$$

$$\dot{x}_7 = x_8 \tag{5.10}$$

$$\dot{x}_8 = f_8(x, u) \tag{5.11}$$

$$\dot{x}_9 = x_{10} \tag{5.12}$$

$$\dot{x}_{10} = f_{10}(x, u) \tag{5.13}$$

$$\dot{x}_{11} = x_{12} \tag{5.14}$$

$$\dot{x}_{12} = f_{12}(x, u) \tag{5.15}$$

Note that this first-order model is only how the dynamics are implemented in Simulink and Matlab. All figures and text for the rest of this chapter will refer to the original second-order system with q as the state vector.

5.3. Open loop with desired torque

In open loop there is no feedback from the system output. In other words, no information about the joint variables and its derivatives is available when computing the input torque. Figure 5.1 shows the open loop model in Simulink, where the block called IRB 140contains all the dynamics.

Due to the excitation of gravity on the links being dependent on the joint variables, it is quite intuitive that controlling the system in open loop is impossible. The behavior of the system can be studied by driving the system with the desired torque, that is the constant torque derived when substituting in the dynamic equations for the desired joint variables and derivatives. If $q_{des} = 0$, this control torque can be explained as the constant torque which is needed to keep the manipulator steady in the desired position.

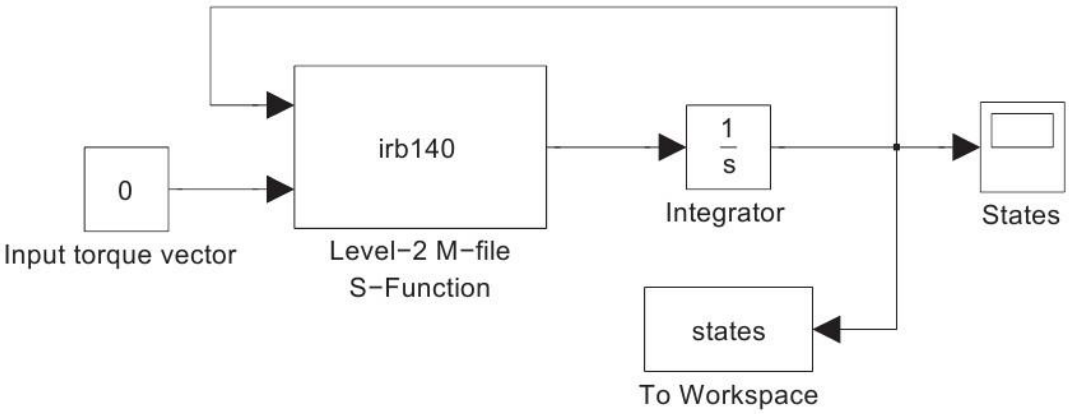


Figure 5.1 the open loop model

The desired position and velocity are set to

$$q_{des} = \begin{bmatrix} 0 & \frac{\pi}{2} & -\frac{\pi}{2} & 0 & 0 & 0 \end{bmatrix}^T \tag{5.16}$$

$$\dot{q}_{des} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{5.17}$$

which is the position when the manipulator arm is stretched out to the maximum in the direction. By substituting the desired position and velocity in the dynamic equations ($\dot{q}_{des} \rightarrow \ddot{q}_{des} = 0$), the control torque becomes

$$u_{des} = \begin{bmatrix} 0 \\ -2.409 \cdot g - 13.782 \cdot g \\ -2.409 \cdot g \\ 0 \\ -0.029 \cdot g \\ 0 \end{bmatrix} \tag{5.18}$$

From an intuitive perspective this control torque is as expected. To keep the manipulator steady in the chosen desired position, joint 2, 3 and 5 will have to be actuated to compensate for the gravity, based on the law of action and reaction. Joint 1, 4 and 6 will not be influenced by gravity as long as $\dot{q} = 0$, and is therefore given zero control torque.

Four simulations, each with different initial conditions, shows the behavior of the system when applied this control torque. The gravity acceleration is set to $g = 9.81 \frac{m}{s^2}$, the gravity of earth.

5.4. Closed loop position control

The open loop analysis with desired torque in Section 5.3 showed that controlling the system in open loop is impossible. This section deals with the attempt of controlling the system in closed loop. In closed loop, feedback controllers observe the output and calculate the error between this output and a reference. To achieve desired output, controllers can take one or more of three standard control elements that were described in Section 2.10.

Closed loop position control is also called the set-point tracking problem. The goal is to demonstrate that the manipulator can move from the position given as initial conditions, position A, to the position given as the reference value, position B. The joint torque input is continuously calculated by the feedback controllers. The path taken from A to B, as well as how long the motion lasts, is not controlled in the set-point tracking problem.

Section 5.4.1 presents a mathematical proof showing that a simple PD control structure works great for position control of systems in the general form (5.1). Then in Section 5.4.2, PD controllers are added to the model in Simulink, and simulations verify that the system is stable and that the position control is satisfying.

5.4.1. PD control with gravity compensation

It is a remarkable fact that the simple PD scheme for set-point control can be shown to work in the general case of a system model in the form of Equation (5.1). This can be proved in a Lyapunov stability analysis, as shown in [3]. This proof is of such importance and relevance to this thesis that it will be restated in this section.

The proof is based on independent joint control, which means that each joint is controlled as a single-input/single-output (SISO) system. Adding PD controllers in the model, the input torque u can be written in vector form as

$$u = -K_p\left(q_{ref} - q\right) - K_d\dot{q} = -K_p\tilde{q} - K_d\dot{q} \tag{5.19}$$

where q is the error between the joint references and the actual joint variables, and $K_p$and $K_d$are positive definite diagonal matrices of proportional and derivative gains.

It can be assumed that the gravitational acceleration is constant and known, such that $g(q)$ can be computed explicitly for all instants. By adding $g(q)$ to the input, gravity compensation is achieved such that the complete system model is now given by

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = u \tag{5.20}$$

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = -K_p\tilde{q} - K_d\dot{q} + g(q) \tag{5.21}$$

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} = -K_p\tilde{q} - K_d\dot{q} \tag{5.22}$$

To show that the input torque given in Equation (5.21) achieves asymptotic tracking, consider the Lyapunov function candidate

$$V = \frac{1}{2}\dot{q}^T M(q)\dot{q} + \frac{1}{2}\tilde{q}^T K_p\tilde{q} \tag{5.23}$$

For the manipulator, V represents the total energy that would result if the actuators were replaced by springs with stiffness constants represented by K_p, and with equilibrium position in $q = q_{ref}$. Thus, V is a positive function except in the equilibrium position $q = q_{ref}$with $\dot{q} = 0$, at which point V is zero. If it can be shown that V is decreasing along any motion, this implies that the robot is moving toward that equilibrium position.

Noting that q_{ref} is constant, the derivative of V is given by

$$\dot{V} = \dot{q}^T M(q)\ddot{q} + \frac{1}{2}\dot{q}^T\dot{M}(q)\dot{q} + \dot{q}^T K_p\tilde{q} \tag{5.24}$$

Solving for $M(q)\ddot{q}$in Equation (5.20) and substituting into the (5.24) yields

$$\dot{V} = \dot{q}^T\left(u - C(q,\dot{q})\dot{q} - g(q)\right) + \frac{1}{2}\dot{q}^T M(q)\dot{\dot{q}} + \dot{q}^T K_p\tilde{q} = \dot{q}^T\left(u - g(q) + K_p\tilde{q}\right) +$$
$$\frac{1}{2}\dot{q}^T\left[\dot{M}(q) - 2C(q,\dot{q})\right]\dot{q} = \dot{q}^T(u - g(q) + K_p\tilde{q}) \quad (5.25)$$

where $\dot{M}(q) - 2C(q,\dot{q}))$ is skew symmetric, then according to the subsection 2.2.2 it can be written as $\dot{q}^T\left[\dot{M}(q) - 2C(q,\dot{q})\right]\dot{q} = 0$. Substituting the input torque in Equation (5.21) for $u$in (5.25) above yields

$$\dot{V} = -\dot{q}^T K_d\dot{q} \leq 0 \quad (5.26)$$

The above analysis shows that V is decreasing as long as q is not zero.

Moreover it is necessary to prove that the manipulator cannot reach a position where $\dot{q} = 0$ but $q \neq q_{ref}$- Suppose $\dot{V} \equiv 0$, meaning that V is zero for all instants. Since K_d is a positive definite, this implies that $\dot{q} \equiv 0$and hence $\ddot{q} \equiv 0$. Substituting this in the system model (5.22), the result becomes

$$0 = -K_p\tilde{q} \quad (5.27)$$

which implies that $\tilde{q} = 0$. Finally, La Salle's theorem then proves that the equilibrium position $q = q_{ref}$ is globally asymptotic stable.

It should be noted that if the gravitational terms $g(q)$ are unknown, they cannot be added to the input because then the input cannot be computed. Controlling the system would then require controllers with robust and adaptive properties.

5.4.1. Simulations with PD control

The goal of this section is to perform simulations of the system with PD controllers, checking for asymptotic stability. If this can be accomplished, the mathematical proof in Section 5.4.1 is verified for the model. Figure 5.2 shows the Simulink model of the system in closed loop. With gravity compensation the model becomes

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} = -K_p\tilde{q} - K_d\dot{q} \quad (5.28)$$

where the input is

$$u = -K_p\tilde{q} - K_d\dot{q} \quad (5.29)$$

Note that to increase the efficiency of the simulations, it is chosen to remove the gravitational terms directly in the model (in the *IRB 140* block) instead of adding it to the input.

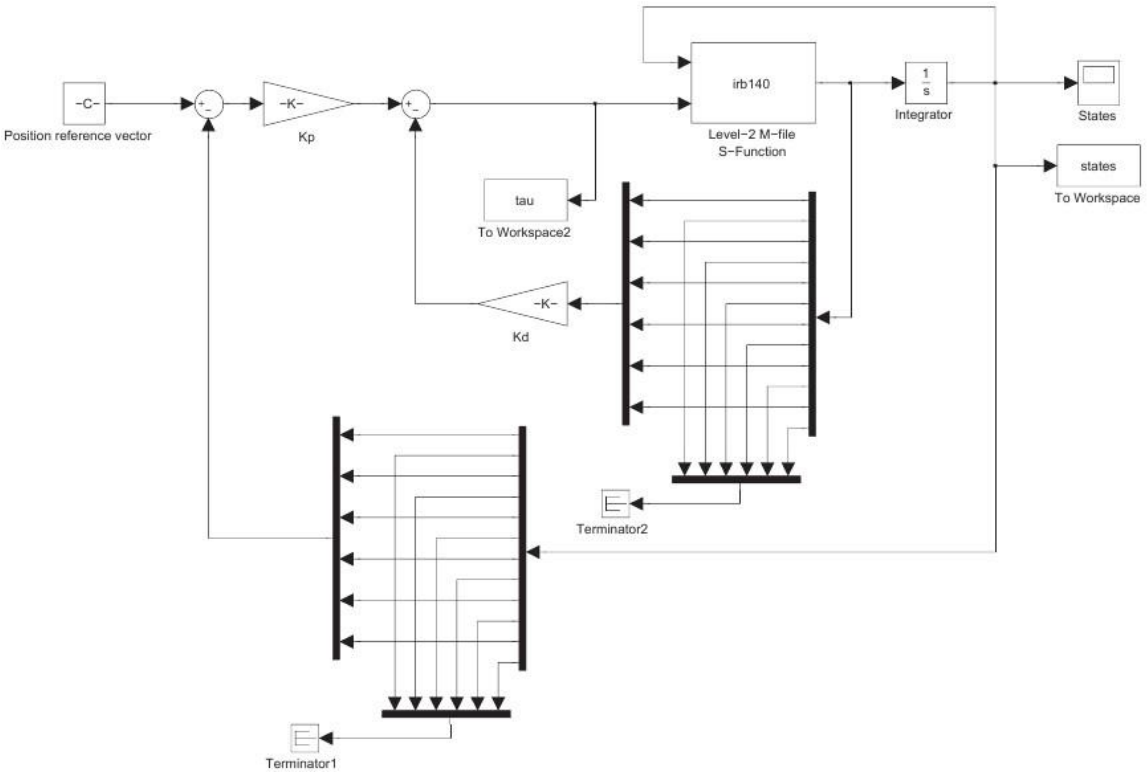


Figure 5.2 Closed loop simulation model with PD-controller

The mathematical proof gives no other bounds on K_p and K_d except for being positive definite. Adjusting these controller gains optimally have not been a priority, because it will not be decisive for global asymptotic stability.

A set of satisfying gain matrices was found as simple as

$$K_p = \begin{bmatrix} 50 & 0 & 0 & 0 & 0 & 0 \\ 0 & 50 & 0 & 0 & 0 & 0 \\ 0 & 0 & 50 & 0 & 0 & 0 \\ 0 & 0 & 0 & 50 & 0 & 0 \\ 0 & 0 & 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 0 & 0 & 60 \end{bmatrix} \tag{5.30}$$

$$K_d = \begin{bmatrix} 20 & 0 & 0 & 0 & 0 & 0 \\ 0 & 20 & 0 & 0 & 0 & 0 \\ 0 & 0 & 20 & 0 & 0 & 0 \\ 0 & 0 & 0 & 20 & 0 & 0 \\ 0 & 0 & 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 0 & 0 & 22 \end{bmatrix} \tag{5.31}$$

6. Comparison of results

In this work considering the two main approaches for dynamic modeling of robot manipulators

- Method based on the Newton-Euler formulation
- Method based on the product of exponential formula

Since in a present-day world the mechanic laws remain unchanged then method based on product of exponential formula, from the point of view of mechanic laws the similar with Lagrange-Euler method. Distinguishing feature is a way of determine inertia matrix.

In Chapter 2.9.1 it was stated that there is no clear answer to the question of which of the methods is better than the other, because of all the factors that influence the computation time. However the Chapter 6 proves at least that a recursive procedure is more efficient than treating the manipulator as a whole.

The purpose of this chapter is to compare the behavior and computation times of the models derived by the Newton-Euler formulation and by method based on product of exponential formula.

6.1. Simulation and comparison

6.1.1. The open loop

Simulation 1

The initial conditions for simulation, results are cited on Figure 6.1

$$q_{ne,init} = \begin{bmatrix} 0 & \frac{\pi}{2} & -\frac{\pi}{2} & 0 & 0 & 0 \end{bmatrix}^T \quad (6.1)$$

$$\dot{q}_{ne,init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (6.2)$$

$$\tau_{ne,init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (6.3)$$

$$q_{exp,init} = \begin{bmatrix} 0 & \frac{\pi}{2} & -\frac{\pi}{2} & 0 & 0 & 0 \end{bmatrix}^T \quad (6.4)$$

$$\dot{q}_{exp,init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (6.5)$$

$$\tau_{exp,init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (6.6)$$

Simulation 2

The initial conditions for simulation, results are cited on Figure 6.2

$$q_{ne,init} = \begin{bmatrix} 0 & \pi & -\frac{\pi}{2} & 0 & 0 & 0 \end{bmatrix}^T \quad (6.7)$$

$$\dot{q}_{ne,init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (6.8)$$

$$\tau_{ne,init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (6.9)$$

$$q_{exp,init} = \begin{bmatrix} 0 & \pi & -\frac{\pi}{2} & 0 & 0 & 0 \end{bmatrix}^T \quad (6.10)$$

$$\dot{q}_{exp,init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (6.11)$$

$$\tau_{exp,init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (6.12)$$

The initial conditions correspond to the configuration where link 2 is hanging straight down, while link 4 and 6 represents a double inverted pendulum on top of link 2.

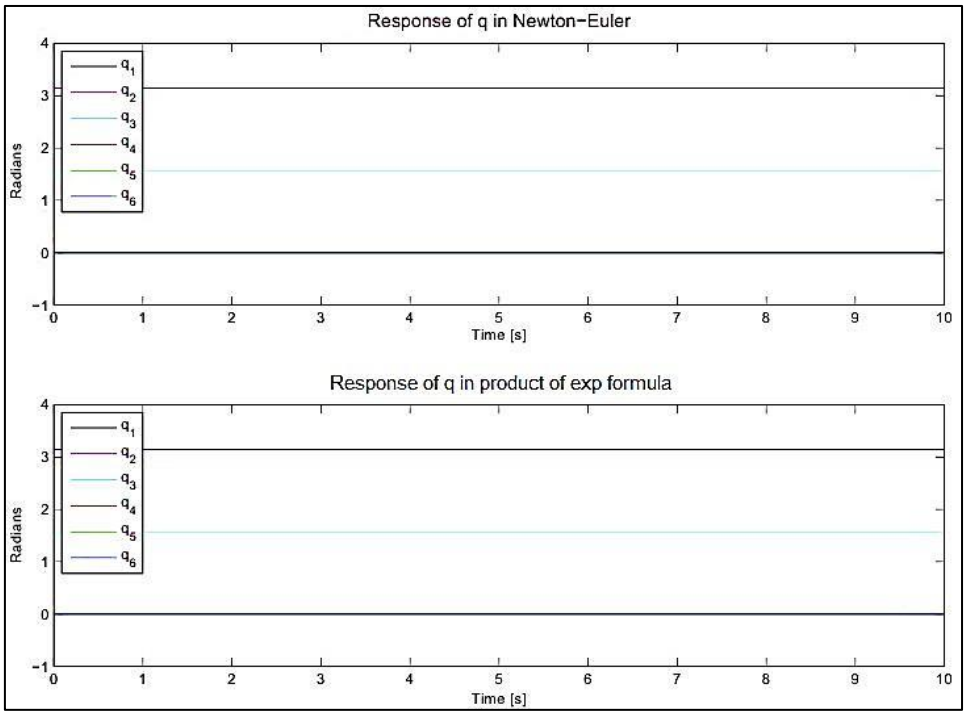


Figure 6.1 Comparison of the method based on Newton-Euler formulation with method based on product of exponential formula

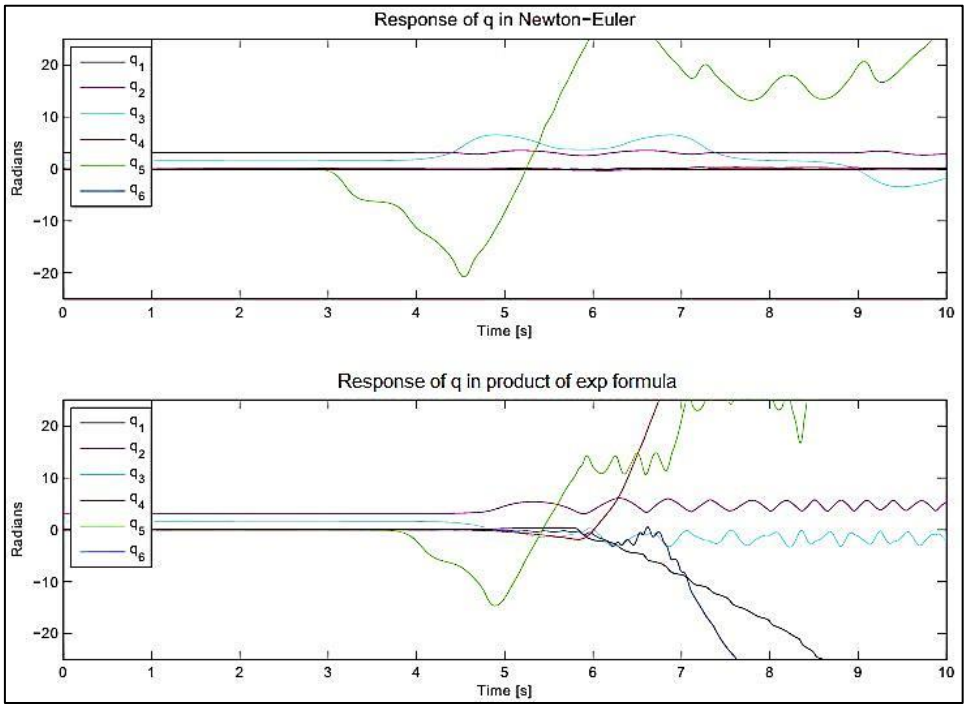


Figure 6.2 Comparison of the method based on Newton-Euler formulation with method based on product of exponential formula

6.1.2. Comments

Simulation 2 show a clearly unstable behavior when attempting to control the system to a desired position that is not in immediate proximity to the initial conditions. As mentioned, this is

just as expected because the excitation of gravity on the links is dependent on the joint variables. In simulation 1, the initial conditions are equal to the desired position and velocity, and the graph is showing the expected response. The joints are actuated exactly as required to compensate for the gravity and to keep the manipulator steady in the desired state.

The conclusion corresponds to what was assumed in advance of the simulations. The behavior of the system is unstable, and just the slightest disturbance in the system leads to a completely uncontrollable motion because the gravity on the links is dependent on the joint variables, and the input is computed without observing the output. The system requires feedback controllers to be stabilized.

6.1.1. Closed loop

Simulation 1

The initial conditions and reference value are set up equal to

$$q_{init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \tag{6.13}$$

$$\dot{q}_{init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \tag{6.14}$$

$$q_{ref} = \left[\frac{\pi}{2} \quad 0 \quad -\frac{\pi}{2} \quad \pi \quad \frac{\pi}{2} \quad -\pi\right]^T \tag{6.15}$$

on Figure 6.3 are cited characteristics of manipulator position, on Figure 6.4 are cited characteristics of input torque.

Simulation 2

The initial conditions and reference value are set up equal to

$$q_{init} = \left[0 \quad \pi \quad -\frac{\pi}{2} \quad 0 \quad 0 \quad 0\right]^T \tag{6.16}$$

$$\dot{q}_{init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \tag{6.17}$$

$$q_{ref} = \left[\pi \quad 0 \quad 0 \quad \pi \quad \frac{\pi}{2} \quad -\pi\right]^T \tag{6.18}$$

on Figure 6.5 are cited characteristics of manipulator position, on Figure 6.6 are cited characteristics of input torque.

Simulation 3

The initial conditions and reference value are set up equal to

$$q_{init} = \left[0 \quad \frac{\pi}{2} \quad -\frac{\pi}{2} \quad 0 \quad 0 \quad 0\right]^T \tag{6.19}$$

$$\dot{q}_{init} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \tag{6.20}$$

$$q_{ref} = \left[-\pi \quad \pi \quad -\pi \quad -\pi \quad -\frac{\pi}{2} \quad \pi\right]^T \tag{6.21}$$

on Figure 6.7 are cited characteristics of manipulator position, on Figure 6.8 are cited characteristics of input torque.

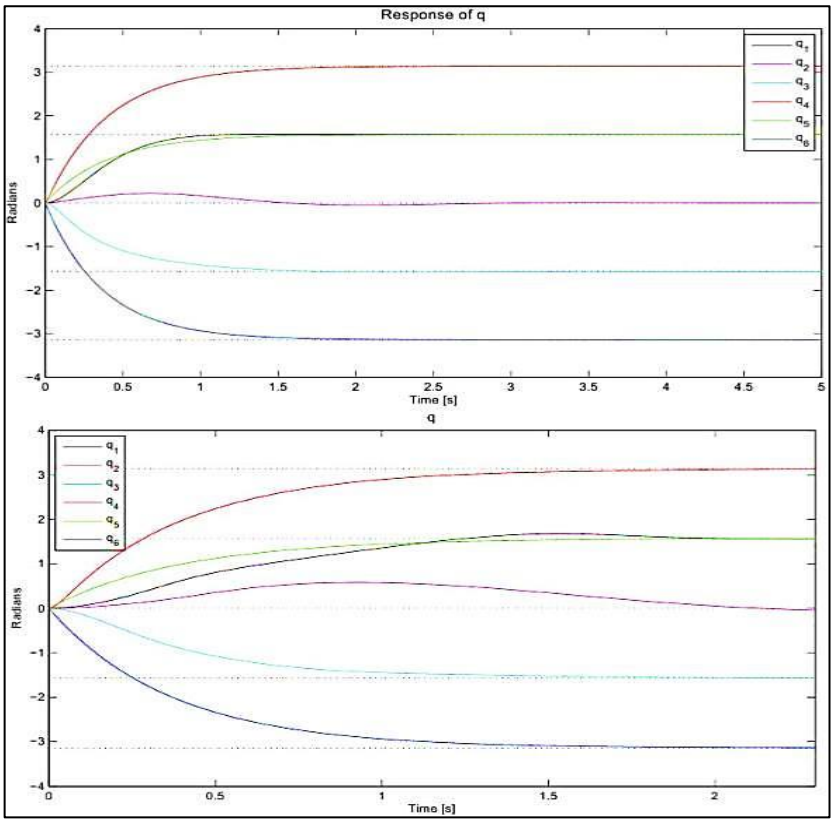


Figure 6.3 Comparison of the characteristic of system with closed loop, position control , Simulation 1

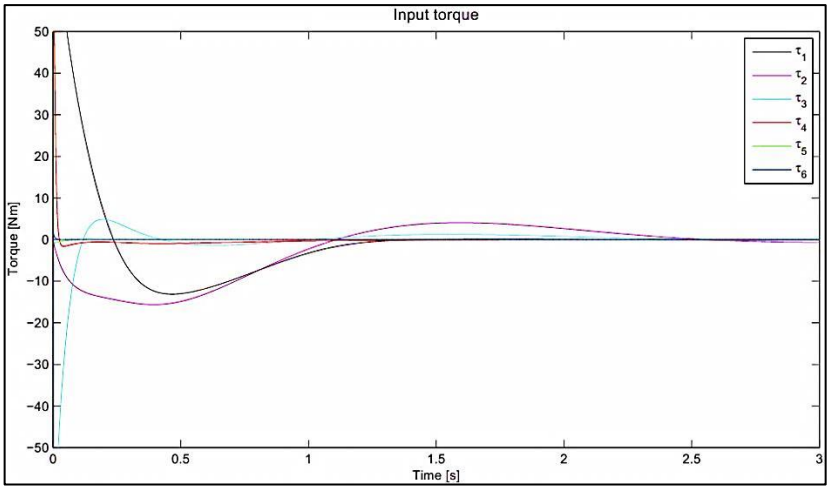


Figure 6.4 Characteristic of system with closed loop, input torque, Simulation 1

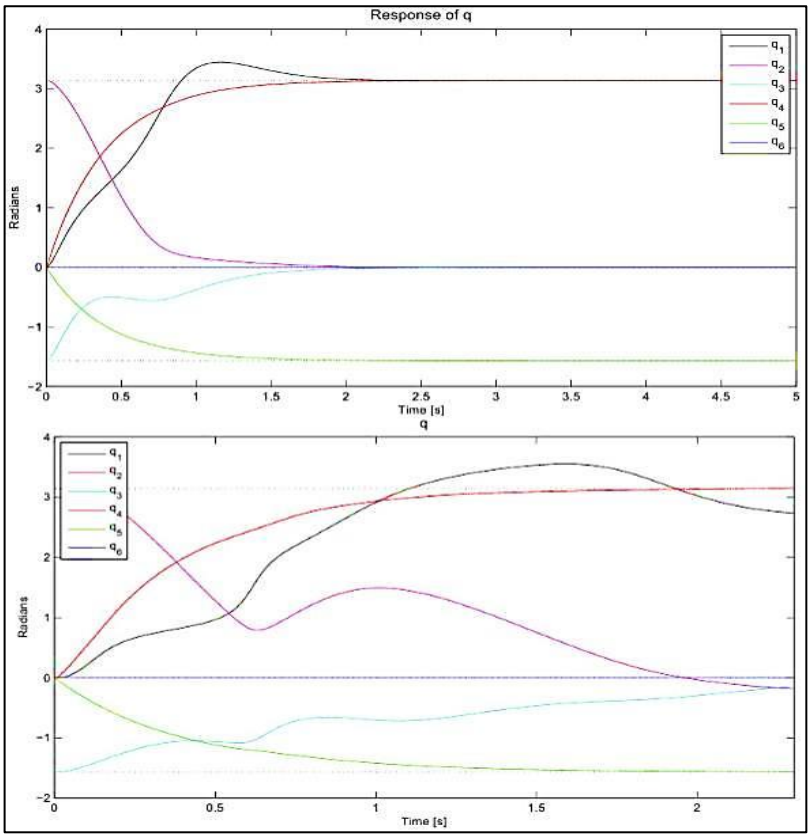


Figure 6.5 Comparison of the characteristic of system with closed loop, position control, Simulation 2

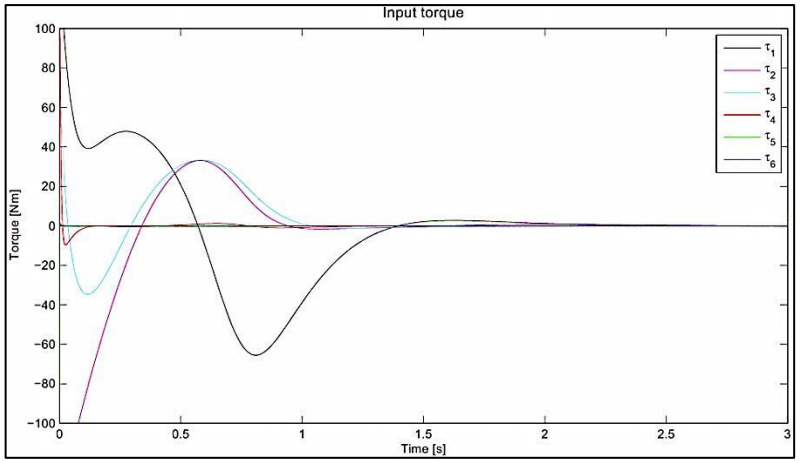


Figure 6.6 characteristic of system with closed loop, input torque, Simulation 2

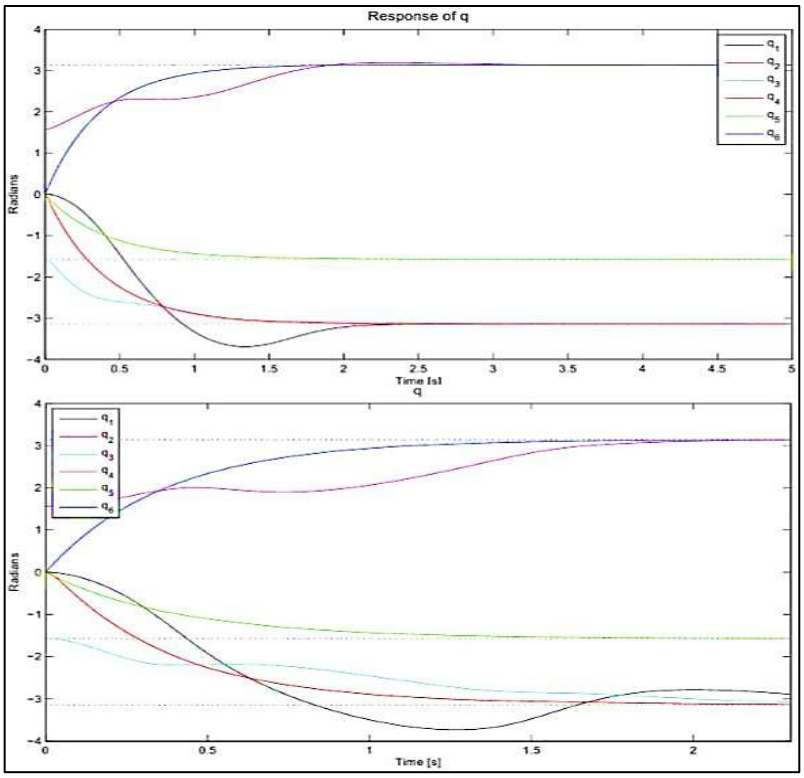


Figure 6.7 Comparison of the characteristic of system with closed loop, position control, Simulation 3

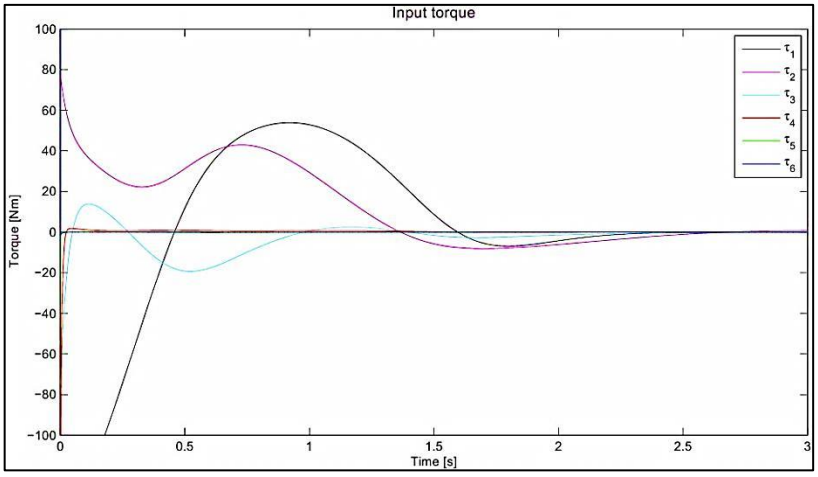


Figure 6.8 characteristic of system with closed loop, input torque, Simulation 3

6.1.2. Comments

All simulations show that the states converge to the reference in about 3 seconds. Asymptotic stability is verified, and the response is very satisfying. Nevertheless, several factors deserve to be emphasized. First of all, actuators cannot supply infinite torque. The nominal torque of the actuators and their gear ratio limits the maximum input torque. This constraint can be included in Simulink simply by saturating the input, but doing this is not necessary of reasons explained as follows. In the proof in Section 5.4.1 the gravitational terms $g(q)$ were added to the input because the gravitational acceleration was assumed to be constant and known. With this simplification it is taken for granted that the maximum input torque in the actuators is larger than $g(q)$. The data sheets for the IRB 140 do not state any torque values or other motor characteristics, but obviously this assumption is valid since the manipulator is observed to "beat the gravity" in a real environment. The mathematical proof gives no other bounds on the input torque, thus global asymptotic stability is proved also for saturated inputs. The only difference in the simulations will be the increased time to reach steady state.

Secondly, actuators cannot change the input torque value from τ_a to τ_b in zero time. In other words, the input can never be a perfect step function. The IRB 140 are controlled by electric AC-motors which supplies torque by passing electricity to an electromagnet creating a magnetic field. How fast this magnetic field is created will determine the maximum rate of change in input torque. Rate limiters can be included in Simulink, but it is assumed that electric motors create their electric fields very quickly. Consequently, rate limiters will not make any significant difference in the simulations.

Some limitations have been chosen deliberately. First, joint friction is not taken into account because of two reasons. First, it will be like a shot in the dark to estimate the friction parameters without any given information. Secondly, it does not really make a difference to the simulations anyway when the input is not saturated. However, if joint friction was to be taken into account, the simplest way to include it would be to only model viscous friction, being proportional to the joint velocity. The system model would then be

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + F_v\dot{q} + \dot{q} + g(q) = u$$

where $F_v$is a diagonal matrix of the joint friction coefficients.

Note also that the simulations do not take into account the workspace of the manipulator at all. Since the main goal of this chapter is to prove the validity of the model, and not to optimize a control system for a specific job task, it was found convenient to not include the workspace restrictions. The joints are allowed to revolve freely, and no obstacles, floor, roof or walls are considered. The data sheet (Attachment A1) specifies the actual working range for the joints.

It should be mentioned that there exists several other control techniques and methodologies that can be applied to the control of manipulators. The choice of control structure should therefore match the requirements for the robot operation. If there are obstacles within the workspace of the manipulator, continuous path tracking could be necessary to avoid collisions. Many operations may also require that the manipulator moves from point A to point B in a precise fixed time interval. If

the robot operation requires objects to be moved around, robust and adaptive controllers are superior. Note that this can often be the case for the IRB 140, as it is designed to handle payloads of up to 6 kg. The mechanical design, motor characteristics, and problems due to backlash, friction and gear reduction, may also affect the choice of control structure.

6.2. Computation times

This section will investigate the efficiency of the formulations described in terms of computation times in open loop and closed loop. To distinguish clearly, the simulation time chosen in Simulink will be referred to as *simulation time*, and the actual time recorded during the simulation will be referred to as *real time.*

6.2.1. Open loop

The simulations showed in Figure 6.2 are used to compare computation times in open loop. The simulation times for both models were set to 10 seconds. The real times were recorded as 7 minutes for the model based o n product of exponential formula and only 6 seconds for the Newton-Euler model.

6.2.2. Closed loop

In subsection 6.1.2 three different simulations for the both of models were performed in closed loop with PD controllers (see Section 5.4.1).

The analyze of model based on Newton-Euler formulation showed that the states converged to the reference in about 3 seconds of simulation time. The total simulation times were 5 seconds for all simulations, and the real times were recorded to be 28 minutes, 32 minutes and 27 minutes respectively.

Equivalent simulations in closed loop with PD controllers have been performed with the model based on product of exponential formula, and the results are quite remarkable. The simulations were awfully time-consuming and they required so much computer capacity that it was chosen to stop the simulations after 2.3 seconds of simulation time. The real time was then at about 18 hours for all three simulations.

6.2.3. Comments

All simulation times and recorded real times in this comparison are summarized in Table 6.1. Several times throughout this thesis it has been pointed out that a recursive procedure is faster than treating the manipulator as a whole.

Diagrams	Newton - Euler		Product of exponential formula	
	Sim.time	Real time	Sim.time	Real time
Figure 6.2	10 sec	6 min	10 min	7 min
Figure 6.3	5 sec	28 min	2.3 min	18 h
Figure 6.5	5 sec	32 min	2.3 min	18 h
Figure 6.7	5 sec	27 min	2.3 min	18 h

Since exist at others software for simulation, which can have a other results. I was counted the quantity of math operations in each of dynamic equations. This analyze showed that recursion method based on Newton-Euler formulation more useful for work in real time, the results are given in table 6.2:

Newton - Euler	Product of exponential formula
16080	39001

7. Conclusion

The main task of this thesis has been in the comparison of dynamic modeling and simulation of robot manipulators. Two different methods for dynamic modeling have been introduced, method based on Newton-Euler formulation and method based on product of exponential formula. The results which were obtained during investigation shows that method based on Newton-Euler formulation more efficiency in the view of practical using, but method based on product of exponential formula more useful in determine the kinematic map. Although it is a difficult conclusion to the question of which method is better than the other in general. The computation time depends on several aspects in the system to be analyzed, and the approaches provide different insights such that personal preference becomes a factor as well.

It has been shown that estimating the dynamic parameters accurately is a hard and time-consuming challenge. It requires either the possibility to measure the state variables and its derivatives during motion of the manipulator, or specific knowledge about other identification techniques as for example CAD modeling. Even if such an attempt is to be performed, the dynamic parameters will not be perfectly accurate. In the model for the IRB 140, the dynamic parameters have been estimated based on inspecting the manipulator carefully, making intuitive guesses when required.

Simulations of the dynamic model had as main purpose to prove the validity of the model. Open loop simulations with desired torque showed that the behavior of the system was unstable just as assumed; the slightest disturbance in the system led to a completely uncontrollable motion.

Global asymptotic stability of the system with PD control and gravity compensation was proved mathematically in a Lyapunov stability analysis. Afterwards, this was confirmed to be the case for the model by simulations with PD control.

It was mentioned that the computation times of the Newton-Euler formulation and model based on product of exponential formula depends on several factors. However, in case of dynamic model, it is a fact that a recursive procedure is more efficient than treating the manipulator as a whole.

REFERENCE

[1] SICILIANO B., KHATIB O.(Eds.). Handbook of Robotics. Berlin, Springer-Verlag, 2008.

[2] ANGELES J. Fundamentals of Robotics Mechanical Systems. New York, Springer-Verlag, 2003.

[3] MURRAY R. M., LI Z., SASTRY S. S. A Mathematical Introduction to Robotic Manipulation. CRC Press, 1994

[4] CANUDAS DE WIN C., SICILIANY B., BASTIN G.. Theory of Robot Control. Springer-Verlag, 1996.

[5] *PTC Mathcad. Mathcad 14* [online]. [viewed].
Available from: http://ru.ptc.com/product/mathcad

[6] *Maplesoft. Maple* [online]. [viewed].
Available from: http://www.maplesoft.com/products/Maple/

[7] *Mathworks. Matlab* [online]. [viewed].
Available from: http://www.mathworks.com/products/

[8] *Microsoft. MicrosoftVisio* [online]. [viewed].
Available from: http://office.microsoft.com/ru-ru/FX103472299.aspx

[9] *SImetric*. [online]. [viewed].
Available from: http://www.simetric.co.uk/si_metals.htm

[10] *ABB* [online]. [viewed].
Available from: http://www.abb.ru/product/seitp327/d6b7a86a0170b4f544257c05003ff19a.aspx

[11] *PID–controller* [online]. [viewed].
Available from: https://ru.wikipedia.org/wiki/ПИД-регулятор

ATTACHMENT

A.1.

Robotics

IRB 140
Industrial Robot

Main Applications

Arc welding
Assembly
Cleaning/Spraying
Machine tending
Material handling
Packing
Deburring

Small, Powerful and Fast

Compact, powerful IRB 140 industrial robot.
Six axis multipurpose robot that handles payload of 6 kg, with long reach (810 mm). The IRB 140 can be floor mounted, inverted or on the wall in any angle. Available as Standard, Foundry Plus 2, Clean Room and Wash versions, all mechanical arms completely IP67 protected, making IRB 140 easy to integrate in and suitable for a variety of applications. Uniquely extended radius of working area due to bend-back mechanism of upper arm, axis 1 rotation of 360 degrees even as wall mounted.
The compact, robust design with integrated cabling adds to overall flexibility. The Collision Detection option with full path retraction makes robot reliable and safe.

Using IRB 140T, cycle-times are considerably reduced where axis 1 and 2 predominantly are used.
Reductions between 15-20 % are possible using pure axis 1 and 2 movements. This faster versions is well suited for packing applications and guided operations together with PickMaster.
IRB 140 Foundry Plus 2 and Wash versions are suitable for operating in extreme foundry environments and other harch environments with high requirements on corrosion resistance and tightness. In addition to the IP67 protection, excellent surface treatment makes the robot high pressure steam washable. Also available in white Clean Room ISO class 6 version, making it especially suited for environments with stringent cleanliness standards.

Power and productivity
for a better world™

IRB 140

Specification

Robot versions	Handling capacity	Reach of 5th axis	Remarks
IRB 140/IRB 140T	6 kg	810 mm	
IRB 140F/IRB 140TF	6 kg	810 mm	Foundry Plus 2 Protection
IRB 140CR/IRB 140TCR	6 kg	810 mm	Clean Room
IRB 140W/IRB 140TW	6 kg	810 mm	SteamWash Protection
Supplementary load (on upper arm alt. wrist)			
on upper arm		1 kg	
on wrist		0.5 kg	
Number of axes			
Robot manipulator		6	
External devices		6	
Integrated signal supply	12 signals on upper arm		
Integrated air supply	Max. 8 bar on upper arm		
IRC5 Controller variants:	Single cabinet, Dual cabinet, Compact, Panel mounted		

Performance

Position repeatability 0.03 mm (average result from ISO test)

Axis movement	Axis	Working range
	1	360°
	2	200°
	3	280°
	4	Unlimited (400° default)
	5	240°
	6	Unlimited (800° default)
Max. TCP velocity		2.5 m/s
Max. TCP acceleration		20 m/s^2
Acceleration time 0-1 m/s		0.15 sec

Velocity *)

Axis no.	IRB 140	IRB 140T
1	200°/s	250°/s
2	200°/s	250°/s
3	260°/s	260°/s
4	360°/s	360°/s
5	360°/s	360°/s
6	450°/s	450°/s

*) Max velocity is reduced at single phase power supply, e.g. Compact controller. Please, see the Product specification for further details.

Cycle time

5 kg Picking side	IRB 140	IRB 140T
cycle 25 x 300 x 25 mm	0.85s	0.77s

Electrical Connections

Supply voltage	200–600 V, 50/60 Hz
Rated power	
Transformer rating	4.5 kVA
Power consumption typicly	0.4 kW

Physical

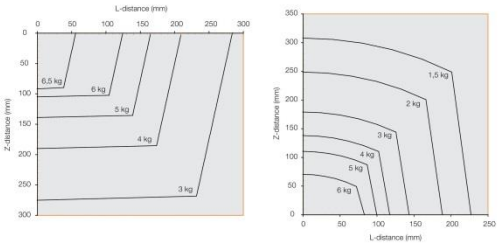
Robot mounting	Any angle
Dimensions	
Robot base	400 x 450 mm
Robot controller H x W x D	950 x 800 x 620 mm
Weight	
Robot manipulator	98 kg

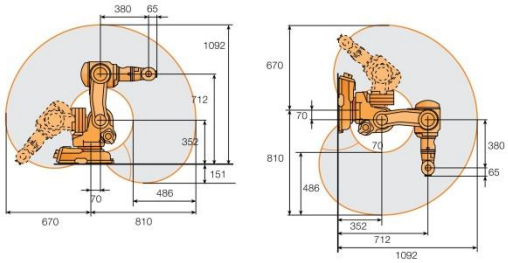
Environment

Ambient temperature for	
Robot manipulator	5 – 45°C
Relative humidity	Max. 95%
Degree of protection,	
Manipulator	IP67
Options	Foundry Plus 2 SteamWash (High pressure steam washable) Clean Room, class 6 (certified by IPA)
Noise level	Max. 70 dB (A)
Safety	Double circuits with supervision, emergency stops and safety functions, 3-position enable device
Emission	EMC/EMI-shielded

Data and dimensions may be changed without notice

Working range





© Copyright ABB Robotics. PR10031EN_R14 March 2012.

www.abb.com/robotics

Power and productivity for a better world™

A.2.

```
> #Dynamic model of ABB IRB 140
> #based on recursion mthod of Newton-Euler
>
> restart :
> with(LinearAlgebra) :
>
> #Defining joint variable
>
> q := ⟨q1(t), q2(t), q3(t), q4(t), q5(t), q6(t)⟩ :
> Dq := map(diff, q, t) :
> DDq := map(diff, Dq, t) :
>
> #Defining link length vectors
> r0c1 := ⟨0.014, -0.264, 0.067⟩ :
> r1c2 := ⟨0.201, 0, -0.070⟩ :
> r2c3 := ⟨0, 0, 0⟩ :
> r3c4 := ⟨0, 0.080, 0⟩ :
> r4c5 := ⟨0, 0, 0⟩ :
> r5c6 := ⟨0, 0, 0.029⟩ :
> r01 := ⟨0.070, -0.352, 0⟩ :
> r12 := ⟨0.360, 0, 0⟩ :
> r23 := ⟨0, 0, 0⟩ :
> r34 := ⟨0, 0.380, 0⟩ :
> r45 := ⟨0, 0, 0⟩ :
> r56 := ⟨0, 0, 0.065⟩ :
> r1c1 := r0c1 − r01 :
> r2c2 := r1c2 − r12 :
> r3c3 := r2c3 − r23 :
> r4c4 := r3c4 − r34 :
> r5c5 := r4c5 − r45 :
> r6c6 := r5c6 − r56 :
>
> #Gravity vector in inertial frame
> g0 := ⟨0, 0, -g⟩ :
>
> #Rotation matrices
> R01 := MatrixMatrixMultiply( [ cos(q[1])  -sin(q[1])  0 ]
                               [ sin(q[1])   cos(q[1])  0 ]
                               [    0           0       1 ] , [ 1  0  0 ]
                                                              [ 0  0  -1 ]
                                                              [ 0  1  0 ] ) :
> R12 := MatrixMatrixMultiply( [ cos(q[2])  -sin(q[2])  0 ]
                               [ sin(q[2])   cos(q[2])  0 ]
                               [    0           0       1 ] , [ 1  0  0 ]
                                                              [ 0  1  0 ]
                                                              [ 0  0  1 ] ) :
```

```
> R23 := MatrixMatrixMultiply( [[cos(q[3]), -sin(q[3]), 0], [sin(q[3]), cos(q[3]), 0], [0, 0, 1]], [[1, 0, 0], [0, 0, -1], [0, 1, 0]] ) :
> R34 := MatrixMatrixMultiply( [[cos(q[4]), -sin(q[4]), 0], [sin(q[4]), cos(q[4]), 0], [0, 0, 1]], [[1, 0, 0], [0, 0, -1], [0, 1, 0]] ) :
> R45 := MatrixMatrixMultiply( [[cos(q[5]), -sin(q[5]), 0], [sin(q[5]), cos(q[5]), 0], [0, 0, 1]], [[1, 0, 0], [0, 0, -1], [0, 1, 0]] ) :
> R56 := MatrixMatrixMultiply( [[cos(q[6]), -sin(q[6]), 0], [sin(q[6]), cos(q[6]), 0], [0, 0, 1]], [[1, 0, 0], [0, 1, 0], [0, 0, 1]] ) :
> R02 := MatrixMatrixMultiply(R01, R12) :
> R03 := MatrixMatrixMultiply(R02, R23) :
> R04 := MatrixMatrixMultiply(R03, R34) :
> R05 := MatrixMatrixMultiply(R04, R45) :
> R06 := MatrixMatrixMultiply(R05, R56) :
>
> #Axis of rotation of joint i expressed in frame i
> z0 := <0, 0, 1> :
> b1 := MatrixMatrixMultiply( (R01^%T), z0) :
> b2 := combine(MatrixMatrixMultiply( (R02^%T), MatrixVectorMultiply(R01, z0) ), trig) :
> b3 := combine(MatrixMatrixMultiply( (R03^%T), MatrixVectorMultiply(R02, z0) ), trig) :
> b4 := combine(MatrixMatrixMultiply( (R04^%T), MatrixVectorMultiply(R03, z0) ), trig) :
> b5 := combine(MatrixMatrixMultiply( (R05^%T), MatrixVectorMultiply(R04, z0) ), trig) :
> b6 := combine(MatrixMatrixMultiply( (R06^%T), MatrixVectorMultiply(R05, z0) ), trig) :
>
> #Link Masses
> m1 := 27 :
> m2 := 22 :
> m3 := 0 :
> m4 := 25 :
> m5 := 0 :
> m6 := 1 :
>
> #Cylinder link dimentions parametrs
> r1 := 0.191 :
> r2 := 0.151 :
> r3 := 0 :
```

```
> r4 := 0.115 :
> r5 := 0 :
> r6 := 0.044 :
>
> h1 := 0.363 :
> h2 := 0.515 :
> h3 := 0 :
> h4 := 0.583 :
> h5 := 0 :
> h6 := 0.107 :
>
> #Inertia matrices
```

$$> I1 := \begin{bmatrix} \frac{1}{12}\cdot m1\cdot\left(3\cdot r1^2 + h1^2\right) & 0 & 0 \\ 0 & \frac{1}{2}\cdot m1\cdot r1^2 & 0 \\ 0 & 0 & \frac{1}{12}\cdot m1\cdot\left(3\cdot r1^2 + h1^2\right) \end{bmatrix} :$$

$$> I2 := \begin{bmatrix} \frac{1}{2}\cdot m2\cdot r2^2 & 0 & 0 \\ 0 & \frac{1}{12}\cdot m2\cdot\left(3\cdot r2^2 + h2^2\right) & 0 \\ 0 & 0 & \frac{1}{12}\cdot m2\cdot\left(3\cdot r2^2 + h2^2\right) \end{bmatrix} :$$

$$> I3 := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} :$$

$$> I4 := \begin{bmatrix} \frac{1}{12}\cdot m4\cdot\left(3\cdot r4^2 + h4^2\right) & 0 & 0 \\ 0 & \frac{1}{2}\cdot m4\cdot r4^2 & 0 \\ 0 & 0 & \frac{1}{12}\cdot m4\cdot\left(3\cdot r4^2 + h4^2\right) \end{bmatrix} :$$

$$> I5 := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} :$$

```
> I6 := [ 1/12 ·m6·(3·r6^2 + h6^2)        0                    0
          0                              1/12 ·m6·(3·r6^2 + h6^2)  0
          0                              0                    1/2 ·m6·r6^2 ] :
>
> #Forward recursion:Link 1
> ω1 := b1·Dq[1] :
> α1 := b1·DDq[1] + CrossProduct(ω1, b1·Dq[1]) :
> Dω1 := map(diff, ω1, t) :
> ae1 := CrossProduct(Dω1, r01) + CrossProduct(ω1, CrossProduct(ω1, r01)) :
> ac1 := CrossProduct(Dω1, r0c1) + CrossProduct(ω1, CrossProduct(ω1, r0c1)) :
> g1 := MatrixVectorMultiply((R01^%T), g0) :
>
> #Forward recursion:Link 2
> ω2 := combine(MatrixVectorMultiply((R12^%T), ω1) + b2·Dq[2], trig) :
> α2 := combine(MatrixVectorMultiply((R12^%T), α1) + b2·DDq[2] + CrossProduct(ω2, b2
      ·Dq[2]), trig) :
> Dω2 := map(diff, ω2, t) :
> ae2 := combine(MatrixVectorMultiply((R12^%T), ae1) + CrossProduct(Dω2, r12)
      + CrossProduct(ω2, r12)), trig) :
> ac2 := combine(MatrixVectorMultiply((R12^%T), ae1) + CrossProduct(Dω2, r1c2)
      + CrossProduct(ω2, CrossProduct(ω2, r1c2)), trig) :
> g2 := combine(MatrixVectorMultiply((R02^%T), g0), trig) :
>
> #Forward recursion:Link 3
> ω3 := combine(MatrixVectorMultiply((R23^%T), ω2) + b3·Dq[3], trig) :
> α3 := combine(MatrixVectorMultiply((R23^%T), α2) + b3·DDq[3] + CrossProduct(ω3, b3
      ·Dq[3]), trig) :
> Dω3 := map(diff, ω3, t) :
> ae3 := combine(MatrixVectorMultiply((R23^%T), ae2), trig) :
> ac3 := combine(MatrixVectorMultiply((R23^%T), ae2), trig) :
> g3 := combine(MatrixVectorMultiply((R03^%T), g0), trig) :
>
> #Forward recursion:Link 4
> ω4 := combine(MatrixVectorMultiply((R34^%T), ω3) + b4·Dq[4], trig) :
> α4 := combine(MatrixVectorMultiply((R34^%T), α3) + b4·DDq[4] + CrossProduct(ω4, b4
      ·Dq[4]), trig) :
```

```
> Dω4 := map(diff, ω4, t) :
> ae4 := combine(MatrixVectorMultiply( (R34^%T), ae3) + CrossProduct(Dω4, r34)
      + CrossProduct( ω4, CrossProduct( ω4, r34) ), trig) :
> ac4 := combine(MatrixVectorMultiply( (R34^%T), ae3) + CrossProduct(Dω4, r3c4)
      + CrossProduct( ω4, CrossProduct( ω4, r3c4) ), trig) :
> g4 := combine(MatrixVectorMultiply( (R04^%T), g0), trig) :
>
> #Forward recursion:Link 5
> ω5 := combine(MatrixVectorMultiply( (R45^%T), ω4) + b5·Dq[5], trig) :
> α5 := combine(MatrixVectorMultiply( (R45^%T), α4) + b5·DDq[5] + CrossProduct( ω5, b5
      ·Dq[5]), trig) :
> Dω5 := map(diff, ω5, t) :
> ae5 := combine(MatrixVectorMultiply( (R45^%T), ae4), trig) :
> ac5 := combine(MatrixVectorMultiply( (R45^%T), ae4), trig) :
> g5 := combine(MatrixVectorMultiply( (R05^%T), g0), trig) :
>
> #Forward recursion:Link 6
> ω6 := combine(MatrixVectorMultiply( (R56^%T), ω5) + b6·Dq[6], trig) :
> α6 := combine(MatrixVectorMultiply( (R56^%T), α5) + b6·DDq[6] + CrossProduct( ω6, b6
      ·Dq[6]), trig) :
> Dω6 := map(diff, ω6, t) :
> ae6 := combine(MatrixVectorMultiply( (R56^%T), ae5) + CrossProduct(Dω6, r56)
      + CrossProduct( ω6, CrossProduct( ω6, r56) ), trig) :
> ac6 := combine(MatrixVectorMultiply( (R56^%T), ae5) + CrossProduct(Dω6, r5c6)
      + CrossProduct( ω6, CrossProduct( ω6, r5c6) ), trig) :
> g6 := combine(MatrixVectorMultiply( (R06^%T), g0), trig) :
>
> #Backward recursion:Link 6
> f6 := Add(m6·ac6, -m6·g6) :
> τ6 := CrossProduct( -f6, r5c6) + MatrixVectorMultiply(I6, α6) + CrossProduct( ω6,
      MatrixVectorMultiply(I6, ω6) ) :
> τ6z := collect( combine( MatrixVectorMultiply( (b6^%T), τ6), trig), {DDq[1], DDq[2],
      DDq[3], DDq[4], DDq[5], DDq[6], Dq[1], Dq[2], Dq[3], Dq[4], Dq[5], Dq[6]}) :
>
> #Backward recursion:Link 5
> f5 := MatrixVectorMultiply(R56, f6) :
> τ5 := MatrixVectorMultiply(R56, τ6) + MatrixVectorMultiply(I5, α5) + CrossProduct( ω5,
      MatrixVectorMultiply(I5, ω5) ) :
> τ5y := collect( combine( MatrixVectorMultiply( (b5^%T), τ5), trig), {DDq[1], DDq[2],
      DDq[3], DDq[4], DDq[5], DDq[6], Dq[1], Dq[2], Dq[3], Dq[4], Dq[5], Dq[6]}) :
```

```
>
> #Backward recursion:Link 4
> f4 := MatrixMatrixMultiply(R45, f5) + Add(m4·ac4, -m4·g4) :
> τ4 := MatrixVectorMultiply(R45, τ5) − CrossProduct(f4, r3c4)
      + CrossProduct(MatrixVectorMultiply(R45, f5), r4c4) + MatrixVectorMultiply(I4, α4)
      + CrossProduct(ω4, MatrixVectorMultiply(I4, ω4)) :
> τ4x := collect(combine(MatrixVectorMultiply((b4^%T), τ4), trig), {DDq[1], DDq[2],
      DDq[3], DDq[4], DDq[5], DDq[6], Dq[1], Dq[2], Dq[3], Dq[4], Dq[5], Dq[6]}) :
>
> #Backward recursion:Link 3
> f3 := MatrixVectorMultiply(R34, f4) :
> τ3 := MatrixVectorMultiply(R34, τ4) + CrossProduct(ω3, MatrixVectorMultiply(I3, ω3))
      + MatrixVectorMultiply(I3, α3) :
> τ3z := collect(combine(MatrixVectorMultiply((b3^%T), τ3), trig), {DDq[1], DDq[2],
      DDq[3], DDq[4], DDq[5], DDq[6], Dq[1], Dq[2], Dq[3], Dq[4], Dq[5], Dq[6]}) :
>
> #Backward recursion:Link 2
> f2 := MatrixVectorMultiply(R23, f3) + Add(m2·ac2, -m2·g2) :
> τ2 := MatrixVectorMultiply(R23, τ3) − CrossProduct(f2, r1c2)
      + CrossProduct(MatrixVectorMultiply(R23, f3), r2c2) + MatrixVectorMultiply(I2, α2)
      + CrossProduct(ω2, MatrixVectorMultiply(I2, ω2)) :
> τ2y := collect(combine(MatrixVectorMultiply((b2^%T), τ2), trig), {DDq[1], DDq[2],
      DDq[3], DDq[4], DDq[5], DDq[6], Dq[1], Dq[2], Dq[3], Dq[4], Dq[5], Dq[6]}) :
>
> #Backward recursion:Link 1
> f1 := MatrixVectorMultiply(R12, f2) + Add(m1·ac1, -m1·g1) :
> τ1 := MatrixVectorMultiply(R12, τ2) − CrossProduct(f1, r0c1)
      + CrossProduct(MatrixVectorMultiply(R12, f2), r1c1) + MatrixVectorMultiply(I1, α1)
      + CrossProduct(ω1, MatrixVectorMultiply(I1, ω1)) :
> τ1x := collect(combine(MatrixVectorMultiply((b1^%T), τ1), trig), {DDq[1], DDq[2],
      DDq[3], DDq[4], DDq[5], DDq[6], Dq[1], Dq[2], Dq[3], Dq[4], Dq[5], Dq[6]}) :
>
> #Setting up the matrix elements
> m[1, 1] := eval(τ1x, {DDq[1] = 1, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[1, 2] := eval(τ1x, {DDq[1] = 0, DDq[2] = 1, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[1, 3] := eval(τ1x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 1, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[1, 4] := eval(τ1x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 1, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
```

```
> m[1, 5] := eval( τ1x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 1,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[1, 6] := eval( τ1x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 1, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
>
> m[2, 1] := eval( τ2y, {DDq[1] = 1, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[2, 2] := eval( τ2y, {DDq[1] = 0, DDq[2] = 1, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[2, 3] := eval( τ2y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 1, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[2, 4] := eval( τ2y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 1, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[2, 5] := eval( τ2y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 1,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[2, 6] := eval( τ2y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 1, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
>
> m[3, 1] := eval( τ3z, {DDq[1] = 1, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
      = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[3, 2] := eval( τ3z, {DDq[1] = 0, DDq[2] = 1, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
      = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[3, 3] := eval( τ3z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 1, DDq[4] = 0, DDq[5] = 0, DDq[6]
      = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[3, 4] := eval( τ3z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 1, DDq[5] = 0, DDq[6]
      = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[3, 5] := eval( τ3z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 1, DDq[6]
      = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[3, 6] := eval( τ3z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
      = 1, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
>
> m[4, 1] := eval( τ4x, {DDq[1] = 1, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[4, 2] := eval( τ4x, {DDq[1] = 0, DDq[2] = 1, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[4, 3] := eval( τ4x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 1, DDq[4] = 0, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[4, 4] := eval( τ4x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 1, DDq[5] = 0,
      DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[4, 5] := eval( τ4x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 1,
```

```
    DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[4, 6] := eval(τ4x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
    DDq[6] = 1, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
>
> m[5, 1] := eval(τ5y, {DDq[1] = 1, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
    DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[5, 2] := eval(τ5y, {DDq[1] = 0, DDq[2] = 1, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
    DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[5, 3] := eval(τ5y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 1, DDq[4] = 0, DDq[5] = 0,
    DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[5, 4] := eval(τ5y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 1, DDq[5] = 0,
    DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[5, 5] := eval(τ5y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 1,
    DDq[6] = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[5, 6] := eval(τ5y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0,
    DDq[6] = 1, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
>
> m[6, 1] := eval(τ6z, {DDq[1] = 1, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
    = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[6, 2] := eval(τ6z, {DDq[1] = 0, DDq[2] = 1, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
    = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[6, 3] := eval(τ6z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 1, DDq[4] = 0, DDq[5] = 0, DDq[6]
    = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[6, 4] := eval(τ6z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 1, DDq[5] = 0, DDq[6]
    = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[6, 5] := eval(τ6z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 1, DDq[6]
    = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
> m[6, 6] := eval(τ6z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
    = 1, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0, g = 0}) :
>
> g6 := eval(τ6z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6] = 0,
    Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0}) :
> g5 := eval(τ5y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
    = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0}) :
> g4 := eval(τ4x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
    = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0}) :
> g3 := eval(τ3z, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6] = 0,
    Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0}) :
> g2 := eval(τ2y, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
    = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0}) :
```

```
> g1 := eval( τ1x, {DDq[1] = 0, DDq[2] = 0, DDq[3] = 0, DDq[4] = 0, DDq[5] = 0, DDq[6]
      = 0, Dq[1] = 0, Dq[2] = 0, Dq[3] = 0, Dq[4] = 0, Dq[5] = 0, Dq[6] = 0}) :
> ##############################
> #Setting up the dynamic system
```

$$> M0 := \begin{bmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} & m_{1,5} & m_{1,6} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} & m_{2,5} & m_{2,6} \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} & m_{3,5} & m_{3,6} \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} & m_{4,5} & m_{4,6} \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} & m_{5,5} & m_{5,6} \\ m_{6,1} & m_{6,2} & m_{6,3} & m_{6,4} & m_{6,5} & m_{6,6} \end{bmatrix} :$$

```
> G0 := ⟨g1, g2, g3, g4, g5, g6⟩ :
> τ0 := ⟨τ1x, τ2y, τ3z, τ4x, τ5y, τ6z⟩ :
> C0Dq := combine( τ0 − MatrixVectorMultiply(M0, DDq) − G0, trig) :
>
> #Matlab conversion
> with(CodeGeneration) :
> #Matrix M
```

▶ **m11**

▶ **m12**

▶ **m13**

▶ **m14**

▶ **m15**

▶ **m16**

```
>
```

▶ **m21**

▶ **m22**

▶ **m23**

▶ **m24**

▶ **m25**

▶ **m26**

[>

▶ **m31**

▶ **m32**

▶ **m33**

▶ **m34**

▶ **m35**

▶ **m36**

[>

▶ **m41**

▶ **m42**

▶ **m43**

▶ **m44**

▶ **m45**

▶ **m46**

[>

▶ **m51**

▶ **m52**

▶ **m53**

▶ **m54**

▶ **m55**

▶ **m56**

```
>
```

▶ **m61**

▶ **m62**

▶ **m63**

▶ **m64**

▶ **m65**

▶ **m66**

```
>
> #Gravity vector
```

▶ **g1**

▶ **g2**

▶ **g3**

▶ **g4**

▶ **g5**

▶ **g6**

```
>
> #Coriolis vector
```

▶ **cdq1**

▶ **cdq2**

▶ **cdq3**

▶ **cdq4**

▶ **cdq5**

▶ **cdq6**

A.3.

```
> #Dynamic model of ABB IRB 140
> #based on exponential formula

> restart :
> with(LinearAlgebra) :

> #Defining joint variable
>
> q := ⟨q1, q2, q3, q4, q5, q6⟩ :
> qt := ⟨q1(t), q2(t), q3(t), q4(t), q5(t), q6(t)⟩ :
> Dqt := map(diff, qt, t) :
>
> #Defining manipulator parametrs

> #Link Masses
> m1 := 27 :
> m2 := 22 :
> m3 := 0 :
> m4 := 25 :
> m5 := 0 :
> m6 := 1 :

> #Cylinder link dimentions parametrs
> #Radius
> r1 := 0.191 :
> r2 := 0.151 :
> r3 := 0 :
> r4 := 0.115 :
> r5 := 0 :
> r6 := 0.044 :
> #Height
> h1 := 0.363 :
> h2 := 0.515 :
> h3 := 0 :
> h4 := 0.583 :
> h5 := 0 :
> h6 := 0.107 :
>
> #Twists
> ξ1 := ⟨0, -0.070, 0, 0, 0, 1⟩ :
> ξ2 := ⟨-0.352, 0, 0.070, 0, 1, 0⟩ :
> ξ3 := ⟨-0.712, 0, 0.070, 0, 1, 0⟩ :
> ξ4 := ⟨0, 0.712, 0.070, 1, 0, 0⟩ :
> ξ5 := ⟨-0.712, 0, 45, 0, 1, 0⟩ :
```

```
> ξ6 := ⟨0, 0.712, 0, 1, 0, 0⟩ :
>
> #Constants
> a1 := 0.070 :
> e1 := 0.352 :
> e2 := 0.360 :
> e4 := 0.380 :
> e6 := 0.065 :
> r1c := 0.014 :
> r2c := 0.067 :
> r3c := 0.264 :
> r4c := -0.070 :
> r5c := 0.201 :
> r6c := 0.080 :
> r7c := 0.029 :

> #Gravity vector in inertial frame
> g0 := ⟨0, 0, g⟩ :

> #Rotation matricies
>
```

$$> R1 := \begin{bmatrix} \cos(q[1]) & -\sin(q[1]) & 0 \\ \sin(q[1]) & \cos(q[1]) & 0 \\ 0 & 0 & 1 \end{bmatrix} :$$

$$> R2 := \begin{bmatrix} \cos(q[2]) & 0 & \sin(q[2]) \\ 0 & 1 & 0 \\ -\sin(q[2]) & 0 & \cos(q[2]) \end{bmatrix} :$$

$$> R3 := \begin{bmatrix} \cos(q[3]) & 0 & \sin(q[3]) \\ 0 & 1 & 0 \\ -\sin(q[3]) & 0 & \cos(q[3]) \end{bmatrix} :$$

$$> R4 := \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(q[4]) & -\sin(q[4]) \\ 0 & \sin(q[4]) & \cos(q[4]) \end{bmatrix} :$$

$$> R5 := \begin{bmatrix} \cos(q[5]) & 0 & \sin(q[5]) \\ 0 & 1 & 0 \\ -\sin(q[5]) & 0 & \cos(q[5]) \end{bmatrix} :$$

```
> R6 := [ 1      0          0
        [ 0   cos(q[6])   -sin(q[6])
        [ 0   sin(q[6])    cos(q[6]) ] :
>
> R01 := R1 :
> R02 := MatrixMatrixMultiply(R01, R2) :
> R03 := MatrixMatrixMultiply(R02, R3) :
> R04 := MatrixMatrixMultiply(R03, R4) :
> R05 := MatrixMatrixMultiply(R04, R5) :
> R06 := MatrixMatrixMultiply(R05, R6) :
>
> #Adjoint transformation
> #A:=Matrix([[   I , 0 ,     0 ,   0 , 0 ,   0 ],
  #           [ A21 ,   I ,   0 ,   0 , 0 ,   0 ],
  #           [ A31 , A32 ,   I ,   0 , 0 ,   0 ],
  #           [ A41 , A42 , A43 ,   I , 0 ,   0 ],
  #           [ A51 , A52 , A53 , A54 , I ,   0 ],
  #           [ A61 , A62 , A63 , A64 , A65 , I ]]);
```

$$> a[1,1] := \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} :$$

```
>
```

$$> a[2,1] := [[\cos(2\cdot q[2]), 0, -\sin(2\cdot q[2]), 0, 2\cdot a1\cdot(\sin(q[2]))^2 - e1\cdot\sin(2\cdot q[2]) - 2\cdot a1 \cdot(\sin(2\cdot q[2]))^2 + e1\cdot\sin(4\cdot q[2]), 0],$$
$$[0, 1, 0, 2\cdot a1\cdot(\sin(q[2]))^2 - e1\cdot\sin(2\cdot q[2]), 0, 2\cdot e1\cdot((\cos(q[2]))^2 - 1) - a1\cdot\sin(2\cdot q[2])],$$
$$[\sin(2\cdot q[2]), 0, \cos(2\cdot q[2]), 0, 2\cdot e1\cdot(\sin(2\cdot q[2]))^2 - a1\cdot\sin(2\cdot q[2]) - 2\cdot e1\cdot(\sin(q[2]))^2 + a1\cdot\sin(4\cdot q[2]), 0],$$
$$[0, 0, 0, \cos(2\cdot q[2]), 0, -\sin(2\cdot q[2])],$$
$$[0, 0, 0, 0, 1, 0],$$
$$[0, 0, 0, \sin(2\cdot q[2]), 0, \cos(2\cdot q[2])]] :$$

```
> a[2,2] := Matrix([[1,0,0,0,0,0],[0,1,0,0,0,0],[0,0,1,0,0,0],[0,0,0,1,0,0],[0,0,0,0,1,0],[0,0,0,0,0,1]]):
>
> a[3,1] := [[cos(q[2]+q[3]), 0, -sin(q[2]+q[3]), 0, 2*e2*(sin(q[3]/2))^2 + 2*e1*(sin(q[2]/2 + q[3]/2))^2 - a1*sin(q[2]+q[3]), 0],
  [0, 1, 0, 2*e1*(sin(q[2]/2 + q[3]/2))^2 + 2*e2*(sin(q[2]/2 + q[3]/2))^2 - 2*e2*(sin(q[2]/2))^2 + a1*sin(q[2]+q[3]), 0, a1*cos(q[2]+q[3]) - a1 + e1*sin(q[2]+q[3]) + e2*sin(q[2]+q[3]) - e2*sin(q[2])],
  [sin(q[2]+q[3]), 0, cos(q[2]+q[3]), 0, a1*cos(q[2]+q[3]) - a1 - e1*sin(q[2]+q[3]) - e2*sin(q[3]), 0],
  [0, 0, 0, cos(q[2]+q[3]), 0, -sin(q[2]+q[3])],
  [0, 0, 0, 0, 1, 0],
  [0, 0, 0, sin(q[2]+q[3]), 0, cos(q[2]+q[3])]]:
> a[3,2] := [[cos(2*q[3]), 0, -sin(2*q[3]), 0, 2*sin(q[3])*(e1*sin(q[3]) - a1*cos(q[3]) + e2*sin(q[3])), 0],
  [0, 1, 0, a1*sin(2*q[3]) + 2*e1*(sin(q[3]))^2 + 2*e2*(sin(q[3]))^2, 0, 2*a1*(cos(q[3]))^2 - 2*a1 + e1*sin(2*q[3]) + e2*sin(2*q[3])],
  [sin(2*q[3]), 0, cos(2*q[3]), 0, 2*a1*(cos(q[3]))^2 - 2*a1 - e1*sin(2*q[3]) - e2*sin(2*q[3]), 0],
  [0, 0, 0, cos(2*q[3]), 0, -sin(2*q[3])],
  [0, 0, 0, 0, 1, 0],
  [0, 0, 0, sin(2*q[3]), 0, cos(2*q[3])]]:
> a[3,3] := Matrix([[1,0,0,0,0,0],[0,1,0,0,0,0],[0,0,1,0,0,0],[0,0,0,1,0,0],[0,0,0,0,1,0],[0,0,0,0,0,1]]):
```

$$a[4,1] := \left[\left[\cos(q[2]),\ 0,\ -\sin(q[2]),\ -\sin(q[2])\cdot\sin(q[4])\cdot(e1+e2),\ 2\cdot e1\cdot\left(\frac{\cos(q[4])}{2}-\frac{1}{2}\right)-2\cdot e1\cdot\left(\frac{\cos(q[2])}{2}-\frac{1}{2}\right)+2\cdot e2\cdot\left(\frac{\cos(q[4])}{2}-\frac{1}{2}\right)-a1\cdot\sin\left(q\left[2\right]\right),\ -\cos(q[2])\cdot\sin(q[4])\cdot(e1+e2)\right],\right.$$

$$\left[\sin(q[2])\cdot\sin(q[4]),\ \cos(q[4]),\ \cos(q[2])\cdot\sin(q[4]),\ e1\cdot\cos(q[4])+a1\cdot\cos(q[4])\cdot\sin(q[2])+e2\cdot\cos(q[2])\cdot\cos(q[4])-e1\cdot\cos(2\cdot q[4])\cdot\cos(q[2])-e2\cdot\cos(2\cdot q[4])\cdot\cos(q[2]),\ -\sin\left(q\left[4\right]\right)\cdot\left(2\cdot a1\cdot\left(\sin\left(\frac{q[2]}{2}\right)\right)^2+e1\cdot\sin\left(q\left[2\right]\right)\right),\ a1\cdot\cos(q[2])\cdot\cos(q[4])-e1\cdot\sin(q[2])-e2\cdot\sin(q[2])-a1\cdot\cos(q[4])-e2\cdot\cos(q[4])\cdot\sin(q[2])+2\cdot e1\cdot(\cos(q[4]))^2\cdot\sin(q[2])+2\cdot e2\cdot(\cos(q[4]))^2\cdot\sin(q[2])\right],$$

$$\left[\cos(q[4])\cdot\sin(q[2]),\ -\sin(q[4]),\ \cos(q[2])\cdot\cos(q[4]),\ -\sin(q[4])\cdot(e1+a1\cdot\sin(q[2])+e2\cdot\cos(q[2])-2\cdot e1\cdot\cos(q[2])\cdot\cos(q[4])-2\cdot e2\cdot\cos(q[2])\cdot\cos(q[4])),\ \left(2\cdot a1\cdot\left(\sin\left(\frac{q[2]}{2}\right)\right)^2+e1\cdot\sin\left(q\left[2\right]\right)\right)\cdot\left(2\cdot\left(\sin\left(\frac{q[4]}{2}\right)\right)^2-1\right),\ -\sin(q[4])\cdot(a1\cdot\cos(q[2])-a1-e2\cdot\sin(q[2])+2\cdot e1\cdot\cos(q[4])\cdot\sin(q[2])+2\cdot e2\cdot\cos(q[4])\cdot\sin(q[2]))\right],$$

$$\left[0,\ 0,\ 0,\ \cos(q[2]),\ 0,\ -\sin(q[2])\right],$$

$$\left[0,\ 0,\ 0,\ \sin(q[2])\cdot\sin(q[4]),\ \cos(q[4]),\ \cos(q[2])\cdot\sin(q[4])\right],$$

$$\left.\left[0,\ 0,\ 0,\ \cos(q[4])\cdot\sin(q[2]),\ -\sin(q[4]),\ \cos(q[2])\cdot\cos(q[4])\right]\right]:$$

$$a[4,2] := [[\cos(q[3]),\ 0,\ -\sin(q[3]),\ -\sin(q[3])\cdot\sin(q[4])\cdot(e1+e2),\ e1\cdot\cos(q[4])-e1\cdot\cos(q[3])-a1\cdot\sin(q[3])-e2\cdot\cos(q[3])+e2\cdot\cos(q[4]),\ -\cos(q[3])\cdot\sin(q[4])\cdot(e1+e2)],$$

$$\left[\sin(q[3])\cdot\sin(q[4]),\ \cos(q[4]),\ \cos(q[3])\cdot\sin(q[4]),\ e1\cdot\cos(q[4])+e2\cdot\cos(q[4])+a1\cdot\cos(q[4])\cdot\sin(q[3])-e1\cdot\cos(2\cdot q[4])\cdot\cos(q[3])-e2\cdot\cos(2\cdot q[4])\cdot\cos(q[3]),\ -\sin\left(q\left[4\right]\right)\cdot\left(2\cdot a1\cdot\left(\sin\left(\frac{q[3]}{2}\right)\right)^2+e1\cdot\sin\left(q\left[3\right]\right)+e2\cdot\sin\left(q\left[3\right]\right)\right),\ a1\cdot\cos(q[3])\cdot\cos(q[4])-e1\cdot\sin(q3)-e2\cdot\sin(q[3])-a1\cdot\cos(q[4])+2\cdot e1\cdot(\cos(q[4]))^2\cdot\sin(q[3])+2\cdot e2\cdot(\cos(q[4]))^2\cdot\sin(q[3])\right],$$

$$\left[\cos(q[4])\cdot\sin(q[3]),\ -\sin(q[4]),\ \cos(q[3])\cdot\cos(q[4]),\ -\sin(q[4])\cdot(e1+e2+a1\cdot\sin(q[3])-2\cdot e1\cdot\cos(q[3])\cdot\cos(q[4])-2\cdot e2\cdot\cos(q[3])\cdot\cos(q[4])),\ \left(2\cdot\left(\sin\left(\frac{q[4]}{2}\right)\right)^2-1\right.\right.$$

$$\Big)\cdot\left(2\cdot a1\cdot\left(\sin\left(\frac{q[3]}{2}\right)\right)^2+e1\cdot\sin\left(q\left[3\right]\right)+e2\cdot\sin\left(q\left[3\right]\right)\right),\ -\sin(q[4])\cdot(a1\cdot\cos(q[3])$$
$$-a1+2\cdot e1\cdot\cos(q[4])\cdot\sin(q[3])+2\cdot e2\cdot\cos(q[4])\cdot\sin(q[3]))\Big],$$
$$\left[0,\ 0,\ 0,\ \cos(q[3]),\ 0,\ -\sin(q[3])\right],$$
$$\left[0,\ 0,\ 0,\ \sin(q[3])\cdot\sin(q[4]),\ \cos(q[4]),\ \cos(q[3])\cdot\sin(q[4])\right],$$
$$\left[0,\ 0,\ 0,\ \cos(q[4])\cdot\sin(q[3]),\ -\sin(q[4]),\ \cos(q[3])\cdot\cos(q[4])\right]\Big]:$$

> $a[4,3]:=[[1,0,0,0,2\cdot(e1+e2)\cdot((\cos(q[4]))^2-1),\ -\sin(2\cdot q[4])\cdot(e1+e2)],$
$[0,\cos(2\cdot q[4]),\sin(2\cdot q[4]),(\cos(2\cdot q[4])-\cos(4\cdot q[4]))\cdot(e1+e2),0,0],$
$[0,\ -\sin(2\cdot q[4]),\cos(2\cdot q[4]),\ -(\sin(2\cdot q[4])-\sin(4\cdot q[4]))\cdot(e1+e2),0,0],$
$[0,0,0,1,0,0],$
$[0,0,0,0,\cos(2\cdot q[4]),\sin(2\cdot q[4])],$
$[0,0,0,0,\ -\sin(2\cdot q[4]),\cos(2\cdot q[4])]]:$

> $a[4,4]:=\begin{bmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\end{bmatrix}:$

>

> $a[5,1]:=\Big[\Big[\cos(q[2]+q[5]),\ 0,\ -\sin(q[2]+q[5]),\ 0,\ 2\cdot e2\cdot\left(\sin\left(\frac{q[5]}{2}\right)\right)^2+2\cdot e1\cdot\left(\sin\left(\frac{q[2]}{2}+\frac{q[5]}{2}\right)\right)^2-a1\cdot\sin\left(q\left[2\right]+q\left[5\right]\right)-e4\cdot\sin\left(q\left[5\right]\right),0\Big],$
$\left[0,1,0,2\cdot e1\cdot\left(\sin\left(\frac{q[2]}{2}+\frac{q[5]}{2}\right)\right)^2+2\cdot e2\cdot\left(\sin\left(\frac{q[2]}{2}+\frac{q[5]}{2}\right)\right)^2-2\cdot e2\cdot\left(\sin\left(\frac{q[2]}{2}\right)\right)^2+a1\cdot\sin\left(q\left[2\right]+q\left[5\right]\right)+e4\cdot\sin\left(q\left[2\right]+q\left[5\right]\right)-e4\cdot\sin\left(q\left[2\right]\right),0,a1\cdot\cos(q[2]+q[5])-a1+e4\cdot\cos(q[2]+q[5])+e1\cdot\sin(q[2]+q[5])+e2\cdot\sin(q[2]+q[5])-e4\cdot\cos(q[2])-e2\cdot\sin(q[2])\right],$
$[\sin(q[2]+q[5]),0,\cos(q[2]+q[5]),0,a1\cdot\cos(q[2]+q[5])-e4-a1-e1\cdot\sin(q[2]+q[5])+e4\cdot\cos(q[5])-e2\cdot\sin(q[5]),0],$
$\left[0,\ 0,\ 0,\ \cos(q[2]+q[5]),\ 0,\ -\sin(q[2]+q[5])\right],$
$\left[0,0,0,0,1,0\right],$
$\left[0,\ 0,\ 0,\ \sin(q[2]+q[5]),\ 0,\ \cos(q[2]+q[5])\right]\Big]:$

$$
\begin{aligned}
&> a[5,2] := \Bigg[\Bigg[\cos(q[3]+q[5]),\ 0,\ -\sin(q[3]+q[5]),\ 0,\ 2\cdot e1\cdot\left(\sin\left(\frac{q[3]}{2}+\frac{q[5]}{2}\right)\right)^2+2\\
&\quad\cdot e2\cdot\left(\sin\left(\frac{q[3]}{2}+\frac{q[5]}{2}\right)\right)^2-a1\cdot\sin\left(q\left[3\right]+q\left[5\right]\right)-e4\cdot\sin\left(q\left[5\right]\right),\ 0\Bigg],\\
&\quad\Bigg[0,\ 1,\ 0,\ 2\cdot e1\cdot\left(\sin\left(\frac{q[3]}{2}+\frac{q[5]}{2}\right)\right)^2+2\cdot e2\cdot\left(\sin\left(\frac{q[3]}{2}+\frac{q[5]}{2}\right)\right)^2+a1\cdot\sin\left(q\left[3\right]\\
&\quad+q\left[5\right]\right)+e4\cdot\sin\left(q\left[3\right]+q\left[5\right]\right)-e4\cdot\sin\left(q\left[3\right]\right),\ 0,\ a1\cdot\cos(q[3]+q[5])-a1+e4\cdot\cos(q[3\\
&\quad]+q[5])+e1\cdot\sin(q[3]+q[5])+e2\cdot\sin(q[3]+q[5])-e4\cdot\cos(q[3])\Bigg],\\
&\quad[\sin(q[3]+q[5]),\ 0,\ \cos(q[3]+q[5]),\ 0,\ a1\cdot\cos(q[3]+q[5])-e4-a1-e1\cdot\sin(q[3]\\
&\quad+q[5])-e2\cdot\sin(q[3]+q[5])+e4\cdot\cos(q[5]),\ 0],\\
&\quad\Bigg[0,\ 0,\ 0,\ \cos(q[3]+q[5]),\ 0,\ -\sin(q[3]+q[5])\Bigg],\\
&\quad\Bigg[0,\ 0,\ 0,\ 0,\ 2,\ 0\Bigg],\\
&\quad\Bigg[0,\ 0,\ 0,\ \sin(q[3]+q[5]),\ 0,\ \cos(q[3]+q[5])\Bigg]\Bigg]:
\end{aligned}
$$

$$
\begin{aligned}
&> a[5,3] := [[\cos(q[5]),\ \sin(q[4])\cdot\sin(q[5]),\ -\cos(q[4])\cdot\sin(q[5]),\ \sin(q[4])\cdot(2\cdot a1\cdot\cos(q\\
&\quad[4])+2\cdot e4\cdot\cos(q[4])+e1\cdot\sin(q[5])+e2\cdot\sin(q[5])-2\cdot a1\cdot\cos(q[4])\cdot(\cos(q[5]))^2\\
&\quad-2\cdot e4\cdot\cos(q[4])\cdot(\cos(q[5]))^2-2\cdot e1\cdot\cos(q[4])\cdot\cos(q[5])\cdot\sin(q[5])-2\cdot e2\cdot\cos(q[4]\\
&\quad)\cdot\cos(q[5])\cdot\sin(q[5])),\ e1\cdot\cos(q[4])-e1\cdot\cos(q[5])+e2\cdot\cos(q[4])-e2\cdot\cos(q[5])-a1\\
&\quad\cdot\cos(q[4])\cdot\sin(q[5])-e4\cdot\cos(q[4])\cdot\sin(q[5]),\ \sin(q[4])\cdot(e1\cdot(2\cdot(\sin(q[5]))^2-1)\\
&\quad+e2\cdot(2\cdot(\sin(q[5]))^2-1)+a1\cdot\sin(2\cdot q[5])+e4\cdot\sin(2\cdot q[5])-a1\cdot\sin(q[5])-e4\cdot\sin(q\\
&\quad[5]))],\\
&\quad\Bigg[0,\ \cos(q[4]),\ \sin(q[4]),\ e1\cdot\cos(q[4])+e2\cdot\cos(q[4])+a1\cdot\cos(2\cdot q[4])\cdot\sin(q[5])-e1\\
&\quad\cdot\cos(2\cdot q[4])\cdot\cos(q[5])-e2\cdot\cos(2\cdot q[4])\cdot\cos(q[5])+e4\cdot\cos(2\cdot q[4])\cdot\sin(q[5]),\ -\sin\left(q\right.\\
&\quad\left.\left[4\right]\right)\cdot\left(2\cdot a1\cdot\left(\frac{\cos(q[5])}{2}-\frac{1}{2}\right)+2\cdot e4\cdot\left(\frac{\cos(q[5])}{2}-\frac{1}{2}\right)+e1\cdot\sin\left(q\left[5\right]\right)+e2\cdot\sin\left(q\right.\right.\\
&\quad\left.\left.\left[5\right]\right)\right),\ \cos(q[4])\cdot(a1\cdot\cos(q[5])-e4-a1+e4\cdot\cos(q[5])+e1\cdot\sin(q[5])+e2\cdot\sin(q[5])\\
&\quad)\Bigg],\\
&\quad[\sin(q[5]),\ -\cos(q[5])\cdot\sin(q[4]),\ \cos(q[4])\cdot\cos(q[5]),\ -\cos(q[5])\cdot\sin(q[4])\cdot(e1+e2\\
&\quad+2\cdot a1\cdot\cos(q[4])\cdot\sin(q[5])-2\cdot e1\cdot\cos(q[4])\cdot\cos(q[5])-2\cdot e2\cdot\cos(q[4])\cdot\cos(q[5])+2\\
&\quad\cdot e4\cdot\cos(q[4])\cdot\sin(q[5])),\ a1\cdot\cos(q[4])\cdot\cos(q[5])-e4\cdot\cos(q[4])-e1\cdot\sin(q[5])-e2\cdot\sin\\
&\quad(q[5])-a1\cdot\cos(q[4])+e4\cdot\cos(q[4])\cdot\cos(q[5]),\ -\sin(q[4])\cdot(a1\cdot\cos(2\cdot q[5])+e4\cdot\cos\\
&\quad(2\cdot q[5])+e1\cdot\sin(2\cdot q[5])+e2\cdot\sin(2\cdot q[5])-a1\cdot\cos(q[5])-e4\cdot\cos(q[5]))],\\
&\quad\Bigg[0,\ 0,\ 0,\ \cos(q[5]),\ \sin(q[4])\cdot\sin(q[5]),\ -\cos(q[4])\cdot\sin(q[5])\Bigg],\\
&\quad\Bigg[0,\ 0,\ 0,\ 0,\ \cos(q[4]),\ \sin(q[4])\Bigg],\\
&\quad\Bigg[0,\ 0,\ 0,\ \sin(q[5]),\ -\cos(q[5])\cdot\sin(q[4]),\ \cos(q[4])\cdot\cos(q[5])\Bigg]\Bigg]:
\end{aligned}
$$

$$a[5,4] := \left[\left[\cos(2\cdot q[5]),\ 0,\ -\sin(2\cdot q[5]),\ 0,\ 2\cdot\sin\left(q\left[5\right]\right)\cdot\left(a1\cdot\left(2\cdot\left(\sin\left(\frac{q[5]}{2}\right)\right)^2-1\right)+e4\cdot\left(2\cdot\left(\sin\left(\frac{q[5]}{2}\right)\right)^2-1\right)+e1\cdot\sin\left(q\left[5\right]\right)+e2\cdot\sin\left(q\left[5\right]\right)\right),\ 0\right],\right.$$

$$\left[0,\ 1,\ 0,\ a1\cdot\sin(2\cdot q[5])+e4\cdot\sin(2\cdot q[5])+2\cdot e1\cdot(\sin(q[5]))^2+2\cdot e2\cdot(\sin(q[5]))^2,\ 0,\right.$$

$$\left.e1\cdot\sin(2\cdot q[5])+e2\cdot\sin(2\cdot q[5])+2\cdot a1\cdot((\cos(q[5]))^2-1)+2\cdot e4\cdot((\cos(q[5]))^2-1)\right],$$

$$[\sin(2\cdot q[5]),\ 0,\ \cos(2\cdot q[5]),\ 0,\ 2\cdot a1\cdot((\cos(q[5]))^2-1)-e2\cdot\sin(2\cdot q[5])-e1\cdot\sin(2\cdot q[5])+2\cdot e4\cdot((\cos(q[5]))^2-1),\ 0],$$

$$\left[0,\ 0,\ 0,\ \cos(2\cdot q[5]),\ 0,\ -\sin(2\cdot q[5])\right],$$

$$\left[0,\ 0,\ 0,\ 0,\ 1,\ 0\right],$$

$$\left.\left[0,\ 0,\ 0,\ \sin(2\cdot q[5]),\ 0,\ \cos(2\cdot q[5])\right]\right]:$$

$$a[5,5] := \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}:$$

>

$$a[6,1] := \left[\left[\cos(q[2]),\ 0,\ -\sin(q[2]),\ -\sin(q[2])\cdot\sin(q[6])\cdot(e1+e2),\ 2\cdot e1\cdot\left(\frac{\cos(q[6])}{2}-\frac{1}{2}\right)-2\cdot e1\cdot\left(\frac{\cos(q[2])}{2}-\frac{1}{2}\right)+2\cdot e2\left(\frac{\cos(q[6])}{2}-\frac{1}{2}\right)-a1\cdot\sin\left(q\left[2\right]\right),\ -\cos(q[2])\cdot\sin(q[6])\cdot(e1+e2)\right],\right.$$

$$\left[\sin(q[2])\cdot\sin(q[6]),\ \cos(q[6]),\ \cos(q[2])\cdot\sin(q[6]),\ e1\cdot\cos(q[6])+a1\cdot\cos(q[6])\cdot\sin(q[2])+e2\cdot\cos(q[2])\cdot\cos(q[6])-e1\cdot\cos(2\cdot q[6])\cdot\cos(q[2])-e2\cdot(2\cdot q[6])\cdot\cos(q[2]),\right.$$

$$-\sin\left(q\left[6\right]\right)\cdot\left(2\cdot a1\cdot\left(\sin\left(\frac{q[2]}{2}\right)\right)^2+e1\cdot\sin\left(q\left[2\right]\right)\right),\ a1\cdot\cos(q[2])\cdot\cos(q[6])-e1\cdot\sin(q[2])-e2\cdot\sin(q[2])-a1\cdot\cos(q[6])-e2\cdot\cos(q[6])\cdot\sin(q[2])+2\cdot e1\cdot(\cos(q[6]))^2\cdot\sin(q[2])+2\cdot e2\cdot(\cos(q[6]))^2\cdot\sin(q[2])\left.\right],$$

$$\left[\cos(q[6])\cdot\sin(q[2]),\ -\sin(q[6]),\ \cos(q[2])\cdot\cos(q[6]),\ -\sin(q[6])\cdot(e1+a1\cdot\sin(q[2])+e2\cdot\cos(q[2])-2\cdot e1\cdot\cos(q[2])\cdot\cos(q[6])-2\cdot e2\cdot\cos(q[2])\cdot\cos(q[6])),\ \left(2\cdot a1\cdot\left(\sin\right.\right.\right.$$

$$\left(\frac{q[2]}{2}\right)\right)^2 + e1\cdot\sin\left(q\left[2\right]\right)\right)\cdot\left(2\cdot\left(\sin\left(\frac{q[6]}{2}\right)\right)^2 - 1\right),\ -\sin(q[6])\cdot(a1\cdot\cos(q[2]) - a1$$
$$- e2\cdot\sin(q[2]) + 2\cdot e1\cdot\cos(q[6])\cdot\sin(q[2]) + 2\cdot e2\cdot\cos(q[6])\cdot\sin(q[2]))\Big],$$
$$\left[0, 0, 0, \cos(q[2]), 0, -\sin(q[2])\right],$$
$$\left[0, 0, 0, \sin(q[2])\cdot\sin(q[6]), \cos(q[6]), \cos(q[2])\cdot\sin(q[6])\right],$$
$$\left[0, 0, 0, \cos(q[6])\cdot\sin(q[2]), -\sin(q[6]), \cos(q[2])\cdot\cos(q[6])\right]\Big]:$$

> $a[6,2] := [[\cos(q[3]), 0, -\sin(q[3]), -\sin(q[3])\cdot\sin(q[6])\cdot(e1 + e2), e1\cdot\cos(q[6]) - e1$
$\cdot\cos(q[3]) - e2\cdot\cos(q[3]) - a1\cdot\sin(q[3]) + e2\cdot\cos(q[6]), -\cos(q[3])\cdot\sin(q[6])\cdot(e1 + e2$
$)],$
$$\left[\sin(q[3])\cdot\sin(q[6]), \cos(q[6]), \cos(q[3])\cdot\sin(q[6]), e1\cdot\cos(q[6]) + e2\cdot\cos(q[6]) + a1\right.$$
$$\left.\cdot\cos(q[6])\cdot\sin(q[3]) - e1\cdot\cos(2\cdot q[6])\cdot\cos(q[3]) - e2\cdot\cos(2\cdot q[6])\cdot\cos(q[3]), -\sin\left(q\left[6\right]\right.\right.$$
$$\left.\left.\right)\cdot\left(2\cdot a1\cdot\left(\sin\left(\frac{q[3]}{2}\right)\right)^2 + e1\cdot\sin\left(q\left[3\right]\right) + e2\cdot\sin\left(q\left[3\right]\right)\right), a1\cdot\cos(q[3])\cdot\cos(q[6])\right.$$
$$\left. - e1\cdot\sin(q[3]) - e2\cdot\sin(q[3]) - a1\cdot\cos(q[6]) + 2\cdot e1\cdot(\cos(q[6]))^2\cdot\sin(q[3]) + 2\cdot e2\right.$$
$$\left.\cdot(\cos(q[6]))^2\cdot\sin(q[3])\right],$$
$$\left[\cos(q[6])\cdot\sin(q[3]), -\sin(q[6]), \cos(q[3])\cdot\cos(q[6]), -\sin(q[6])\cdot(e1 + e2 + a1\cdot\sin\right.$$
$$\left.(q[3]) - 2\cdot e1\cdot\cos(q[3])\cdot\cos(q[6]) - 2\cdot e2\cdot\cos(q[3])\cdot\cos(q[6])), \left(2\cdot\left(\sin\left(\frac{q[6]}{2}\right)\right)^2 - 1\right.\right.$$
$$\left.\left.\right)\cdot\left(2\cdot a1\cdot\left(\sin\left(\frac{q[3]}{2}\right)\right)^2 + e1\cdot\sin\left(q\left[3\right]\right) + e2\cdot\sin\left(q\left[3\right]\right)\right), -\sin(q[6])\cdot(a1\cdot\cos(q[3])\right.$$
$$\left. - a1 + 2\cdot e1\cdot\cos(q[6])\cdot\sin(q[3]) + 2\cdot e2\cdot\cos(q[6])\cdot\sin(q[3]))\right],$$
$$\left[0, 0, 0, \cos(q[3]), 0, -\sin(q[3])\right],$$
$$\left[0, 0, 0, \sin(q[3])\cdot\sin(q[6]), \cos(q[6]), \cos(q[3])\cdot\sin(q[6])\right],$$
$$\left[0, 0, 0, \cos(q[6])\cdot\sin(q[3]), -\sin(q[6]), \cos(q[3])\cdot\cos(q[6])\right]\Big]:$$

> $a[6,3] := [[[1, 0, 0, 0, (\cos(q[4]+q[6]) - 1)\cdot(e1 + e2), -\sin(q[4]+q[6])\cdot(e1 + e2)],$
$[0, \cos(q[4]+q[6]), \sin(q[4]+q[6]), (e1 + e2)\cdot(-2\cdot(\cos(q[4]+q[6]))^2 + \cos(q[4]$
$+ q[6]) + 1), 0, 0],$
$[0, -\sin(q[4]+q[6]), \cos(q[4]+q[6]), (\sin(2\cdot q[4]+2\cdot q[6]) - \sin(q[4]+q[6]))\cdot(e1$
$+ e2), 0, 0],$
$[0, 0, 0, 1, 0, 0],$
$[0, 0, 0, 0, \cos(q[4]+q[6]), \sin(q[4]+q[6])],$
$[0, 0, 0, 0, -\sin(q[4]+q[6]), \cos(q[4]+q[6])]]:$

> $a[6,4] := [[\cos(q[5]), 0, -\sin(q[5]), -\sin(q[5])\cdot\sin(q[6])\cdot(e1 + e2), e1\cdot\cos(q[6]) - e1$

$$\cdot\cos(q[5]) - a1\cdot\sin(q[5]) - e2\cdot\cos(q[5]) + e2\cdot\cos(q[6]) - e4\cdot\sin(q[5]),\ -\cos(q[5])\cdot\sin(q[6])\cdot(e1+e2)\Big],$$
$$\Big[\sin(q[5])\cdot\sin(q[6]),\ \cos(q[6]),\ \cos(q[5])\cdot\sin(q[6]),\ e1\cdot\cos(q[6]) + e2\cdot\cos(q[6]) + a1\cdot\cos(q[6])\cdot\sin(q[5]) + e4\cdot\cos(q[6])\cdot\sin(q[5]) - e1\cdot\cos(2\cdot q[6])\cdot\cos(q[5]) - e2\cdot\cos(2\cdot q[6])\cdot\cos(q[5]),\ -\sin(q[6])\cdot(a1 + e4 - a1\cdot\cos(q[5]) - e4\cdot\cos(q[5]) + e1\cdot\sin(q[5]) + e2\cdot\sin(q[5]))\Big],$$
$$\Big[\cos(q[6])\cdot\sin(q[5]),\ -\sin(q[6]),\ \cos(q[5])\cdot\cos(q[6]),\ -\sin(q[6])\cdot(e1 + e2 + a1\cdot\sin(q[5]) + e4\cdot\sin(q[5]) - 2\cdot e1\cdot\cos(q[5])\cdot\cos(q[6]) - 2\cdot e2\cdot\cos(q[5])\cdot\cos(q[6])),\ \Big(2\cdot\Big(\sin\Big(\frac{q[6]}{2}\Big)\Big)^2 - 1\Big)\cdot\Big(2\cdot a1\cdot\Big(\sin\Big(\frac{q[5]}{2}\Big)\Big)^2 + 2\cdot e4\cdot\Big(\sin\Big(\frac{q[5]}{2}\Big)\Big)^2 + e1\cdot\sin\Big(q\Big[5\Big]\Big) + e2\cdot\sin\Big(q\Big[5\Big]\Big)\Big),\ -\sin(q[6])\cdot(a1\cdot\cos(q[5]) - e4 - a1 + e4\cdot\cos(q[5]) + 2\cdot e1\cdot\cos(q[6])\cdot\sin(q[5]) + 2\cdot e2\cdot\cos(q[6])\cdot\sin(q[5]))\Big],$$
$$\Big[0,\ 0,\ 0,\ \cos(q[5]),\ 0,\ -\sin(q[5])\Big],$$
$$\Big[0,\ 0,\ 0,\ \sin(q[5])\cdot\sin(q[6]),\ \cos(q[6]),\ \cos(q[5])\cdot\sin(q[6])\Big],$$
$$\Big[0,\ 0,\ 0,\ \cos(q[6])\cdot\sin(q[5]),\ -\sin(q[6]),\ \cos(q[5])\cdot\cos(q[6])\Big]\Big]:$$

> $a[6,5] := [[1, 0, 0, 0, 2\cdot(e1+e2)\cdot((\cos(q[6]))^2 - 1),\ -\sin(2\cdot q[6])\cdot(e1+e2)],$
$[0, \cos(2\cdot q[6]), \sin(2\cdot q[6]), (\cos(2\cdot q[6]) - \cos(4\cdot q[6]))\cdot(e1+e2), 0, 0],$
$[0,\ -\sin(2\cdot q[6]), \cos(2\cdot q[6]),\ -(\sin(2\cdot q[6]) - \sin(4\cdot q[6]))\cdot(e1+e2), 0, 0],$
$[0, 0, 0, 1, 0, 0],$
$[0, 0, 0, 0, \cos(2\cdot q[6]), \sin(2\cdot q[6])],$
$[0, 0, 0, 0,\ -\sin(2\cdot q[6]), \cos(2\cdot q[6])]]:$

> $a[6,6] := \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}:$

>

> $A := \begin{bmatrix} a_{1,1} & 0 & 0 & 0 & 0 & 0 \\ a_{2,1} & a_{2,2} & 0 & 0 & 0 & 0 \\ a_{3,1} & a_{3,2} & a_{3,3} & 0 & 0 & 0 \\ a_{4,1} & a_{4,2} & a_{4,3} & a_{4,4} & 0 & 0 \\ a_{5,1} & a_{5,2} & a_{5,3} & a_{5,4} & a_{5,5} & 0 \\ a_{6,1} & a_{6,2} & a_{6,3} & a_{6,4} & a_{6,5} & a_{6,6} \end{bmatrix}:$

>

>

> **#Transformed inertia matrices for the ith link**

>

> $M01 := \left[\left[m1, 0, 0, 0, m1 \cdot r3c, m1 \cdot r2c\right], \left[0, m1, 0, -m1 \cdot r3c, 0, m1 \cdot r1c\right], \left[0, 0, m1, -m1 \cdot r2c, -m1 \cdot r1c, 0\right], \left[0, -m1 \cdot r3c, -m1 \cdot r2c, \frac{1}{12} \cdot m1 \cdot \left(3 \cdot r1^2 + h1^2\right), 0, 0\right], \left[m1 \cdot r3c, 0, -m1 \cdot r1c, 0, \frac{1}{12} \cdot m1 \cdot \left(3 \cdot r1^2 + h1^2\right), 0\right], \left[m1 \cdot r2c, m1 \cdot r1c, 0, 0, 0, \frac{1}{2} \cdot m1 \cdot r1^2\right]\right]:$

>

> $M02 := \left[\left[m2, 0, 0, 0, m1 \cdot (e1 + r5c), m1 \cdot r4c\right], \left[0, m2, 0, -m1 \cdot (e1 + r5c), 0, 0\right], \left[0, 0, m2, -m1 \cdot r4c, 0, 0\right], \left[0, -m1 \cdot (e1 + r5c), -m1 \cdot r4c, \frac{1}{12} \cdot m2 \cdot \left(3 \cdot r2^2 + h2^2\right), 0, 0\right], \left[m1 \cdot (e1 + r5c), 0, 0, 0, \frac{1}{12} \cdot m2 \cdot \left(3 \cdot r2^2 + h2^2\right), 0\right], \left[m1 \cdot r4c, 0, 0, 0, 0, \frac{1}{2} \cdot m2 \cdot r2^2\right]\right]:$

>

$$> M03 := \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} :$$

$$> M04 := \left[\left[m4, 0, 0, 0, m4 \cdot (e1 + e2), 0 \right], \left[0, m4, 0, -m4 \cdot (e1 + e2), 0, m4 \cdot (a1 - r6c) \right], \left[0, 0, m4, 0, -m4 \cdot (a1 - r6c), 0 \right], \left[0, -m4 \cdot (e1 + e2), 0, \frac{1}{2} \cdot m4 \cdot r4^2, 0, 0 \right], \left[m4 \cdot (e1 + e2), 0, -m4 \cdot (a1 - r6c), 0, \frac{1}{12} \cdot m4 \cdot \left(3 \cdot r4^2 + h4^2 \right), 0 \right], \left[0, m4 \cdot (a1 - r6c), 0, 0, 0, \frac{1}{12} \cdot m4 \cdot \left(3 \cdot r4^2 + h4^2 \right) \right] \right] :$$

$$> M05 := \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} :$$

$$> M06 := \left[\left[m6, 0, 0, 0, m6 \cdot (e1 + e2), 0 \right], \left[0, m6, 0, -m6 \cdot (e1 + e2), 0, m6 \cdot (a1 + e4 + r7c) \right], \left[0, 0, m6, 0, -m6 \cdot (a1 + e4 + r7c), 0 \right], \left[0, -m6 \cdot (e1 + e2), 0, \frac{1}{2} \cdot m6 \cdot r6^2, 0, 0 \right], \left[m6 \cdot (e1 + e2), 0, -m6 \cdot (a1 + e4 + r7c), 0, \frac{1}{12} \cdot m6 \cdot \left(3 \cdot r6^2 + h6^2 \right), 0 \right], \left[0, m6 \cdot (a1 + e4 + r7c), 0, 0, 0, \frac{1}{12} \cdot m6 \cdot \left(3 \cdot r6^2 + h6^2 \right) \right] \right] :$$

$$> M0 := table([M01, M02, M03, M04, M05, M06]) :$$

```
> #Elements for dynamics equation
>
>  #elements of inertia matrix
>
> n[1,1]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ1^%T
      , a[l,1]^%T), M0[l]), a[l,1]), ξ1), 1 = max(1,1)..6) :
> n[1,2]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ1^%T
      , a[l,1]^%T), M0[l]), a[l,2]), ξ2), 1 = max(1,2)..6) :
> n[1,3]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ1^%T
      , a[l,1]^%T), M0[l]), a[l,3]), ξ3), 1 = max(1,3)..6) :
> n[1,4]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ1^%T
      , a[l,1]^%T), M0[l]), a[l,4]), ξ4), 1 = max(1,4)..6) :
> n[1,5]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ1^%T
      , a[l,1]^%T), M0[l]), a[l,5]), ξ5), 1 = max(1,5)..6) :
> n[1,6]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ1^%T
      , a[l,1]^%T), M0[l]), a[l,6]), ξ6), 1 = max(1,6)..6) :
>
> n[2,1]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ2^%T
      , a[l,2]^%T), M0[l]), a[l,1]), ξ1), 1 = max(2,1)..6) :
> n[2,2]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ2^%T
      , a[l,2]^%T), M0[l]), a[l,2]), ξ2), 1 = max(2,2)..6) :
> n[2,3]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ2^%T
      , a[l,2]^%T), M0[l]), a[l,3]), ξ3), 1 = max(2,3)..6) :
> n[2,4]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ2^%T
      , a[l,2]^%T), M0[l]), a[l,4]), ξ4), 1 = max(2,4)..6) :
> n[2,5]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ2^%T
      , a[l,2]^%T), M0[l]), a[l,5]), ξ5), 1 = max(2,5)..6) :
> n[2,6]
```

```
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ2^%T
      , a[l, 2]^%T), M0[l]), a[l, 6]), ξ6), 1 = max(2, 6) ..6) :
>
> n[3, 1]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ3^%T
      , a[l, 3]^%T), M0[l]), a[l, 1]), ξ1), 1 = max(3, 1) ..6) :
> n[3, 2]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ3^%T
      , a[l, 3]^%T), M0[l]), a[l, 2]), ξ2), 1 = max(3, 2) ..6) :
> n[3, 3]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ3^%T
      , a[l, 3]^%T), M0[l]), a[l, 3]), ξ3), 1 = max(3, 3) ..6) :
> n[3, 4]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ3^%T
      , a[l, 3]^%T), M0[l]), a[l, 4]), ξ4), 1 = max(3, 4) ..6) :
> n[3, 5]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ3^%T
      , a[l, 3]^%T), M0[l]), a[l, 5]), ξ5), 1 = max(3, 5) ..6) :
> n[3, 6]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ3^%T
      , a[l, 3]^%T), M0[l]), a[l, 6]), ξ6), 1 = max(3, 6) ..6) :
>
> n[4, 1]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ4^%T
      , a[l, 4]^%T), M0[l]), a[l, 1]), ξ1), 1 = max(4, 1) ..6) :
> n[4, 2]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ4^%T
      , a[l, 4]^%T), M0[l]), a[l, 2]), ξ2), 1 = max(4, 2) ..6) :
> n[4, 3]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ4^%T
      , a[l, 4]^%T), M0[l]), a[l, 3]), ξ3), 1 = max(4, 3) ..6) :
> n[4, 4]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ4^%T
      , a[l, 4]^%T), M0[l]), a[l, 4]), ξ4), 1 = max(4, 4) ..6) :
> n[4, 5]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ4^%T
      , a[l, 4]^%T), M0[l]), a[l, 5]), ξ5), 1 = max(4, 5) ..6) :
> n[4, 6]
```

```
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ4^%T
      , a[l, 4]^%T), M0[l]), a[l, 6]), ξ6), 1 = max(4, 6) ..6) :
>
> n[5, 1]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ5^%T
      , a[l, 5]^%T), M0[l]), a[l, 1]), ξ1), 1 = max(5, 1) ..6) :
> n[5, 2]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ5^%T
      , a[l, 5]^%T), M0[l]), a[l, 2]), ξ2), 1 = max(5, 2) ..6) :
> n[5, 3]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ5^%T
      , a[l, 5]^%T), M0[l]), a[l, 3]), ξ3), 1 = max(5, 3) ..6) :
> n[5, 4]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ5^%T
      , a[l, 5]^%T), M0[l]), a[l, 4]), ξ4), 1 = max(5, 4) ..6) :
> n[5, 5]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ5^%T
      , a[l, 5]^%T), M0[l]), a[l, 5]), ξ5), 1 = max(5, 5) ..6) :
> n[5, 6]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ5^%T
      , a[l, 5]^%T), M0[l]), a[l, 6]), ξ6), 1 = max(5, 6) ..6) :
>
> n[6, 1]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ6^%T
      , a[l, 6]^%T), M0[l]), a[l, 1]), ξ1), 1 = max(6, 1) ..6) :
> n[6, 2]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ6^%T
      , a[l, 6]^%T), M0[l]), a[l, 2]), ξ2), 1 = max(6, 2) ..6) :
> n[6, 3]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ6^%T
      , a[l, 6]^%T), M0[l]), a[l, 3]), ξ3), 1 = max(6, 3) ..6) :
> n[6, 4]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ6^%T
      , a[l, 6]^%T), M0[l]), a[l, 4]), ξ4), 1 = max(6, 4) ..6) :
> n[6, 5]
      := add(Multiply(VectorMatrixMultiply(VectorMatrixMultiply(VectorMatrixMultiply(ξ6^%T
      , a[l, 6]^%T), M0[l]), a[l, 5]), ξ5), 1 = max(6, 5) ..6) :
> n[6, 6]
```

$$:= add\left(Multiply\left(VectorMatrixMultiply\left(VectorMatrixMultiply\left(VectorMatrixMultiply\left(\xi 6^{\%T}, a[l,6]^{\%T}\right), M0[l]\right), a[l,6]\right), \xi 6\right), 1 = \max(6,6)..6\right):$$

```
>
> #Elements of Coriolis mstrix
```

$$> o[1,1] := add(((diff(n[1,1],q[k]) + diff(n[1,k],q[1]) - diff(n[k,1],q[1]))\cdot Dqt[k]), k = 1..6):$$

$$> o[1,2] := add(((diff(n[1,2],q[k]) + diff(n[1,k],q[2]) - diff(n[k,2],q[1]))\cdot Dqt[k]), k = 1..6):$$

$$> o[1,3] := add(((diff(n[1,3],q[k]) + diff(n[1,k],q[3]) - diff(n[k,3],q[1]))\cdot Dqt[k]), k = 1..6):$$

$$> o[1,4] := add(((diff(n[1,4],q[k]) + diff(n[1,k],q[4]) - diff(n[k,4],q[1]))\cdot Dqt[k]), k = 1..6):$$

$$> o[1,5] := add(((diff(n[1,5],q[k]) + diff(n[1,k],q[5]) - diff(n[k,5],q[1]))\cdot Dqt[k]), k = 1..6):$$

$$> o[1,6] := add(((diff(n[1,6],q[k]) + diff(n[1,k],q[6]) - diff(n[k,6],q[1]))\cdot Dqt[k]), k = 1..6):$$

```
>
```

$$> o[2,1] := add(((diff(n[2,1],q[k]) + diff(n[2,k],q[1]) - diff(n[k,1],q[2]))\cdot Dqt[k]), k = 1..6):$$

$$> o[2,2] := add(((diff(n[2,2],q[k]) + diff(n[2,k],q[2]) - diff(n[k,2],q[2]))\cdot Dqt[k]), k = 1..6):$$

$$> o[2,3] := add(((diff(n[2,3],q[k]) + diff(n[2,k],q[3]) - diff(n[k,3],q[2]))\cdot Dqt[k]), k = 1..6):$$

$$> o[2,4] := add(((diff(n[2,4],q[k]) + diff(n[2,k],q[4]) - diff(n[k,4],q[2]))\cdot Dqt[k]), k = 1..6):$$

$$> o[2,5] := add(((diff(n[2,5],q[k]) + diff(n[2,k],q[5]) - diff(n[k,5],q[2]))\cdot Dqt[k]), k = 1..6):$$

$$> o[2,6] := add(((diff(n[2,6],q[k]) + diff(n[2,k],q[6]) - diff(n[k,6],q[2]))\cdot Dqt[k]), k = 1..6):$$

```
>
```

$$> o[3,1] := add(((diff(n[3,1],q[k]) + diff(n[3,k],q[1]) - diff(n[k,1],q[3]))\cdot Dqt[k]), k = 1..6):$$

$$> o[3,2] := add(((diff(n[3,2],q[k]) + diff(n[3,k],q[2]) - diff(n[k,2],q[3]))\cdot Dqt[k]), k = 1..6):$$

$$> o[3,3] := add(((diff(n[3,3],q[k]) + diff(n[3,k],q[3]) - diff(n[k,3],q[3]))\cdot Dqt[k]), k = 1..6):$$

$$> o[3,4] := add(((diff(n[3,4],q[k]) + diff(n[3,k],q[4]) - diff(n[k,4],q[3]))\cdot Dqt[k]), k = 1..6):$$

$$> o[3,5] := add(((diff(n[3,5],q[k]) + diff(n[3,k],q[5]) - diff(n[k,5],q[3]))\cdot Dqt[k]), k = 1..6):$$

$$> o[3,6] := add(((diff(n[3,6],q[k]) + diff(n[3,k],q[6]) - diff(n[k,6],q[3]))\cdot Dqt[k]), k = 1..6):$$

```
>
```

$$> o[4,1] := add(((diff(n[4,1],q[k]) + diff(n[4,k],q[1]) - diff(n[k,1],q[4]))\cdot Dqt[k]), k = 1..6):$$

$$> o[4,2] := add(((diff(n[4,2],q[k]) + diff(n[4,k],q[2]) - diff(n[k,2],q[4]))\cdot Dqt[k]), k = 1..6):$$

```
> o[4,3] := add(((diff(n[4,3],q[k]) + diff(n[4,k],q[3]) - diff(n[k,3],q[4]))·Dqt[k]), k
      = 1..6):
> o[4,4] := add(((diff(n[4,4],q[k]) + diff(n[4,k],q[4]) - diff(n[k,4],q[4]))·Dqt[k]), k
      = 1..6):
> o[4,5] := add(((diff(n[4,5],q[k]) + diff(n[4,k],q[5]) - diff(n[k,5],q[4]))·Dqt[k]), k
      = 1..6):
> o[4,6] := add(((diff(n[4,6],q[k]) + diff(n[4,k],q[6]) - diff(n[k,6],q[4]))·Dqt[k]), k
      = 1..6):
>
> o[5,1] := add(((diff(n[5,1],q[k]) + diff(n[5,k],q[1]) - diff(n[k,1],q[5]))·Dqt[k]), k
      = 1..6):
> o[5,2] := add(((diff(n[5,2],q[k]) + diff(n[5,k],q[2]) - diff(n[k,2],q[5]))·Dqt[k]), k
      = 1..6):
> o[5,3] := add(((diff(n[5,3],q[k]) + diff(n[5,k],q[3]) - diff(n[k,3],q[5]))·Dqt[k]), k
      = 1..6):
> o[5,4] := add(((diff(n[5,4],q[k]) + diff(n[5,k],q[4]) - diff(n[k,4],q[5]))·Dqt[k]), k
      = 1..6):
> o[5,5] := add(((diff(n[5,5],q[k]) + diff(n[5,k],q[5]) - diff(n[k,5],q[5]))·Dqt[k]), k
      = 1..6):
> o[5,6] := add(((diff(n[5,6],q[k]) + diff(n[5,k],q[6]) - diff(n[k,6],q[5]))·Dqt[k]), k
      = 1..6):
>
> o[6,1] := add(((diff(n[6,1],q[k]) + diff(n[6,k],q[1]) - diff(n[k,1],q[6]))·Dqt[k]), k
      = 1..6):
> o[6,2] := add(((diff(n[6,2],q[k]) + diff(n[6,k],q[2]) - diff(n[k,2],q[6]))·Dqt[k]), k
      = 1..6):
> o[6,3] := add(((diff(n[6,3],q[k]) + diff(n[6,k],q[3]) - diff(n[k,3],q[6]))·Dqt[k]), k
      = 1..6):
> o[6,4] := add(((diff(n[6,4],q[k]) + diff(n[6,k],q[4]) - diff(n[k,4],q[6]))·Dqt[k]), k
      = 1..6):
> o[6,5] := add(((diff(n[6,5],q[k]) + diff(n[6,k],q[5]) - diff(n[k,5],q[6]))·Dqt[k]), k
      = 1..6):
> o[6,6] := add(((diff(n[6,6],q[k]) + diff(n[6,k],q[6]) - diff(n[k,6],q[6]))·Dqt[k]), k
      = 1..6):
>
> #Elements of gravity vector
>
> g1 := MatrixVectorMultiply((R01^%T), g0):
> g2 := combine(MatrixVectorMultiply((R02^%T), g0), trig):
> g3 := combine(MatrixVectorMultiply((R03^%T), g0), trig):
> g4 := combine(MatrixVectorMultiply((R04^%T), g0), trig):
> g5 := combine(MatrixVectorMultiply((R05^%T), g0), trig):
> g6 := combine(MatrixVectorMultiply((R06^%T), g0), trig):
> ##############################
> #Inertia matrix
```

> $M := \begin{bmatrix} n_{1,1} & n_{1,2} & n_{1,3} & n_{1,4} & n_{1,5} & n_{1,6} \\ n_{2,1} & n_{2,2} & n_{2,3} & n_{2,4} & n_{2,5} & n_{2,6} \\ n_{3,1} & n_{3,2} & n_{3,3} & n_{3,4} & n_{3,5} & n_{3,6} \\ n_{4,1} & n_{4,2} & n_{4,3} & n_{4,4} & n_{4,5} & n_{4,6} \\ n_{5,1} & n_{5,2} & n_{5,3} & n_{5,4} & n_{5,5} & n_{5,6} \\ n_{6,1} & n_{6,2} & n_{6,3} & n_{6,4} & n_{6,5} & n_{6,6} \end{bmatrix}$:

> **#Coriolis matrix**

> $C := \begin{bmatrix} o_{1,1} & o_{1,2} & o_{1,3} & o_{1,4} & o_{1,5} & o_{1,6} \\ o_{2,1} & o_{2,2} & o_{2,3} & o_{2,4} & o_{2,5} & o_{2,6} \\ o_{3,1} & o_{3,2} & o_{3,3} & o_{3,4} & o_{3,5} & o_{3,6} \\ o_{4,1} & o_{4,2} & o_{4,3} & o_{4,4} & o_{4,5} & o_{4,6} \\ o_{5,1} & o_{5,2} & o_{5,3} & o_{5,4} & o_{5,5} & o_{5,6} \\ o_{6,1} & o_{6,2} & o_{6,3} & o_{6,4} & o_{6,5} & o_{6,6} \end{bmatrix}$:

> **#Gravity vector**

> $G0 := \langle g1, g2, g3, g4, g5, g6 \rangle$:

> **#Matlab conversion**

> $with(CodeGeneration)$:

> **#Matrix M**

▶ **n11**

▶ **n12**

▶ **n13**

▶ **n14**

▶ **n15**

▶ **n16**

>

▶ **n21**

▶ **n22**

▶ **n23**

▶ **n24**

▶ **n25**

▶ **n26**

[>

▶ **n31**

▶ **n32**

▶ **n33**

▶ **n34**

▶ **n35**

▶ **n36**

[>

▶ **n41**

▶ **n42**

▶ **n43**

▶ **n44**

▶ **n45**

▶ **n46**

[>

▶ **n51**

▶ **n52**

▶ **n53**

▶ **n54**

▶ **n55**

▶ **n56**

```
>
```

▶ **n61**

▶ **n62**

▶ **n63**

▶ **n64**

▶ **n65**

▶ **n66**

```
> #Matrix C
```

▶ **o11**

▶ **o12**

▶ **o13**

▶ **o14**

▶ **o15**

▶ **o16**


o21
o22
o23
o24
o25
o26
o31
o32
o33
o34
o35
o36
o41
o42
o43
o44
o45

▶ **o46**

```
>
```

▶ **o51**

▶ **o52**

▶ **o53**

▶ **o54**

▶ **o55**

▶ **o56**

```
>
```

▶ **o61**

▶ **o62**

▶ **o63**

▶ **o64**

▶ **o65**

▶ **o66**

```
> #Gravity vector
```

▶ **g1**

▶ **g2**

▶ **g3**

▶ **g4**

▶ **g5**

▶ **g6**