Modeling and Control of Crane on Offshore Vessel

by
Pål Guttorm Syvertsen

NTNU
Norwegian University of
Science and Technology

June 8, 2011

Problem Description

Cranes on offshore vessels are used in various operations all over the world. Dependent on the operations, the cranes differ in size, lifting capacity, number of joints, working area, etc. and are specifically chosen due to the environment and working area. Operations spans from simple bulk handling to advanced subsea lifting operations in harsh weather. For ship owners, it is important to perform crane operations faster, in deeper waters and with increased weather operation window. In addition, necessary precautions towards safety to human lives, environment and property are to be taken, as demanded by classification societies. The objective of this master thesis is to investigate mathematical models and control methods for offshore cranes, and implement and test them in Simulink.

Scope of work:

1. Review necessary literature within the fields of offshore cranes and control strategies like input shaping.
2. Formulate a mathematical model of an given offshore crane with 3 joints and 4 links.
3. Implement the mathematical model in Simulink.
4. Implement a movable foundation so waves and other environmental disturbances can influence the crane.
5. Include a dynamic wire model as given by SINTEF in the simulation model.
6. Investigate and implement, if possible, a planar control algorithm as well as a oscillation control algorithm using e.g. input shaping.
7. Analyse and test stability and performance.
8. Conclude and make a plan for further work.

The report shall be written in English and edited as a research report including literature survey, description of mathematical models, description of control algorithms, simulation results, model test results, discussion and a conclusion including a proposal for further work. Source code should be provided on a CD with code listing enclosed in appendix. It is supposed that Department of Marine Technology, NTNU, can use the results freely in its research work, unless otherwise agreed upon, by referring to the students work. The thesis should be submitted in three copies within June 14th.

Assignment given: January 2011

Supervisor: Asgeir Sørensen (NTNU)
Cosupervisor: Karl Gunnar Aarsæther (SINTEF)
Thor Erling Grahl Nielsen (MacGregor)

Acknowledgements

I would like to thank Thor Erling Grahl Nielsen and Henrik Buestad for introducing me to crane control. They were both patient and helpful during my summer job at MacGregor. As a result they suggested that I could look into crane control at my master thesis. I would also like to thank Karl Gunnar Aarsæther and SINTEF for providing and helping me implement the cable model.

Last but not least, I would like to thank my supervisor Asgeir Sørensen. He has been helpful and always genuinely interested in my work.

Abstract

This master thesis is about modeling and control of crane placed on an offshore vessel. An active heave compensating crane, designed by MacGregor, has been used as a basis for developing the crane model. Cranes of this type are placed on a wide variety of vessels and operate all over the world. Besides MacGregor, several other well reputated crane suppliers design similar cranes. In addition, two inverse kinematics control algorithms and three input shapers have been described, implemented and tested in Simulink.

The model of the MacGregor crane was developed mathematically using robot modeling theory. The resulting model was implemented in Simulink and verified both mathematically and against a SimMechanics model. In addition, a cable model and vessel crane interaction were added to the Simulink model. The total Simulink model responded according to expectations during simulations.

Two inverse kinematics algorithms were implemented and tested. Both had vessel kinematics incorporated. As a result, these two algorithms enabled the crane to counteract the vessel motion. This control method is also of interest to operators. It enables the them to control the crane in the workspace, instead of in the joint space. Simulations show that the two inverse kinematics algorithms were able to keep the crane tip at a fixed point in an earth fixed reference frame. However, in sea state 6 the vessel motion was fluctuating to fast, making the crane unable to keep the crane tip at a fixed point.

The input shaping control algorithm shapes the control input given from the operator in a manner that makes the crane stop at any point with reduced payload swing. The effect of increasing the robustness of the shaper was checked using three different shapers. Simulations showed that the input shapers were able to reduce the residual vibration.

For a 5 meter long cable, the zero vibration shaper reduced the residual vibration from 1 meter to approximately 20 cm, whereas the zero vibration derivative shaper reduced the vibration to about 5 cm. The zero vibration derivative derivative shaper was able to reduce the vibration to about 2.5 cm.

Contents

List of Tables

List of Figures

Nomenclature

AC	Alternating Current
AHC	Active Heave Compensation
AOPS	Automatic Overload Protection System
AT	Auto Tension
CF	Center of Flotation
COG	Center of Gravity
DH	Denavit-Hartenberg
DOF	Degrees of Freedom
EI	Extra Insensitive
GUI	Graphical User Interface
HPU	Hydraulic Power Unit
MRU	Motion Reference Unit
PD	Proportional Derivative
PLC	Programmable Logic Controller
SI	Specified Intensity
UM-ZV	Unity Magnitude Zero Vibration
VAC	Voltage Alternating Current
ZV	Zero Vibration
ZVD	Zero Vibration Derivative
ZVDD	Zero Vibration Derivative Derivative

Chapter 1

Introduction

The aim for this master thesis is to model and develop control strategies for an offshore crane. New control algorithms that make crane operations safer and faster are needed in the future. This thesis is a part of that work and it is a continuation of the authors project assignment.

1.1 Background and Motivation

Cranes on offshore vessels are used in various operations all over the world. Dependent on the operations, the cranes differ in size, lifting capacity, number of joints, etc., and they are designed with regards to the environment and desired working area. Due to the custom-made design, the cranes can be placed on a wide variety of vessels, such as

- Diving support vessel vessels,
- Subsea construction and maintenance vessels,
- Platform supply vessel ,
- Anchor handling tug supply vessels,
- Drill ships,
- Floating production storage and offloading vessels,
- Research vessels,
- Seismic vessels.

Depending on the type of vessel a crane is placed on, operations span from simple bulk handling to advanced subsea lifting in harsh weather. The robust design strengthens the crane's reliability and enables it to perform crane operations for a longer period of time.

Furthermore, MacGregor's HMC3568 150t active boost crane has been used as a basis for developing the crane model. The crane is cable of lifting up to 150 tons, single line.

MacGregor (formerly named Hydramarine) has a long history developing equipment for the offshore industry. The company was founded in 1977 and began selling hydraulic pumps, small cranes and davits. During the 90's they sold a wide range of simple cranes and winches. In 2001 the first large MacGregor active heave compensating (AHC) crane was delivered, with the first 150 ton knuckle-jib crane being delivered the following year (MacGregor, 2010). During the last decade there has been an explosion in software complexity, leading to advanced features such as auto-tension, heave compensation (passive and active) and automatic overload protection. MacGregor, as well as other equipment handling suppliers, saw an increasingly interest in their technology during this decade.

Now, as the energy requirements grow exponentially, so too does the market for offshore cranes. The difficult nature of this market is that it demands even more complex systems and equipment capable of operating at deeper waters, more remote locations and in increasingly severe environments. Increasing the "weather operation window" is a common phrase in the industry, and of great interest to ship owners. To close the gap between today's solutions and what is needed in the future, new technology that is safer, more efficient and more robust has to be developed.

Input Shaping is a control shaping technique that has the potential to bring today's cranes towards the need of the future. The technique works by shaping the operator input with an input shaper so that payload swing is removed. Input shaping is a feedforward control strategy, i.e. it does not require feedback from sensors. Together with a feedback controller, one could obtain an effective and precise control system (Garrido et al., 2008). On board crane vessel, reduction of payload swing is very important due to the constantly changing workspace. A reduction of payload swing would increase safety and enable the operator to accurately position vulnerable payloads (Khalid et al., 2004).

The earliest usage of input shaping can be dated all the way back to O.J.M Smith in the 1950's (Mitchell, 1958). Smith developed the "posicast" control, which is nowadays reffered to as the Zero Vibration (ZV) shaper. The ZV got it's name from the fact that it removes all vibration if the system model is perfect. In modern time, the input shapers have successfully reduced vibration on cranes at nuclear facilities (Singhose et al., 2008), on gantry and overhead cranes (Sorensen et al., 2007) and during experiments on board of the space shuttle Endeavor (Tuttle and Seering, 1995). Input shapers have also proven to be effective on robots, chemical processes, to mention a few. Input shaping has demonstrated that it is an effective control method for reducing vibration, and cranes that utilize the method has shown an improvement in efficiency and safety (Sorensen et al., 2007).

Another method that has the ability to make crane operations safer, more efficient and more robust, is a workspace control. This control method would have to solve the inverse kinematics, in order for the operator to control the crane tip in the x-y-z direction. Today, each joint is controlled separately. Besides being useful for the operator, the vessel orientation could be incorporated in the kinematics, making the algorithm able to generate reference position that is fixed in an earth fixed frame.

1.2 Contribution

Chapter 4 presents a mathematical model of the HMC3568 crane. In addition, it is described how the cable model works, and how the crane model was modified to include vessel-crane interaction.

Chapter 6.5 presents two inverse kinematics algorithms that were implemented and tested. Besides being useful for the operator, these two algorithms are able to actively compensate for the vessel motion.

Chapter 6.6 presents a control shaping technique called input shaping. Performance and robustness of three different input shapers have been analysed and tested. To the authors knowledge, both input shaping and vessel-crane inverse kinematics have never been employed on a crane of this type.

1.3 Outline

Chapter 2 introduce and define technical terms that are essential for the understanding of this master thesis. Chapter 3 present and explain the design of the HMC3568 crane. In Chapter 4, a mathematical model is derived. In addition, the cable model and vessel-crane interaction are accounted for. The mathematical model, as well as the Simulink model, are verified in Chapter 5. Furthermore, Chapter 6 deals with the control of the modeled crane. The inverse kinematics and the input shapers are described in detail in this chapter. Finally, Chapter 7 conclude and give recommendations for further work.

Chapter 2

Preliminaries on Robot Modeling Theory

The purpose of this chapter is to introduce and define basic technical terms that are essential for the understanding of this thesis. Offshore cranes come in various sizes and capabilities. However, most of them share one similarity, they can be described by robot modeling theory. Translations, rotations, torque etc. are all parameters that have to be found. This chapter review some of the most important terms and equations regarding robot modeling theory. The purpose is to provide necessary information so that the following chapters are understandable. If the reader is up to date with robot modeling theory, it should be no problem to understand the following chapters without reading this chapter. Terminology, explanations, figures and equations are a summary from Mark W. Spong et al. (2006) and Sciavicco and Sciavicco (2000).

2.1 Notation

The notation used throughout this thesis is shown in Table 2.1, and on vector form in Table 2.2. Vectors and matrices are printed in bold face. A robot manipulator variable is represented with q_i, and the set of joint variables are represented as $\mathbf{q} = [q_1, q_2, \ldots, q_n]^T$. A coordinate system is written as $o_i x_i y_i z_i$, or in short form as $\{i\}$, where i is the frame number.

2.2 Robot Manipulator

A robot manipulator is a connection between joints and links. Together they form a robot manipulator which is capable of altering joint angles so the end point of the manipulator, often called the end effector, moves towards a desirable point in space.

DOF		forces and moments	linear and angular velocities	position and Euler angles
1	motion in the x-direction (surge)	X	u	x
2	motion in the y-direction (sway)	Y	v	y
3	motion in the z-direction (heave)	Z	w	z
4	rotation about the x-axis (roll)	K	p	ϕ
5	rotation about the y-axis (pitch)	M	q	θ
6	rotation about the z-axis (yaw)	N	r	ψ

Table 2.1: Notation used throughout this paper, inspired by Thor I. Fossen (2010).

NED position	$\mathbf{p}^n = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$	Orientation (Euler angles)	$\boldsymbol{\Theta}_b^o = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}$
Body-fixed linear velocity	$\mathbf{v}^n = \begin{bmatrix} u \\ v \\ w \end{bmatrix}$	Body-fixed angular velocity	$\boldsymbol{\omega} = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$
Body-fixed force	$\mathbf{f}_b^b = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$	Body-fixed moment	$\mathbf{m}_b^b = \begin{bmatrix} K \\ M \\ N \end{bmatrix}$

Table 2.2: Vector representation of table 2.1, inspired by Thor I. Fossen (2010).

In this thesis two kinds of joints have been used; prismatic and revolute. The prismatic is a linear joint. It can alter the length between two links, and is therefore also called a slider. Hydraulic cylinders are an example of prismatic joints. On the other hand, a revolute joint is a rotary joint that alters the relative angle between two links. Hinges on doors and a human elbows are examples of revolute joints.

Both the revolute and prismatic joint are one degree of freedom (DOF) joints. This means that the prismatic joint is only capable of altering the length along one axis. Similarly, the revolute joint is only capable of revolve around one axis. As a result of using one DOF joints, the manipulator's DOF equals the number of joints. A manipulator is said to be kinematically redundant if it got more than six DOF. If it got less than six DOF it is not possible to position the end effector with a arbitrary orientation.

2.3 Operational Space, Workspace and Configuration Space

The representation of the end effector position can be given as minimal number of coordinates with regards to the geometry of the structure. In the 3 dimensional case, a vector

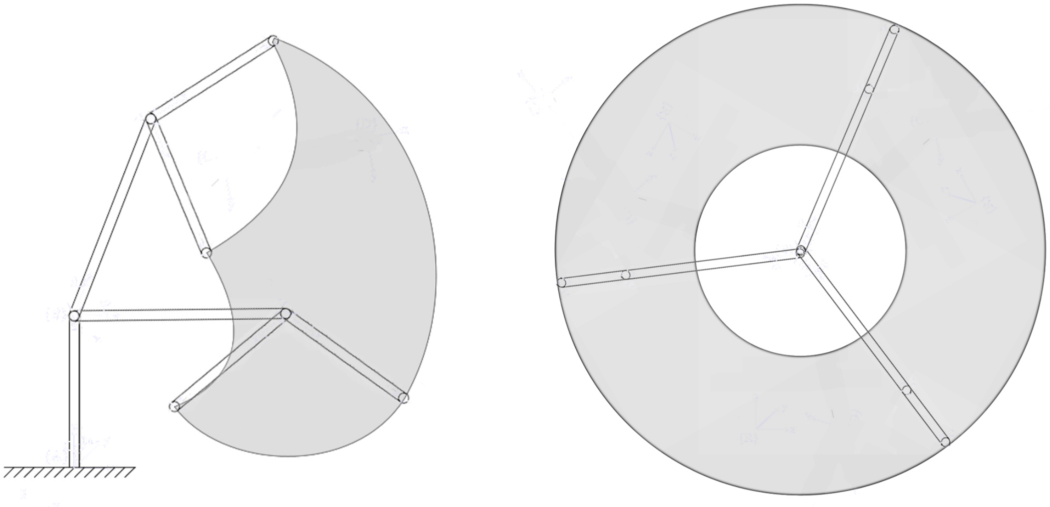


Figure 2.1: Workspace of an elbow manipulator.

with three elements are needed to represent the position. Likewise, a vector with three elements are used to represent the orientation of the end effector. These elements are called Euler angles. This way, the 3×1 position vector $\mathbf{p}$, and the 3×1 orientation vector $\boldsymbol{\Theta}$, are elements in the 6×1 vector

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \boldsymbol{\Theta} \end{bmatrix}. \tag{2.1}$$

This representation of the position and orientation is defined in the space in which the manipulator operations are specified in. Hence, this space is called the operation space.

A subcategory of the operational space, is the workspace. This region is described by the origin of the end effector frame when all the manipulator joints execute all possible motions, i.e.

$$\mathbf{p} = \mathbf{p}(\mathbf{q}), \qquad q_{im} \leq q_i \leq q_{iM}. \tag{2.2}$$

The region is limited by the manipulator geometry and the mechanical joint limits. In (2.2), the maximum and the minimum joint limit are denoted as q_{iM} and q_{im}, respectively. For an n-degree manipulator the reachable workspace is finite, closed and connected, as shown in Figure 2.1.

The configuration space, also called the joint space, denotes the space in which all configurations are defined in. By assuming that the manipulator consists of several individual rigid links, a configuration of a robot manipulator is the n × 1 joint variable vector q.

2.4 Rigid Body Motion

An important part of robot modeling concerns transformations and rotations of various coordinate frames. These operations make it possible to calculate the end effectors position based on the base coordinates of the manipulator.

2.4.1 Rotations

In order to represent the relative position and orientation of one rigid body with respect to another, a coordinate system $o_i x_i y_i z_i$, also called reference frame i, will be attached to each body part. It is possible for two reference frames with the same origin to represent the 3×1 vector $\mathbf{p}_a^0$ in $o_0 x_0 y_0 z_0$ as

$$\mathbf{p}_a^0 = \mathbf{R}_1^0 \mathbf{p}_a^1, \tag{2.3}$$

when $\mathbf{p}_a^1$ is given in $o_1 x_1 y_1 z_0$ as a 3×1 vector. The 3×3 rotation matrix $\mathbf{R_1^0}$ is given as

$$\mathbf{R}_1^0 = \mathbf{R}_{z,\Phi} \mathbf{R}_{y,\Theta} \mathbf{R}_{x,\Psi} =$$

$$= \begin{bmatrix} c_\psi c_\theta & -s_\psi c_\phi + c_\psi s_\theta s_\phi & s_\psi s_\phi + c_\psi c_\phi s_\theta \\ s_\psi c_\theta & c_\psi c_\phi + s_\phi s_\theta s_\psi & -c_\psi s_\phi + s_\theta s_\psi c_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix}. \tag{2.4}$$

Here, $\cos(\psi)$ and $\sin(\psi)$ are shortened to c_ψ and s_ψ for convenience. Equation (2.4) is the rotation matrix that rotates $\mathbf{p}_a$ from $o_1 x_1 y_1 z_1$ to $o_0 x_0 y_0 z_0$. The rotation matrix is expressed in Euler angles and is obtained by first performing a rotation of Φ degrees about the z axis, then θ degrees about y, and Ψ degrees about the x axis. These rotations alone are called basic rotations.

2.4.2 Homogeneous Transformation

A rigid body motion is a translation combined with a rotation. When a motion consist of both a rotation and a translation, it is called a transformation. Let $\mathbf{p}_1^0$ be the 3×1 vector that represent the origin of $o_1 x_1 y_1 z_1$ relative to $o_0 x_0 y_0 z_0$, expressed in $o_0 x_0 y_0 z_0$, and $\mathbf{R}_1^0$ the rotation matrix. $\mathbf{p}_a^0$ can then be expressed as

$$\mathbf{p}_a^0 = \mathbf{R}_1^0 \mathbf{p}_a^1 + \mathbf{p}_1^0. \tag{2.5}$$

Furthermore, $o_2x_2y_2z_2$ is introduced. Point a with respect to $o_1x_1y_1z_1$ can be written as

$$\mathbf{p}_a^1 = \mathbf{R}_2^1\mathbf{p}_a^2 + \mathbf{p}_2^1. \tag{2.6}$$

Now, combining (2.6) and (2.5) yields

$$\mathbf{p}_a^0 = \mathbf{R}_1^0\mathbf{R}_2^1\mathbf{p}_a^2 + \mathbf{R}_1^0\mathbf{p}_2^1 + \mathbf{p}_1^0, \tag{2.7}$$

$$\mathbf{p}_a^0 = \mathbf{R}_0^2\mathbf{p}_a^2 + \mathbf{p}_2^0, \tag{2.8}$$

where

$$\mathbf{R}_2^0 = \mathbf{R}_1^0\mathbf{R}_2^1, \tag{2.9}$$

$$\mathbf{p}_2^0 = \mathbf{p}_1^0 + \mathbf{R}_1^0\mathbf{p}_2^1. \tag{2.10}$$

One can easily see that equation 2.7 could lead to complicated equations if a series transformations were used. By introducing transformations we try to simplify the equations. The transformation is given as

$$\mathbf{r}_a^0 = \mathbf{T}_1^0\mathbf{r}_a^1, \tag{2.11}$$

where $\mathbf{r}_a^0$ and $\mathbf{r}_a^1$ are the 4×1 vector representing $[\mathbf{p}_a^0, 1]^T$ and $[\mathbf{p}_a^1, 1]^T$, respectively. $\mathbf{T}_1^0$ is the 4×4 transformation matrix that transforms a vector from $o_1x_1y_1z_1$ to $o_0x_0y_0z_0$. It is given as

$$\mathbf{T}_1^0 = \begin{bmatrix} \mathbf{R}_1^0 & \mathbf{p}_1^0 \\ \mathbf{0} & 1 \end{bmatrix}. \tag{2.12}$$

Equation 2.7 can now be rewritten to

$$\mathbf{r}_a^0 = \mathbf{T}_1^0\mathbf{T}_2^1\mathbf{r}_a^2 = \begin{bmatrix} \mathbf{R}_2^0 & \mathbf{R}_1^0\mathbf{p}_2^1 + \mathbf{p}_1^0 \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{p}_a^2 \\ 1 \end{bmatrix}. \tag{2.13}$$

This way of representing transformations makes it easy to write and understand the operations , especially when it's a large number of transformations.

2.5 Kinematics

The problem of determine the kinematics can be divided into two parts, namely forward kinematics and inverse kinematics. The main objective of the kinematics is to describe the motion of the robot manipulator without taking torque and forces into consideration. The forward kinematic problem is to determine the position and orientation of the end effector given the configuration of the robot manipulator. The inverse kinematic problem is to determine the configuration of the manipulator given the end effector position and orientation.

2.5.1 Kinematic Chains

As described in the previous sections, a robot manipulator consist of links connected to each other by joints. The joints can either be simple such as prismatic and revolute joints, or more complex such as ball or socket joints. A nice thing with simple joints is that they only got one DOF, which means that the revolute joint revolve around one axis, and that the prismatic alter the linear displacement along one axis.

A manipulator with n joints consist of n+1 links since each joint is connected with two links. The joints are numbered from 1 to n, starting with 1 at the first joint, usually the one connected to the ground/base link. Joint one is therefore connected to link zero and link one, with link zero considered as an unmovable link. When a force or torque actuate joint i, link i moves and the configuration is changed. All links in the configuration have one joint variable q_i. They represent

$$q_i = \begin{cases} \Theta_i & \text{represent an angle if joint i is revolute,} \\ d_i & \text{represent a displacement if joint i is prismatic.} \end{cases} \tag{2.14}$$

In addition, it is attach a coordinate frame rigidly to each link. In particular, link i has reference frame $o_i x_i y_i z_i$ attached. Figure 2.2 illustrates how this can be done for an elbow manipulator.

As figure 2.2 suggest, frame $o_0 x_0 y_0 z_o$ is placed at the bottom of the manipulator. This means that link zero is the ground. As a result, all positions and orientations are expressed with respect to this frame.

2.5.2 The Denavit-Hartenberg Convention

The DH convention provide a systematic procedure to develop robot manipulator kinematics. The kinematics for a n link robot manipulator can be extremely complex and the convention simplifies the analysis considerably. It also provides a universal language engineers can use to communicate with each other.

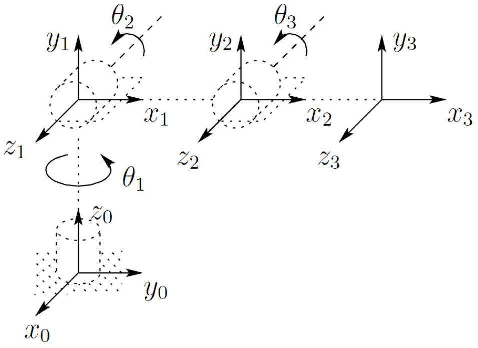


Figure 2.2: Coordinate frames attached to an elbow manipulator.

With this convention, each homogeneous transformation $\mathbf{A}_i$ is represented as a product of two rotations and two translations

$$\begin{aligned}\mathbf{A}_i &= Rot_{z,\Theta_i} Trans_{z,d_i} Trans_{x,a_i} Rot_{x,\alpha_i} \\ &= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} c_{\alpha_i} & s_{\theta_i} s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i} c_{\alpha_i} & -c_{\theta_i} s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}.\end{aligned} \tag{2.15}$$

The transformation matrices can now be found by performing multiple homogeneous transformations

$$\mathbf{T}_i^0 = \mathbf{A}_1(q_1) \cdots \mathbf{A}_i(q_i). \tag{2.16}$$

The four parameters θ_i, d_i, a_i and α_i are associated with joint i. The parameters represent

- d_i - Link offset: The distance from the origin of $o_{i-1}x_{i-1}y_{i-1}z_{i-1}$ to the intersection of z_{i-1} and x_i, measured along z_{i-1}. d_i is the joint variable if the joint is prismatic.
- θ_i - Joint angle: The angle of rotation from x_{i-1} to the x_i axis, measured in a plane normal to z_{i-1}. θ_i is the joint variable if the joint is revolute.
- a_i - Link length: The distance between the axes z_{i-1} and z_i, measured along x_i .
- α_i - Twist angle: The angle between the axes z_{i-1} and z_i, measured in a plane normal to x_i. The positive sense for α is determined from z_0 to z_1 by the right hand rule.

Three of these variables are constants since joint i only got one variable. Accordingly to (2.15) there is four parameters that have to be found, while a transformation without using the D-H convention need six parameters. This simplifies the effort of deriving the transformation matrices. However, one have to place the origin and coordinate axes according to some rules. The coordinate axes should be chosen as

1. The z-axis is in the direction of the joint axis, i.e. along the the revolute axis for revolute or the displacement axis for prismatic.
2. The x-axis is parallel to the common normal, i.e. $x_i = z_{i-1} \times z_i$. If there is no unique common normal (parallel z axes), then x_i is a free parameter. The direction of x_i is from z_{i-1} to z_i.
3. The y-axis follows from the x- and z-axis by choosing it to be a right-handed coordinate system.

In fact, the origin and coordinate axes of figure 2.2 is placed according to the DH conversion. Table 2.3 shows the DH parameters for the elbow manipulator in figure 2.2.

Link	a_i	α_i	d_i	θ_i
1	0	90°	d_1	θ_1*
2	l_1	0°	0	θ_2*
3	l_2	0°	0	θ_3*

Table 2.3: DH parameters for the elbow manipulator.

2.5.3 Velocity - The Jacobian

A moving reference frame got both linear and angular velocity components. The manipulator Jacobian relates the link angular and linear velocity with the joint velocities. It can be split into two parts. One that relates the joint velocity of link i with the linear velocity $\dot{\mathbf{p}}_i$, and the other relates the joints angles to the angular velocity $\boldsymbol{\omega}_i$, both expressed in $o_i x_i y_i z_i$. The velocities for link i can be written as

$$\dot{\mathbf{p}}_i = \boldsymbol{j}^{(i)}_{\dot{p}_1}\dot{q}_1 + \ldots + \boldsymbol{j}^{(i)}_{\dot{p}_i}\dot{q}_i = \mathbf{J}^{(i)}_{\dot{p}}\dot{\mathbf{q}}, \tag{2.17}$$

$$\boldsymbol{\omega}_i = \boldsymbol{j}^{(i)}_{\omega_1}\dot{q}_1 + \ldots + \boldsymbol{j}^{(i)}_{v_i}\dot{q}_i = \mathbf{J}^{(i)}_{\omega}\dot{\mathbf{q}}. \tag{2.18}$$

Here, the subscript of the Jacobians distinguish the linear ($\mathbf{J}^{(i)}_{\dot{p}}$) and the angular ($\mathbf{J}^{(i)}_{\omega}$) Jacobian. The superscript gives the number of links that have been taken into account. The $6 \times n$ Jacobian to consider are then

$$\mathbf{J}^{(i)}_{\dot{p}} = \begin{bmatrix} \boldsymbol{j}^{(i)}_{\dot{p}_1} & \ldots & \boldsymbol{j}^{(i)}_{\dot{p}_i} & 0 & \ldots & 0 \end{bmatrix}, \tag{2.19}$$

$$\mathbf{J}^{(i)}_{\omega} = \begin{bmatrix} \boldsymbol{j}^{(i)}_{\omega_1} & \ldots & \boldsymbol{j}^{(i)}_{\omega_i} & 0 & \ldots & 0 \end{bmatrix}. \tag{2.20}$$

Taking the type of joint into consideration, each element will look like

$$\begin{bmatrix} \boldsymbol{j}_{\dot{p}_j}^{(i)} \\ \boldsymbol{j}_{\omega_j}^{(i)} \end{bmatrix} \begin{cases} \begin{bmatrix} \mathbf{z}_{j-1} \times (\mathbf{p}_i - \mathbf{p}_{j-1}) \\ \mathbf{z}_{j-1} \end{bmatrix} & \text{if joint i is revolute,} \\ \begin{bmatrix} \mathbf{z}_{j-1} \\ \mathbf{0} \end{bmatrix} & \text{if joint i is prismatic.} \end{cases} \tag{2.21}$$

Here, the 3×1 vector $\mathbf{z}_{j-1}$ is given by the the first three elements in the third column of $\mathbf{T}_{j-1}^{0}$, while $\mathbf{p}_{j-1}$ is given by the first three elements of the fourth column of $\mathbf{T}_{j-1}^{0}$.

2.6 Dynamics

While the kinematics describes the motion of a robot manipulator without considering the torque applied, the dynamics describes the relationship between torque and motion.

2.6.1 Lagrange Formulation

The Lagrange formulation is a conceptual simple and systematic procedure to derive the dynamics of a manipulator. The Lagrangian dynamic formulation is an energy based approach of the system, i.e. the approach derives the dynamics using the kinetic and potential energy of the system.

With the Lagrange formulation, the equations of motion can be derived systematically without regards of the reference coordinate frame. A set of variables λ_i, $i = 1, \ldots, n$, named generalized coordinates , are chosen which effectively describe the link positions of a manipulator. The Lagrangian of the mechanical system can be defined as the difference between the kinetic and potential energy

$$\mathcal{L} = \mathcal{K} - \mathcal{P}, \tag{2.22}$$

where $\mathcal{K}$ and $\mathcal{P}$ represents the total kinetic energy and potential energy, respectively. For a manipulator the most natural choice for the generalized coordinates are the joint variables

$$\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{bmatrix} = \mathbf{q}. \tag{2.23}$$

With this choice of generalized coordinates, the Euler-Lagrange equations of motion are

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = \tau_i, \tag{2.24}$$

where τ_i is the force associated with link i. Equation (2.24) establish the relations between the force applied to each joint and the joint positions, velocities and accelerations. Hence, the equations allow us deriving the dynamical model using the potential and kinetic energy of the system.

2.6.2 Computation of Kinetic Energy

The total kinetic energy of a n-link manipulator is given by the sum of the contributions relative to the motion of each link

$$\mathcal{K} = \sum_{i=1}^{n} \mathcal{K}_i. \tag{2.25}$$

where $\mathcal{K}_i$ is the kinetic energy of link i. The kinetic energy contribution of link i is given by

$$\mathcal{K}_i = \frac{1}{2} \int_{V_i} \dot{\mathbf{p}}_i^{*^T} \dot{\mathbf{p}}_i^{*} \rho dV, \tag{2.26}$$

where $\dot{\mathbf{p}}_i^*$ denotes the linear velocity vector, expressed in the base frame, and ρ is the density of the elementary particle of volume dV. The integration is performed over the total volume of link i, V_i. Now, consider the position vector $\boldsymbol{p}_i^*$ of the elementary particle and the position vector $\boldsymbol{p}_i$ of the link's center of mass, both expressed in the base frame. One has

$$\mathbf{r}_i = \begin{bmatrix} r_{ix} & r_{iy} & r_{iz} \end{bmatrix} = \boldsymbol{p}_i^* - \boldsymbol{p}_i, \tag{2.27}$$

with

$$\mathbf{p}_i = \frac{1}{m_i} \int_{V_i} \mathbf{p}_i^* \rho dV, \tag{2.28}$$

where m_i is the mass of link i. As a consequence, the linear velocity of the elementary particle can be expressed as

$$\begin{aligned} \dot{\mathbf{p}}_i^* &= \dot{\mathbf{p}}_i + \boldsymbol{\omega}_i \times \mathbf{r}_i \\ &= \dot{\mathbf{p}}_i + \mathbf{S}(\boldsymbol{\omega}_i)\mathbf{r}_i. \end{aligned} \tag{2.29}$$

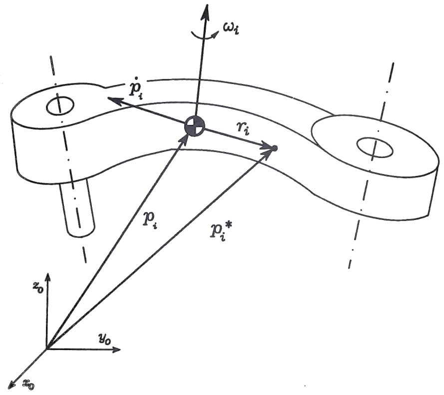


Figure 2.3: Kinematic description of link i.

The position and velocity vectors are shown in Figure 2.3. The translation and rotational part of the kinetic energy are found by substituting (2.29) into (2.26). The translation contribution of the kinetic energy is

$$\frac{1}{2}\int_{V_i} \dot{\mathbf{p}}_i^T \dot{\mathbf{p}}_i \rho dV = \frac{1}{2} m_i \dot{\mathbf{p}}_i^T \dot{\mathbf{p}}_i, \tag{2.30}$$

and the rotational part of the kinetic energy is

$$\frac{1}{2}\int_{V_i} \mathbf{r}_i^T \mathbf{S}(\boldsymbol{\omega}_i)^T \mathbf{S}(\boldsymbol{\omega}_i)\mathbf{r}_i \rho dV = \frac{1}{2}\boldsymbol{\omega}_i^T \left(\int_{V_i} \mathbf{S}(\mathbf{r}_i)^T \mathbf{S}(\mathbf{r}_i)\rho dV\right)\boldsymbol{\omega}_i. \tag{2.31}$$

Here, m_i is the mass of link i and $\mathbf{S}(\cdot)$ is the cross product operator

$$\mathbf{S}(\mathbf{r}_i) = \begin{bmatrix} 0 & -r_{iz} & r_{iy} \\ r_{iz} & 0 & -r_{ix} \\ -r_{iy} & r_{ix} & 0 \end{bmatrix}. \tag{2.32}$$

In (2.31) the property $\mathbf{S}(\boldsymbol{\omega}_i)\mathbf{r}_i = -\mathbf{S}(\mathbf{r}_i)\boldsymbol{\omega}_i$ has been used. Inserting (2.32) into (2.31) yields

$$\frac{1}{2}\boldsymbol{\omega}_i^T \left(\int_{V_{l_i}} \mathbf{S}(\mathbf{r}_i)^T \mathbf{S}(\mathbf{r}_i)\rho dV\right)\boldsymbol{\omega}_i = \frac{1}{2}\boldsymbol{\omega}_i^T \mathbf{I}_i^{(0)} \boldsymbol{\omega}_i. \tag{2.33}$$

$\mathbf{I}_i^{(0)}$ is symmetric and represents the inertia tensor relative to the center of mass of link i, expressed in the base frame. It is worth noticing that the inertia tensor, when expressed in the base frame, is configuration-dependent. If the angular velocity of link i is expressed with reference to a frame attached to the link, as in the DH convention, it is expressed as

$$\boldsymbol{\omega}_i^{(i)} = \mathbf{R}_i^{0^T}\boldsymbol{\omega}_i. \tag{2.34}$$

It can be shown that the following relation then holds

$$\mathbf{I}_i = \mathbf{R}_i^0\mathbf{I}_i^{(i)}\mathbf{R}_i^{0^T}, \tag{2.35}$$

where $\mathbf{I}_i^{(i)}$ is expressed with regards to the link frame. The new inertia tensor matrix $\mathbf{I}_i^{(i)}$ is therefore independent of the manipulator's configuration. It can be written as

$$\mathbf{I}_i^{(i)} = \begin{bmatrix} \int(y^2+z^2)\rho dV & \int xy\rho dV & \int xz\rho dV \\ \int xy\rho dV & \int(x^2+z^2)\rho dV & \int yz\rho dV \\ \int xz\rho dV & \int yz\rho dV & \int(x^2+y^2)\rho dv \end{bmatrix}. \tag{2.36}$$

By summing the translational and rotational contributions, the total kinetic energy of link i can be written as

$$\mathcal{K}_i = \frac{1}{2}m_i\dot{\mathbf{p}}_i^T\dot{\mathbf{p}}_i + \frac{1}{2}\boldsymbol{\omega}_i^T\mathbf{R}_i^0\mathbf{I}_i^{(i)}\mathbf{R}_i^{0^T}\boldsymbol{\omega}_i. \tag{2.37}$$

Using Equation (2.17) and Equation (2.18), the total kinetic energy can be written as

$$\begin{aligned} \mathcal{K} &= \sum_{i=1}^{n}\frac{1}{2}m_i\dot{\mathbf{q}}^T\mathbf{J}_{\dot{p}}^{(i)^T}\mathbf{J}_{\dot{p}}^{(i)}\dot{\mathbf{q}} + \frac{1}{2}\dot{\mathbf{q}}^T\mathbf{J}_{\omega}^{(i)^T}\mathbf{R}_i^0\mathbf{I}_i^{(i)}\mathbf{R}_i^{0^T}\mathbf{J}_{\omega}^{(i)}\dot{\mathbf{q}} \\ &= \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}d_{ij}(\mathbf{q})\dot{q}_i\dot{q}_j \\ &= \frac{1}{2}\dot{\mathbf{q}}^T\mathbf{D}(\mathbf{q})\dot{\mathbf{q}}, \end{aligned} \tag{2.38}$$

where

$$\mathbf{D}(\mathbf{q}) = \left[\sum_{i=1}^{n}\{m_i\mathbf{J}_{\dot{p}_i}^T\mathbf{J}_{\dot{p}_i} + \mathbf{J}_{\omega_i}^T\mathbf{R}_i^0\mathbf{I}_i^{(i)}\mathbf{R}_i^{0^T}\mathbf{J}_{\omega_i}\}\right]. \tag{2.39}$$

Here, $\mathbf{D}(\mathbf{q})$ is the $n \times n$ matrix which is symmetric and positive definite.

2.6.3 Computation of Potential Energy

As done for the kinetic energy, the potential energy stored in the manipulator is the sum of the contribution relative to each link. On the assumption that the manipulator only consists of rigid links, the only source of potential energy is the gravity. The potential is

$$\mathcal{P} = \sum_{i=1}^{n} \mathcal{P}_i, \tag{2.40}$$

where

$$\mathcal{P}_i = -\int_{V_i} \mathbf{g}_0^T \mathbf{p}_i^* \rho dV = -m_i \mathbf{g}_0^T \mathbf{p}_i. \tag{2.41}$$

Here $\mathbf{g}_0$ is the 3×1 gravity acceleration vector in the base frame, and $\mathbf{p}_i$ is the distance to from $o_0 x_0 y_0 z_0$ to CG of link i. If the z is the vertical axis , $\mathbf{g}_0 = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}^T$.

2.6.4 Equations of Motion

Having computed the kinetic and potential energy, the Lagrangian (2.22) of the system can be written as

$$\begin{aligned} \mathcal{L}(\dot{\mathbf{q}}, \mathbf{q}) &= K(\dot{\mathbf{q}}, \mathbf{q}) - P(\mathbf{q}) \\ &= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij}(\mathbf{q}) \dot{q}_i \dot{q}_j + \sum_{i=1}^{n} m_i \mathbf{g}_0^T \mathbf{p}_i. \end{aligned} \tag{2.42}$$

First, taking the derivatives of the kinetic part of the Lagrangian, as required by (2.24), yields

$$\begin{aligned} \frac{d}{dt} \frac{\partial \mathcal{K}}{\partial \dot{q}_i} &= \sum_{j=1}^{n} d_{ij}(\mathbf{q}) \ddot{q}_j + \sum_{j=1}^{n} \frac{d}{dt} d_{ij}(\mathbf{q}) \dot{q}_j \\ &= \sum_{j=1}^{n} d_{ij}(\mathbf{q}) \ddot{q}_j + \sum_{j=1}^{n} \sum_{k=1}^{n} \frac{d}{dq_k} d_{ij}(\mathbf{q}) \dot{q}_k \dot{q}_j, \end{aligned} \tag{2.43}$$

and

$$\frac{\partial \mathcal{K}}{\partial q_i} = \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n}\frac{d}{dq_i}d_{ij}(\mathbf{q})\dot{q}_k\dot{q}_j. \tag{2.44}$$

Here the indices of summation have been switched. Second, the potential contribution to (2.24) is

$$\begin{aligned}\frac{\partial \mathcal{P}}{\partial q_i} &= -\sum_{j=1}^{n} m_j \mathbf{g}_0^T \frac{d}{dq_i}\mathbf{p}_j \\ &= -\sum_{j=1}^{n} m_j \mathbf{g}_0^T \boldsymbol{j}_{\dot{p}_i}^{(j)} = g_i(\mathbf{q}),\end{aligned} \tag{2.45}$$

where the index of summation also has been changed. Due to the fact that the potential energy is independent of $\dot{\mathbf{q}}$, the following relation holds

$$\frac{d}{dt}\frac{\partial \mathcal{P}}{\partial \dot{q}_i} = 0. \tag{2.46}$$

As a result, the equations of motion are

$$\sum_{j=1}^{n} d_{ij}(\mathbf{q})\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n}\left(\frac{d}{dq_k}d_{ij}(\mathbf{q})\dot{q}_k\dot{q}_j - \frac{1}{2}\frac{d}{dq_i}d_{jk}(\mathbf{q})\dot{q}_k\dot{q}_j\right) + g_i(\mathbf{q}) = \tau_i, \quad i = 1,\ldots,n. \tag{2.47}$$

On matrix form it can be written

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}. \tag{2.48}$$

Here $\mathbf{D}(\mathbf{q})$ is defined in (2.39) and $\mathbf{g}(\mathbf{q})$ is given in (2.45). Element (k,j) of the $n \times n$ matrix $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ is given as

$$c_{kj} = \sum_{i=1}^{n}\frac{1}{2}\left[\frac{\partial d_{kj}}{\partial q_j} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k}\right]\dot{q}_i. \tag{2.49}$$

A notable property of the dynamical model, is the skew symmetry of

$$\mathbf{N}(\mathbf{q},\dot{\mathbf{q}}) = \dot{\mathbf{D}}(\mathbf{q}) - 2\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}). \tag{2.50}$$

For more thorough explanations and derivations, see Mark W. Spong et al. (2006) and Sciavicco and Sciavicco (2000).

Chapter 3

Crane Design

This chapter introduces the crane that is used as a basis for the mathematical modeling and the implementation in Simulink. In the first section, a description of the crane and it's main components will be given. Then, a more detailed overview of the overall control system, with focus on actuators and sensors, will be given. In the last section, necessary mechanical restrictions are listed. Throughout the thesis there will be used a Macgregor AHC crane as a basis for the modeling and simulation work. The design is robust and several other crane suppliers develop similar cranes. Most of the information presented here was provided by MacGregor (2010).

3.1 Description

When a crane operation is performed in a moving environment, like on a ship, it is crucial to have a robust and safe system controlling the load at all time. The Macgregor AHC knuckle jib crane has proven to be such a crane. It is 10 yeas since the first model was delivered, and since then, the crane has been continuously improved.

Figure 3.1 shows a 150 metric tonne AHC knuckle-jib crane in operation. It was installed on the supply vessel Seven Sisters in 2008, with a design that focused on hoisting speed/-capacities, as desired by client needs. This is the crane that has been used a basis for the mathematical and Simulink modeling. The physical dimensions are shown in Table 3.1. The dimensions are given as length in x, y and z direction for the knuckle jib and main jib. For the king it is given as a radius and length in z direction. Weight of each part is given in kilos, and the center of gravity (CG) defines where the gravity force acts upon the body part. The CG length is expressed as the distance from joint i to the CG of link i.It will later be described why the center of gravity is placed as it is. For convenience, a schematic drawing of the crane can be seen in Figure 3.2.

Cranes of this type are mounted on a foundation on the star or port side of the vessel. It is preferred that the crane is mounted as close as possible to the center of flotation (CF),

Figure 3.1: Seven Sisters with the AHC knuckle jib crane in operation. Courtesy of Subsea7 (2011).

Name of part	Dimension [m]	Weight [kg]	CG [m]
King	(3.568, 4.250)	200 000	2.125
Main jib	(1.5, 2.5, 21)	30 000	10.5
Knuckle jib	(1.5, 2.5, 12)	23 000	6

Table 3.1: Offshore crane dimensions.

in order to minimize roll and pitch motion. Normally, the crane is mounted close to the transverse CF axis, which reduces the crane motion due to the vessel pitch motion. In contrast, the crane is normally placed on the star or port side of the vessel. As a result, the vessel's roll motion induces crane tip motion. The placement ensures a large operating radius. However, the roll induced motion could have been reduced by placing the crane close to the CF longitudinal axis.

The pedestal is placed on top of the foundation. One of the most important unit for the heave compensation, the motion reference unit (MRU), is placed in the pedestal. The rotating king is installed on top of a rotating slew bearing. The king acts as the main structural element between the pedestal and the jib system. Most of the crane equipment are mounted on the king, which makes this part heavier then the others. Actuators, winch, wire drum, crane cabin etc. are equipment that are mounted on the king. The wire capacity on the HMC3568 is 3000 m and the weight is approximately 20 kg/m. The king structure also houses the hydraulic oil reservoir and the general machine room where most of the control valves are fitted. The total oil volume in the reservoir is approximately 6000 liters, plus oil in system loops. The crane cabin is where the crane operator controls the crane using touch screens, joysticks and other control hardware. Hydraulic actuators provide torque so the king are able to rotate. This motion is called slewing. Two large cylinders connect the king with the main jib. These provide torque such that the Main jib alter its

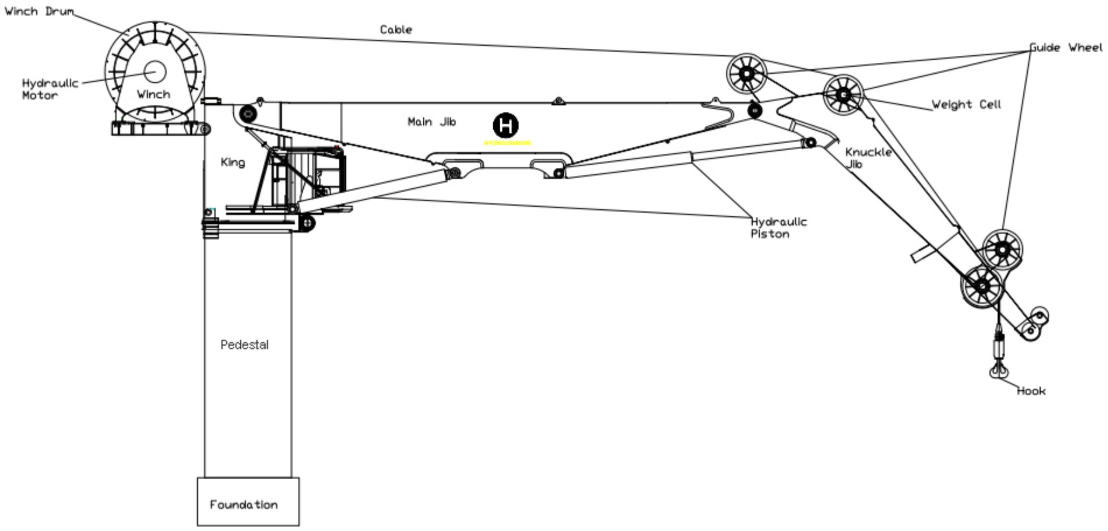


Figure 3.2: Drawing of the knuckle jib crane (Skagestad et al., 2008).

angle relative to the king.

Another pair of cylinders connect the main jib to the knuckle jib. At the end of the knuckle jib, the main and whip winch hangs freely. They can be lowered or elevated to a desired position by the crane operator. A hook is mounted at the end of each wire. This makes the crane able to hook up various objects.

In addition to the main and whip winch, there are a two tugger winches. The two tugger winches are mounted on the lower part of the king, and are mainly used to prevent payload oscillation. The whip winch is used in the same manner as the main winch. However, it has lower capacity than the main winch.

3.2 Actuators and Sensors

The active heave compensating cranes are highly sophisticated. It consist of several sensors, instruments and actuators connected to a control system. For a given input from the sensors, it produces an output to the actuators.

There are four electrical motors, each driving a hydraulic pump on the Hydraulic power unit (HPU). These motors are 300-550KW alternating current (AC) motors, driven by a 690 Voltage alternating current (VAC) three phase supply. The operator can start and stop the motors from the operator panels, but it also has a built in protection in form of circuit breakers. In addition to the four motors, an emergency motor is installed, which make the system redundant. To prevent system downtime, a filtration/cooling assembly is installed. It ensures good oil quality, and equally important, that the oil temperature is as desired.

Four hydraulic pumps feeds hydraulic power into a common control valve system that distribute the power among the actuators, based on signals from the crane control system. Furthermore, the crane control system is connected to joysticks at the operators chair inside the crane cabin. The power unit system is built in such a way that as little as one pump as able to control the entire crane, but with reduced velocity.

The control system get its input from sensors and instruments. They provide information about the current state of the system. There are encoders for measuring hook position, angle encoders for measuring main jib and knuckle jib angle, load sensors for measuring the hook load, pressure transmitters, temperature sensors, level sensors and a MRU. In the crane cabin, the operator got LCD screens, which is used for control and monitoring of the crane's current state.

There are different signal types used for communication between sensor/instruments and the Programmable Logic Controller (PLC) system. These signals are usually current or voltage based, depending on the type of sensor/instrument that is being dealt with, and whether it is an input or output.

In addition to the normal mode, the crane got some special functions. Three of these are:

- Automatic overload protection system (AOPS) - A fail to safe system where the winch will pay out wire if the load reaches a certain high value, and keep paying out only until the tension is below a certain lower value.
- Active heave compensation (AHC) - A control mode where the hook position is kept at a constant desirable hight above the seabed.
- Auto tension (AT) - This feature is a sub-mode of the AHC mode. Instead of keeping a certain height, it keeps a certain wire tension.

A schematic diagram of the crane system is shown in Figure 3.3.

3.3 Crane Limitations

Besides being limited by geometry and mechanical joint limits, there are some other restrictions that have to be accounted for. Table 3.2 shows the maximum and minimum joint angle, velocity and acceleration for the crane. These restrictions are a result of limited hydraulic flow and mechanical limits. One should notice that the restrictions of joint i is placed with regards to reference frame i. The placement of each reference frame will be carried out in the next chapter.

In reality, the maximum velocity and acceleration depend on the crane's configuration. For instance, the crane has higher slew rate if the operating radius is less than 15 meters. This makes Table 3.2 configuration dependent. However, this has not been taken into account in this thesis.

Joint	Orientation [deg]	Angular velocity [deg/s]	Angular acceleration [deg/s^2]
1	[0, 360]	6	2
2	[0, 86]	1	0.5
3	[-155, -37]	2	1

Table 3.2: Crane restrictions.

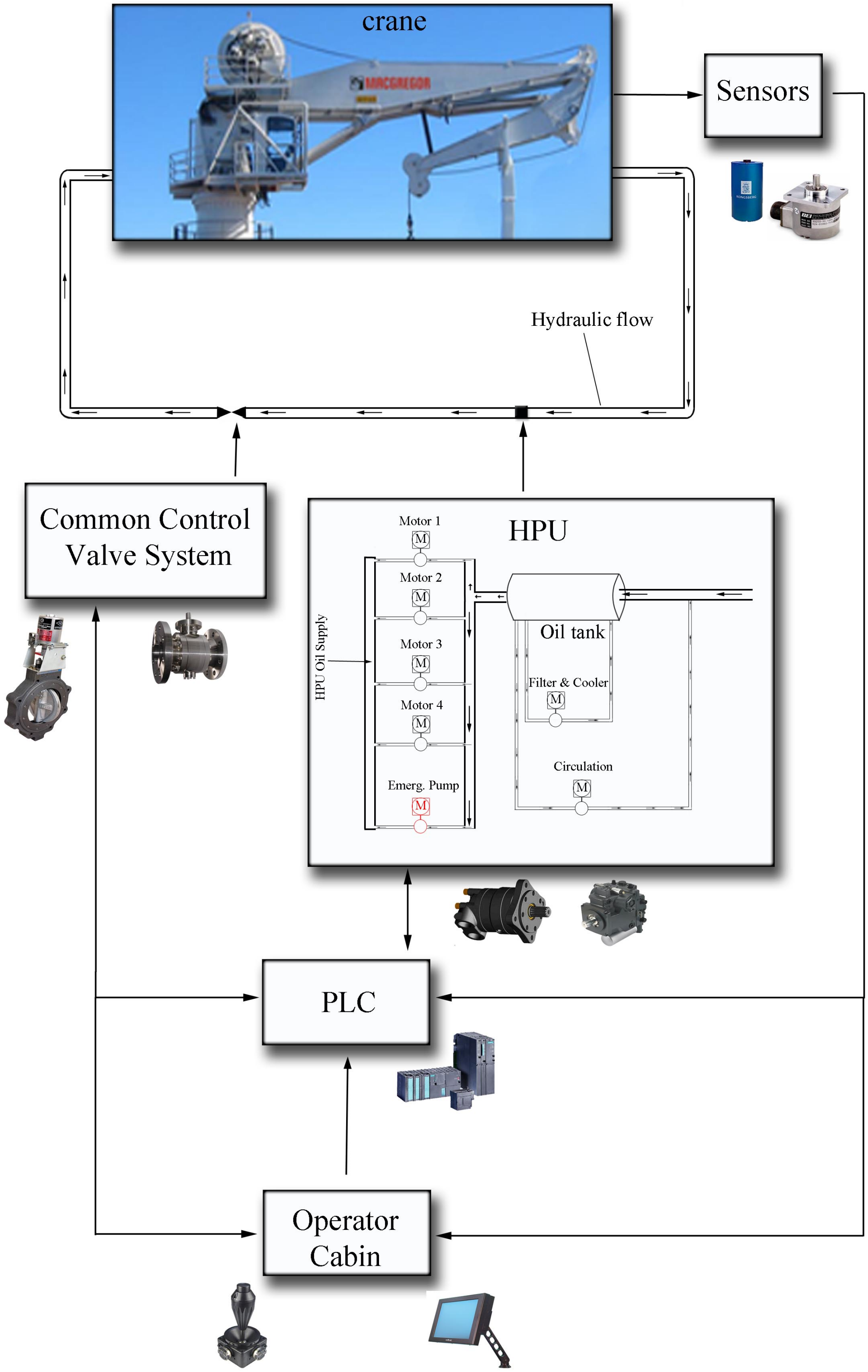


Figure 3.3: System drawing of the crane system.

Chapter 4

Modeling of Crane

In this chapter a crane model will be derived mathematically and implemented Simulink. The first section states the simplifications that was needed in order to use robot modeling theory. In addition, a figure shows how the reference frames were placed. In Section 4.2, a mathematical crane model has been derived using robot modeling theory. The last section concerns the implementation in Simulink. Here it will be described how the vessel-crane dynamics were found, and how the cable model works.

4.1 Simplifications

In order to derive a mathematical model, one first need to do some simplifications. The first simplification that is made is that each body part is symmetric and got an uniformly distributed mass. As a result, the link's CG is located at the center of each structural element. Consequently, the torque that is needed to rotate a joint will be slightly wrong compared to the actual crane. However, this thesis has not focused on producing the right amount of torque in every situation. If this had been the goal, one would have to incorporate an accurate actuator model. Secondly, all body parts are rigid bodies, i.e. the distance between the start and the end of each body part remains the same regardless of the external forces exerted on it. This means that deformation and flexibility is neglected. As a result, the crane tip position might be somewhat different from what it would have been in a real life. However, this difference would be small compared to the crane dimensions.

A drawing of the simplified knuckle jib crane is shown in Figure 4.1. It has labeled the crane coordinate frames {1},{2},{3} and {4}, which is placed according to the DH convention. In addition, the vessel fixed frame {B} and the earth fixed frame {N} has been labeled. They coincide if the vessel's orientation and position is zero. Frame $\{0\}$ is placed at $[-20, 8, 10]$ with regards to $\{B\}$. That means that the crane is placed on the aft port side, and that the foundation and the pedestal are modeled to be about 10 meters high. One should notice that the $\{B\}$ frame has the z-axis pointing upwards. The joint variables are also labeled in the figure.

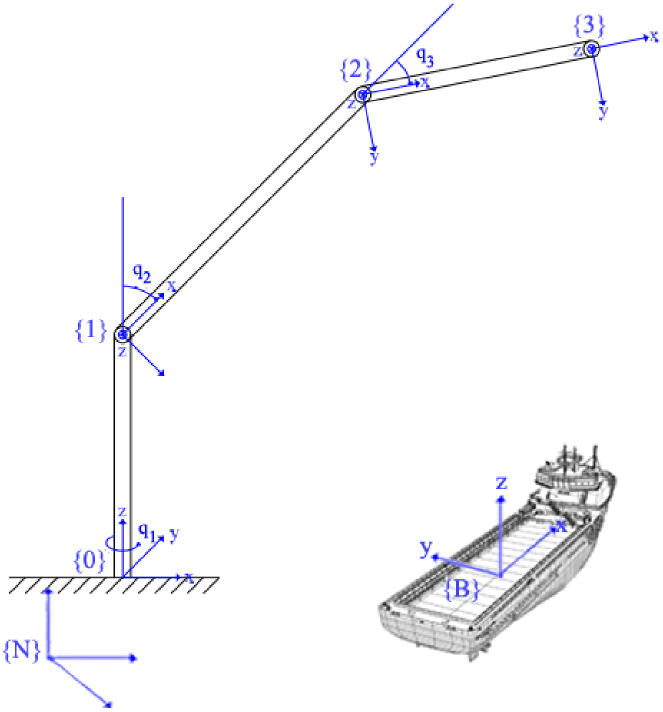


Figure 4.1: Simplified crane with local coordinate systems placed according to DH convention. Courtesy of Skagestad et al. (2008).

4.2 Mathematical Modeling

This section is split in six. In the first section, the inertia tensor matrix of each structural element are found. The next section show how the transformation and Jacobian matrices are derived. In the third section, the crane's inertia matrix and Coriolis and centripetal matrix are calculated using the transformation and Jacobian matrices. The inertia matrix and the Coriolis and centripetal matrix represent the kinetic part of the crane dynamics. The fourth section calculates the potential energy of the system, while the fifth section concerns the skew-symmetry property. Last, it is described how to find the vessel-crane dynamics.

4.2.1 Inertia Tensor Matrices

In this section, the inertia tensor matrix for the king, main jib and knuckle jib are defined. The inertia tensor matrix of link i will from now on be expressed in the local frame. That means that the inertia tensor matrix of link i will be expressed in $\{i\}$. From now on it will be denoted as $\mathbf{I}_i$. Due to the fact that all structural elements are symmetric, the off diagonal elements equals to zero. As a result, the representation of the inertia tensor of link i is

$$\mathbf{I}_i = \begin{bmatrix} I_{ix} & 0 & 0 \\ 0 & I_{iy} & 0 \\ 0 & 0 & I_{iz} \end{bmatrix}. \tag{4.1}$$

The inertia tensor matrix of each structural element are found using (2.36). The following subsections present these matrices.

Inertia Tensor King

$$I_{1x} = \frac{m_1(3r_1^2 + l_1^2}{12}, \tag{4.2}$$

$$I_{1y} = m_1 r_1^2, \tag{4.3}$$

$$I_{1z} = \frac{m_1(3r_1^2 + l_1^2}{12}, \tag{4.4}$$

where r_1 represents the radius and l_1 the length of the king.

Inertia Tensor Main Jib

$$I_{2x} = \frac{m_2(d_2^2 + h_2^2)}{12}, \tag{4.5}$$

$$I_{2y} = \frac{m_2(d_2^2 + l_2^2)}{12}, \tag{4.6}$$

$$I_{2z} = \frac{m_2(h_2^2 + l_2^2)}{12}, \tag{4.7}$$

where d_2 represents the depth, l_2 represents the length and h_2 the height of the main jib.

Inertia Tensor Knuckle Jib

$$I_{3x} = \frac{m_3(d_3^2 + h_3^2)}{12}, \tag{4.8}$$

$$I_{3y} = \frac{m_3(d_3^2 + l_3^2)}{12}, \tag{4.9}$$

$$I_{3z} = \frac{m_3(h_3^2 + l_3^2)}{12}, \tag{4.10}$$

where d_3 represents the depth, l_3 represents the length and h_3 the height of the knuckle jib.

4.2.2 Transformation Matrices and the Jacobian

To start with, the reference frames are placed according to the rules stated in Section 2.5.2. The resulting DH parameters are shown in Table 4.1, and the reference frames were shown in Figure 4.1. In fact, the crane got the same DH parameters, workspace and reference frames as the elbow manipulator, which were shown in Table 2.3 and Figures 2.1 and 2.2.

Link	a_i	α_i	d_i	θ_i
1	0	90°	l_1	θ_1
2	l_2	0°	0	θ_2
3	l_3	0°	0	θ_3

Table 4.1: DH parameters for the crane.

In Table 4.1, l_i represent the length of link i and θ_i represent the joint variable q_i. The DH table makes it very simple to define the homogeneous transformation between each link. Equation (2.15) yields

$$\mathbf{A}_1 = \begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.11}$$

$$\mathbf{A}_2 = \begin{bmatrix} c_2 & -s_2 & 0 & l_2c_2 \\ s_2 & c_2 & 0 & l_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.12}$$

$$\mathbf{A}_3 = \begin{bmatrix} c_3 & -s_3 & 0 & l_3c_3 \\ s_3 & c_3 & 0 & l_3s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.13}$$

Here, the c_i and s_i means $\cos(\theta_i)$ and $\sin(\theta_i)$, respectively. Furthermore, the transformation matrices are found using (2.16). They are found to be

$$\mathbf{T}_1^0 = \mathbf{A}_1, \tag{4.14}$$

$$\mathbf{T}_2^0 = \mathbf{A}_1\mathbf{A}_2 = \begin{bmatrix} c_1c_2 & -c_1s_2 & -s_1 & l_2c_1c_2 \\ c_2s_1 & -s_1s_2 & c_1 & l_2c_2s_1 \\ -s_2 & -c_2 & 0 & l_2s_2 + l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.15}$$

$$\begin{aligned} \mathbf{T}_3^0 &= \mathbf{A}_1\mathbf{A}_2\mathbf{A}_3 \\ &= \begin{bmatrix} c_1c_2c_3 - c_1s_2s_3 & c_1c_2s_3 - c_1c_3s_2 & -s_1 & l_3c_1c_{23} + l_2c_1c_2 \\ s_1c_2c_3 - s_1s_2s_3 & s_1s_3c_2 - s_1s_2c_3 & c_1 & l_3s_1c_{23} + l_2s_1c_2 \\ -s_2c_3 - c_2s_3 & s_2s_3 - c_2c_3 & 0 & l_3s_{23} + l_2s_2 + l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \end{aligned} \tag{4.16}$$

Here, c_{23} and s_{23} means $\cos(\theta_2 + \theta_3)$ and $\sin(\theta_2 + \theta_3)$, respectively. The last column in (4.16) has been simplified by using the following relationships

$$\sin(\alpha + \beta) = \sin(\alpha)\cos(\beta) + \cos(\alpha)\sin(\beta), \tag{4.17}$$

$$\cos(\alpha + \beta) = \cos(\alpha)cos(\beta) - sin(\alpha)\sin(\beta), \tag{4.18}$$

$$\sin^2(\alpha) + cos^2(\alpha) = 1. \tag{4.19}$$

The Jacobian matrix is derived using (2.21). As an effect of having three revolute joints, the Jacobian forms a 6×3 matrix. Using (4.14) - (4.16), the various parameters in (2.21) are found to be

$$\begin{aligned} \mathbf{z}_0 &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, & \mathbf{z}_1 = \mathbf{z}_2 &= \begin{bmatrix} s_1 \\ -c_1 \\ 0 \end{bmatrix}, \\ \mathbf{p}_0 &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, & \mathbf{p}_1 &= \begin{bmatrix} 0 \\ 0 \\ l_1 \end{bmatrix}, \\ \mathbf{p}_2 &= \begin{bmatrix} l_2 c_1 c_2 \\ l_2 c_2 s_1 \\ l_2 s_2 + l_1 \end{bmatrix}, & \mathbf{p}_3 &= \begin{bmatrix} l_3 c_1 c_{23} + l_2 c_1 c_2 \\ l_3 s_1 c_{23} + l_2 s_1 c_2 \\ l_3 s_{23} + l_2 s_2 + l_1 \end{bmatrix}. \end{aligned} \tag{4.20}$$

As a result, the Jacobian $\mathbf{J}_{\dot{p}}^{(i)}$ and $\mathbf{J}_{\omega}^{(i)}$ are

$$\mathbf{J}_{\dot{p}}^{(1)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \mathbf{J}_{\omega}^{(1)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \tag{4.21}$$

$$\mathbf{J}_{\dot{p}}^{(2)} = \begin{bmatrix} -\frac{1}{2} l_2 c_2 s_1 & -\frac{1}{2} l_2 c_1 s_2 & 0 \\ \frac{1}{2} l_2 c_{23} & -\frac{1}{2} l_2 s_1 s_2 & 0 \\ 0 & \frac{1}{2} l_2 c_2 & 0 \end{bmatrix}, \qquad \mathbf{J}_{\omega}^{(2)} = \begin{bmatrix} 0 & s_1 & 0 \\ 0 & -c_1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \tag{4.22}$$

$$\mathbf{J}_{\dot{p}}^{(3)} = \begin{bmatrix} -\frac{1}{2} l_3 s_1 c_{23} - l_2 c_2 s_1 & -\frac{1}{2} l_3 c_1 s_{23} - l_2 c_1 s_2 & -\frac{1}{2} l_3 c_1 s_{23} \\ \frac{1}{2} l_3 c_1 c_{23} + l_2 c_{23} & -\frac{1}{2} l_3 s_1 s_{23} - l_2 s_1 s_2 & -\frac{1}{2} l_3 s_1 s_{23} \\ 0 & \frac{1}{2} l_3 c_{23} + l_2 c_2 & \frac{1}{2} l_3 c_{23} \end{bmatrix}, \tag{4.23}$$

$$\mathbf{J}_{\omega}^{(3)} = \begin{bmatrix} 0 & s_1 & s_1 \\ 0 & -c_1 & -c_1 \\ 1 & 0 & 0 \end{bmatrix}. \tag{4.24}$$

4.2.3 Kinetic Energy

In order to find $\mathbf{D}(\mathbf{q})$, Equations (4.21)-(4.24) are inserted into (2.38). $\mathbf{D}(\mathbf{q})$ is split up into 3 parts, and are later sum up to the total $\mathbf{D}(\mathbf{q})$. The rotation matrix $\mathbf{R}_i^0$ are found from the rotation part of the transformation matrix $\mathbf{T}_i^0$. The first contribution to $\mathbf{D}(\mathbf{q})$ is

$$\mathbf{D}_1(\mathbf{q}) = m_1 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} +$$

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} c_1 & 0 & s_1 \\ s_1 & 0 & -c_1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} I_{1x} & 0 & 0 \\ 0 & I_{1y} & 0 \\ 0 & 0 & I_{1z} \end{bmatrix} \begin{bmatrix} c_1 & s_1 & 0 \\ 0 & 0 & 1 \\ s_1 & -c_1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} I_{1y} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{4.25}$$

The same procedure is used to find $\mathbf{D}_2(\mathbf{q})$

$$\begin{aligned} \mathbf{D}_2(\mathbf{q}) = m_2 & \begin{bmatrix} -\frac{1}{2}l_2c_2s_1 & \frac{1}{2}l_2c_{23} & 0 \\ -\frac{1}{2}l_2c_1s_2 & -\frac{1}{2}l_2s_1s_2 & \frac{1}{2}l_2c_2 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} -\frac{1}{2}l_2c_2s_1 & -\frac{1}{2}l_2c_1s_2 & 0 \\ \frac{1}{2}l_2c_{23} & -\frac{1}{2}l_2s_1s_2 & 0 \\ 0 & \frac{1}{2}l_2c_2 & 0 \end{bmatrix} \\ & + \begin{bmatrix} 0 & 0 & 1 \\ s_1 & -c_1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} c_1c_2 & -c_1s_2 & -s_1 \\ c_2s_1 & -s_1s_2 & c_1 \\ -s_2 & -c_2 & 0 \end{bmatrix} \begin{bmatrix} I_{2x} & 0 & 0 \\ 0 & I_{2y} & 0 \\ 0 & 0 & I_{2z} \end{bmatrix} \\ & \begin{bmatrix} c_1c_2 & c_2s_1 & -s_2 \\ -c_1s_2 & -s_1s_2 & -c_2 \\ -s_1 & c_1 & 0 \end{bmatrix} \begin{bmatrix} 0 & s_1 & 0 \\ 0 & -c_1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \\ & = \begin{bmatrix} m_2(\frac{1}{2}l_2)^2(c_2)^2 + I_{2x}(s_2)^2 + I_{2y}(c_2)^2 & 0 & 0 \\ 0 & m_2\frac{1}{2}l_2)^2 + I_{2z} & 0 \\ 0 & 0 & 0 \end{bmatrix}. \end{aligned} \tag{4.26}$$

The next part is more complex, so it is divided into the the linear kinetic part $\mathbf{D}_{3\dot{p}}$ and the rotation part $\mathbf{D}_{3\omega}$. They can be written as

$$\begin{aligned} \mathbf{D}_{3\dot{p}}(\mathbf{q}) = m_3 & \begin{bmatrix} -\frac{1}{2}l_3s_1c_{23} - l_2c_2s_1 & \frac{1}{2}l_3c_1c_{23} + l_2c_{23} & 0 \\ -\frac{1}{2}l_3c_1s_{23} - l_2c_1s_2 & -\frac{1}{2}l_3s_1s_{23} - l_2s_1s_2 & \frac{1}{2}l_3c_{23} + l_2c_2 \\ -\frac{1}{2}l_3c_1s_{23} & -l_3\frac{1}{2}s_1s_{23} & \frac{1}{2}l_3c_{23} \end{bmatrix} \\ & \begin{bmatrix} -\frac{1}{2}l_3s_1c_{23} - l_2c_2s_1 & -\frac{1}{2}l_3c_1s_{23} - l_2c_1s_2 & -\frac{1}{2}l_3c_1s_{23} \\ \frac{1}{2}l_3c_1c_{23} + l_2c_{23} & -\frac{1}{2}l_3s_1s_{23} - l_2s_1s_2 & -l_3\frac{1}{2}s_1s_{23} \\ 0 & \frac{1}{2}l_3c_{23} + l_2c_2 & \frac{1}{2}l_3c_{23} \end{bmatrix} \\ & = \begin{bmatrix} m_3(\frac{1}{2}c_{23} + l_2c_2)^2 & 0 & 0 \\ 0 & m_3((\frac{1}{2}l_3)^2 + (l_2)^2 + l_2l_3c_3) & m_3((\frac{1}{2}l_3)^2 + \frac{1}{2}l_3l_2c_3) \\ 0 & m_3((\frac{1}{2}l_3)^2 + \frac{1}{2}l_3l_2c_3) & \frac{1}{2}m_3l_3 \end{bmatrix}, \end{aligned} \tag{4.27}$$

$$
\begin{aligned}
\mathbf{D}_{3\omega}(\mathbf{q}) = &\begin{bmatrix} 0 & 0 & 1 \\ s_1 & -c_1 & 0 \\ s_1 & -c_1 & 0 \end{bmatrix} \begin{bmatrix} c_1c_2c_3 - c_1s_2s_3 & c_1c_2s_3 - c_1c_3s_2 & -s_1 \\ s_1c_2c_3 - s_1s_2s_3 & s_1s_3c_2 - s_1s_2c_3 & c_1 \\ -s_2c_3 - c_2s_3 & s_2s_3 - c_2c_3 & 0 \end{bmatrix} \begin{bmatrix} I_{3x} & 0 & 0 \\ 0 & I_{3y} & 0 \\ 0 & 0 & I_{3z} \end{bmatrix} \\
&\begin{bmatrix} c_1c_2c_3 - c_1s_2s_3 & s_1c_2c_3 - s_1s_2s_3 & -s_2c_3 - c_2s_3 \\ c_1c_2s_3 - c_1c_3s_2 & s_1s_3c_2 - s_1s_2c_3 & s_2s_3 - c_2c_3 \\ -s_1 & c_1 & 0 \end{bmatrix} \begin{bmatrix} 0 & s_1 & s_1 \\ 0 & -c_1 & -c_1 \\ 1 & 0 & 0 \end{bmatrix} \\
= &\begin{bmatrix} (s_{23})^2 I_{3x} + (c_{23})^2 I_{3y} & 0 & 0 \\ 0 & I_{3z} & I_{3z} \\ 0 & I_{3z} & I_{3z} \end{bmatrix}. \qquad (4.28)
\end{aligned}
$$

In summary, $\mathbf{D}(\mathbf{q})$ can be written as

$$
\begin{aligned}
\mathbf{D}(\mathbf{q}) &= \mathbf{D}_1(\mathbf{q}) + \mathbf{D}_2(\mathbf{q}) + \mathbf{D}_{3\dot{p}}(\mathbf{q}) + \mathbf{D}_{3\omega}(\mathbf{q}) & (4.29) \\
&= \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{bmatrix}, & (4.30)
\end{aligned}
$$

where

$$
\begin{aligned}
d_{11} =& I_{1y} + I_{2x}(s_2)^2 + I_{2y}(c_2)^2 + I_{3x}(s_{23})^2 + I_{3y}(c_{23})^2 \\
& + m_2(\frac{1}{2}l_2)^2(c_2)^2 + m_3(\frac{1}{2}c_{23} + l_2c_2)^2, \\
d_{12} =& d_{21} = 0, \\
d_{13} =& d_{31} = 0, \qquad (4.31) \\
d_{22} =& I_{2z} + I_{3z} + m_2(\frac{1}{2}l_2)^2 + m_3((\frac{1}{2}l_3)^2 + (l_2)^2 + l_2l_3c_3), \\
d_{23} =& d_{32} = I_{3z} + m_3((\frac{1}{2}l_3)^2 + \frac{1}{2}l_3l_2c_3), \\
d_{33} =& I_{3z} + m_3(\frac{1}{2}l_3)^2.
\end{aligned}
$$

Furthermore, this matrix is used to derive the Coriolis and centripetal matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$. The elements in the Coriolis and centripetal matrix

$$
\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix}, \qquad (4.32)
$$

are given by (2.49). They are found to be

$$
\begin{aligned}
c_{11} =& \frac{1}{2}(\frac{\partial d_{11}}{\partial q_2}\dot{q}_2 + \frac{\partial d_{11}}{\partial q_3}\dot{q}_3) \\
=& (s_2c_2(I_{2x} - I_{2y}) + c_{23}s_{23}(I_{3x} - I_{3y}) - m_2l_{2c}^2c_2s_2 - m_3(l_{3c}c_{23} + l_2c_2)(l_{3c}s_{23} + l_2s_2))\dot{q}_2 \\
&+ (c_{23}s_{23}(I_{3x} - I_{3y}) - m_3l_{3c}s_{23}(l_{3c}c_{23} + l_2c_2))\dot{q}_3, \\
c_{12} =& \frac{1}{2}\frac{\partial d_{11}}{\partial q_2}\dot{q}_1 \\
=& (s_2c_2(I_{2x} - I_{2y}) + c_{23}s_{23}(I_{3x} - I_{3y}) - m_2l_{2c}^2c_2s_2 - m_3(l_{3c}c_{23} + l_2c_2)(l_{3c}s_{23} + l_2s_2))\dot{q}_1, \\
c_{13} =& \frac{1}{2}\frac{\partial d_{11}}{\partial q_3}\dot{q}_1 = (c_{23}s_{23}(I_{3x} - I_{3y}) - m_3l_{3c}s_{23}(l_{3c}c_{23} + l_2c_2))\dot{q}_1, \\
c_{21} =& -\frac{1}{2}\frac{\partial d_{11}}{\partial q_2}\dot{q}_1 = -c_{12}, \\
c_{22} =& \frac{1}{2}\frac{\partial d_{22}}{\partial q_3}\dot{q}_3 = -\frac{1}{2}m_3l_2l_3s_3\dot{q}_3, \\
c_{23} =& \frac{1}{2}(\frac{\partial d_{22}}{\partial q_3}\dot{q}_2 + 2\frac{\partial d_{23}}{\partial q_3}\dot{q}_3) = -\frac{1}{2}m_3l_2l_3s_3\dot{q}_2 - m_3l_{3c}l_2s_3\dot{q}_3, \\
c_{31} =& -\frac{1}{2}\frac{\partial d_{11}}{\partial q_3}\dot{q}_1 = -c_{13}, \\
c_{32} =& -\frac{1}{2}\frac{\partial d_{22}}{\partial q_3}\dot{q}_2 = \frac{1}{2}m_3l_2l_3s_3\dot{q}_2, \\
c_{33} =& 0.
\end{aligned}
$$

Here, l_{ic} represents $l_i/2$.

4.2.4 Potential Energy

The potential energy equation (2.41) is a function of the gravity. It is written as

$$
P = m_1\frac{l_1}{2}g + m_2g(\frac{l_2}{2}s_2 + l_1) + m_3g(\frac{l_3}{2}s_{23} + l_2s_2 + l_1). \tag{4.33}
$$

The $\mathbf{g}(\mathbf{q})$ is found from (2.45). The result is

$$
\mathbf{g}(\mathbf{q}) = \begin{bmatrix} 0 \\ m_2gl_{2c}c_2 + m_3gl_{3c}c_{23} + l_2c_2 \\ m_3gl_{3c}c_{23} \end{bmatrix}. \tag{4.34}
$$

In summary, each term in (2.48) have been accounted for. The system of equations are obtained if $\mathbf{D}(\mathbf{q})$, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ and $\mathbf{g}(\mathbf{q})$ are inserted into (2.48).

4.2.5 Skew-symmetry

As stated in (2.50), the $\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}})$ matrix is supposed to skew symmetric. In other words

$$\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = -\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}})^T. \tag{4.35}$$

Equation (2.50) yields

$$\mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} 0 & -\frac{\partial d_{11}}{\partial q_2}\dot{q}_1 & -\frac{\partial d_{11}}{\partial q_3}\dot{q}_1 \\ \frac{\partial d_{11}}{\partial q_2}\dot{q}_1 & 0 & -\frac{\partial d_{22}}{\partial q_3}\dot{q}_2 - \frac{\partial d_{23}}{\partial q_3}\dot{q}_3 \\ \frac{\partial d_{11}}{\partial q_3}\dot{q}_1 & \frac{\partial d_{22}}{\partial q_3}\dot{q}_2 + \frac{\partial d_{23}}{\partial q_3}\dot{q}_3 & 0 \end{bmatrix}. \tag{4.36}$$

The matrix in (4.36) is obviously skew symmetric. Hence, it is reason to believe that procedure has been carried out in a correct manner.

4.2.6 Vessel Crane Dynamics

The vessel-crane model is derived by introducing a movable foundation, which in our case can be seen as a 6 DOF massless joint. The joint represent the vessel's ability to surge, sway, heave, roll, pitch and yaw about $\{B\}$. For the vessel-crane dynamics, the system of equations will be dependent on the vessel's linear and angular velocity

$$\boldsymbol{\nu} = \begin{bmatrix} u & v & w & p & q & r \end{bmatrix}, \tag{4.37}$$

as well as the position and orientation

$$\boldsymbol{\eta} = \begin{bmatrix} x & y & z & \phi & \theta & \psi \end{bmatrix}. \tag{4.38}$$

The transformation matrix from the sea $\{N\}$ to the vessel $\{B\}$ can be written as

$$\mathbf{T}_B^N = \begin{bmatrix} c_\psi c_\theta & -s_\psi c_\phi + c_\psi s_\theta s_\phi & s_\psi s_\phi + c_\psi c_\phi s_\theta & x \\ s_\psi c_\theta & c_\psi c_\phi + s_\phi s_\theta s_\psi & -c_\psi s_\phi + s_\theta s_\psi c_\phi & y \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi & z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.39}$$

In addition, the crane is placed on the aft port side of the vessel. This is done by multiplying H_B^N by

$$\mathbf{T}_0^B = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.40}$$

where

$$\begin{bmatrix} p_x & p_y & p_z \end{bmatrix}^T = \begin{bmatrix} -20 & 8 & 10 \end{bmatrix}^T . \tag{4.41}$$

This vector represents where the crane is mounted with regards to $\{B\}$. Furthermore, the vessel-crane transformation matrices are found using $\mathbf{T}_N^B$, $\mathbf{T}_B^0$ and $\mathbf{T}_0^3$. Futhermore, the Euler-Lagrange method is employed in order to find the vessel-crane system of equations.

A Matlab script performs the needed operations and stores the matrices in a Matlab function, optimized with respect to performance. These matrices are used by the Simulink model in the calculation of the crane dynamics. They are also used by the controller and the inverse kinematics. This will be described more closely in Chapter 6. The vessel crane model will be used for simulations in Chapters 5 and 6.

4.3 Simulink Model Overview

In this section, the implementation of the vessel-crane-cable model will be described. There will also be given an overview of the total system. The model has been built by the Simulink software, which is a widely used program to implement dynamical systems using differential equations. Simulink code consist of several blocks that have been put together in order to represent differential equations. In addition to the standard Simulink blocks, there exist various toolboxes that utilize different features.

4.3.1 Crane Model

Figure 4.2 shows the block diagram of the overall system, together with a more detailed figure of the crane model. A similar figure will be shown in the later chapters as well, but with focus on the other parts of the Simulink model. The block that is going to be explained will always be represented with a red block, as for the crane in this case.

The detailed crane block diagram shows that (2.48) has been rearranged in order to calculate the joint angles. The inverse block invokes the inverse method in Simulink, and the Matrix Mult. block performs a matrix multiplication. The $\frac{1}{s}$ block represent an integration. In the figure there is also a torque block. This is the crane input, which is provided by the proportional-derivative controller (PD) with gravity compensation ,denoted as PD with Gravity FF. For the vessel-crane dynamics, the matrices will also be dependent of $\boldsymbol{\nu}$ and $\boldsymbol{\eta}$.

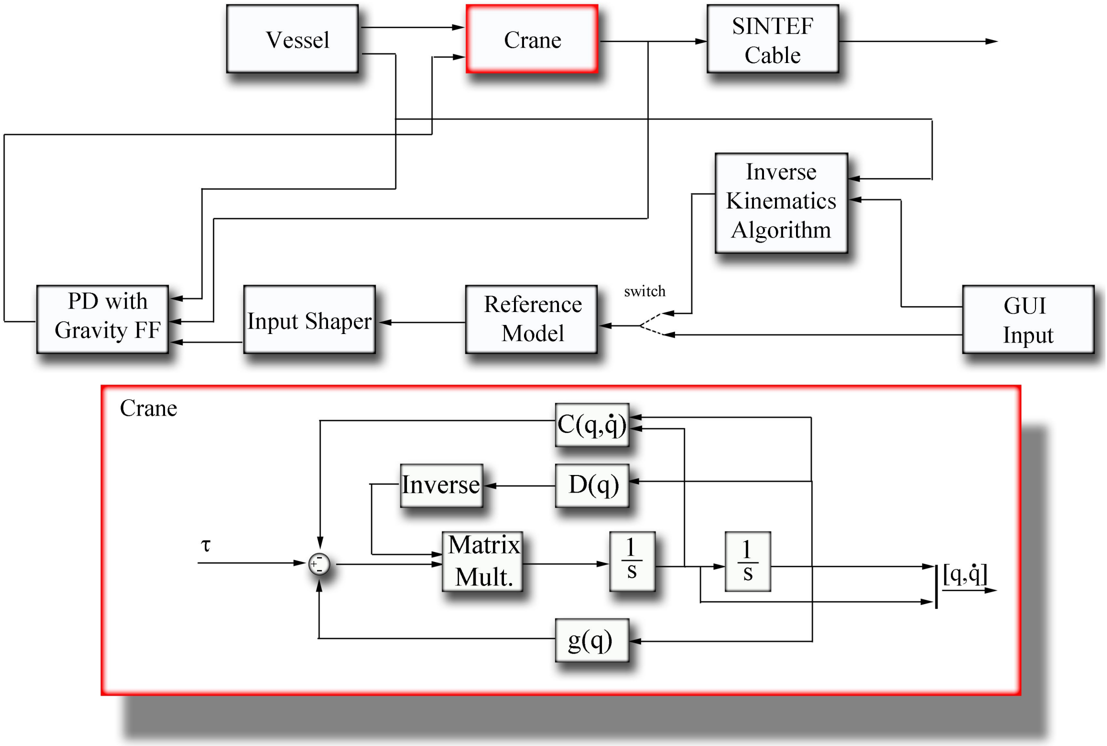


Figure 4.2: Overview of the Simulink model, with the crane dynamics shown in detail.

4.3.2 Cable Model

The cable model shown in Figure 4.3 is provided by SINTEF, which is the largest independent research organisation in Scandinavia. The cable is made of a software tool called FhSim, which is tool for describing and simulating real world systems that may be formulated as ordinary differential equations.

The differential equations are defined by one or several simObjects. These may be implemented as ordinary pre-defined C++ classes or as user defined modules.The cable model implemented here consist of 5 simObjects, where all the elements are of type CCableElDynStiff. Each element is regarded as a straight segment, responsible for calculating the forces acting on each of its ends, as well as the mass associated with each end node. The forces and mass from each element is added to the corresponding nodes of the cable, and the acceleration of each node is calculated in the OdeFcn (SINTEF, 2010). A 30 tonnes payload is connected to the end of the cable, which is supposed to represent a fully loaded ISO standard shipping container.

Each element in the cable model is influenced by

- Internal stiffness,
- Internal damping,

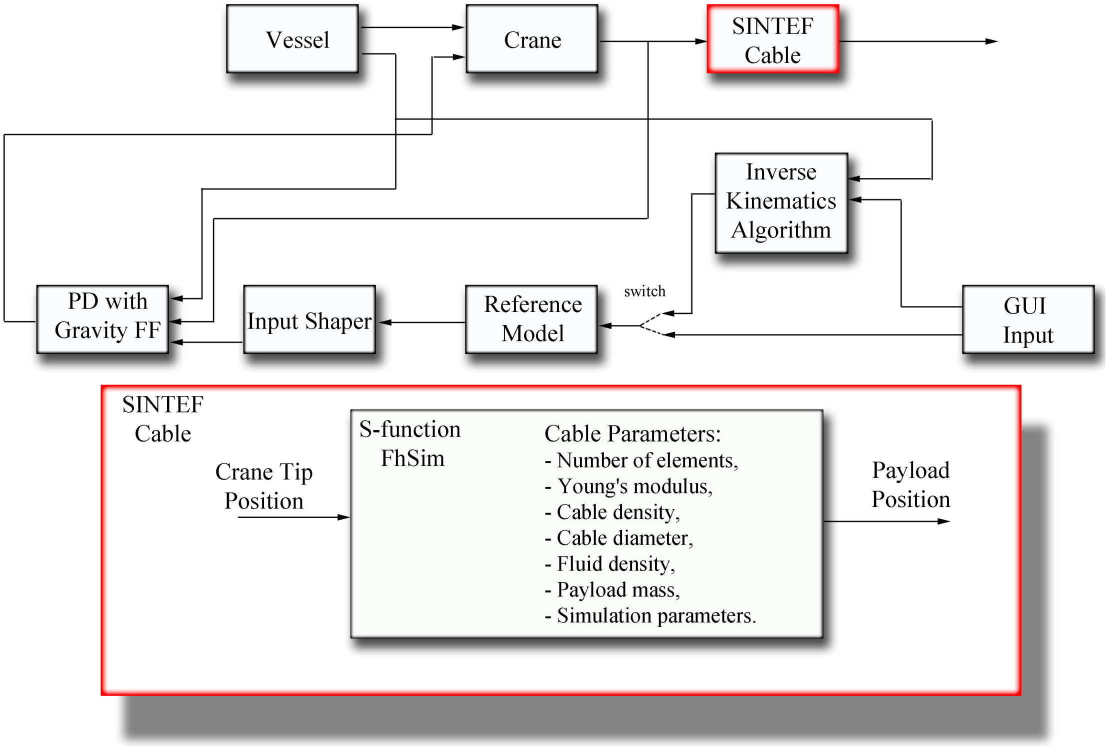


Figure 4.3: Overview of the Simulink model, with the SINTEF cable model shown in detail.

- Hydrodynamic normal and tangential forces,
- Buoyancy and gravity,
- Added mass.

The cable model is coded in C++, and implemented in the Simulink model using a s-function block.

4.3.3 Vessel Model

The vessel block shown in Figure 4.4 generates vessel motion. Due to environmental disturbance, the vessel will experience surge, sway, heave, roll, pitch and yaw. The crane, which is placed at the aft port side of the vessel, will be influenced by the vessel motion. In addition, the vessel roll, pitch and yaw will induce extra surge, sway and heave crane motion, due to the off center placement of the crane.

In this thesis the vessel surge and sway will be assumed to be taken care by the DP system. This is only partially true because the main objective of the DP system is to take care of the slowly varying motion, not the highly fluctuating force. The largest slowly varying

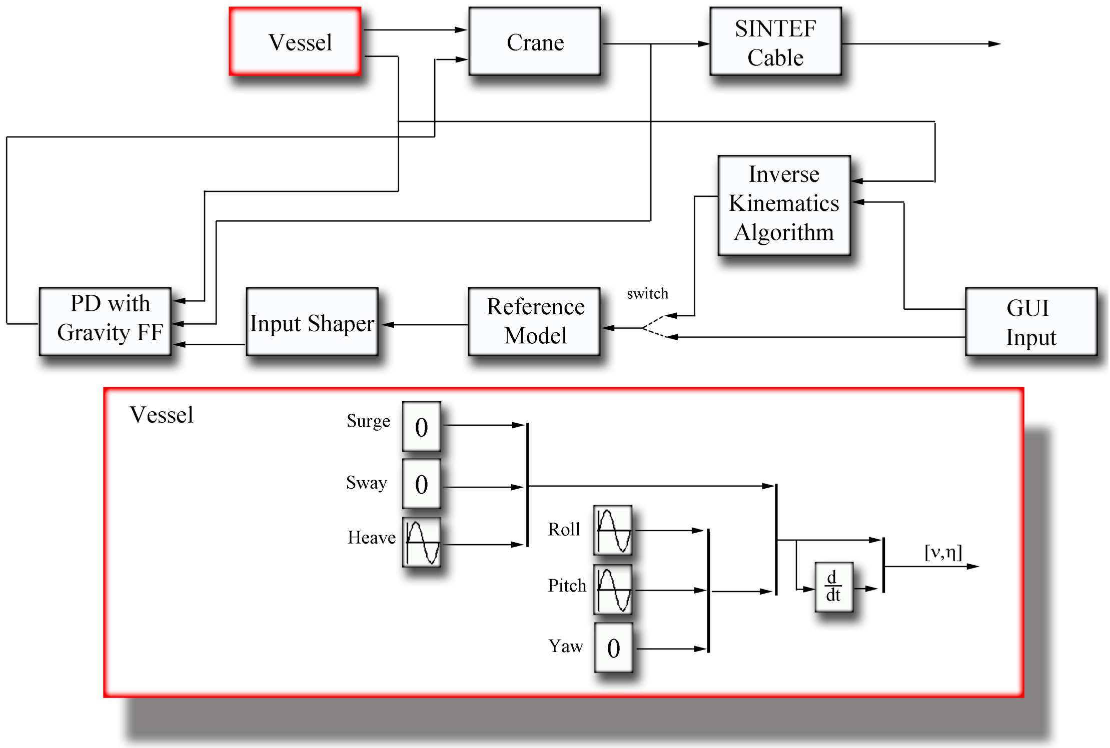


Figure 4.4: Overview of the Simulink model, with the vessel model shown in detail.

Sea State	Heave Amp [m]	Roll Amp [deg]	Pitch Amp [deg]	Heave Per. [s]	Roll Per. [s]	Pitch Per. [s]
2	0.1	2	0.5	6	15	6
4	0.5	5	1	7	16	7
6	3.5	13	3	8	17	8

Table 4.2: Vessel motion description, inspired by Love et al. (2003).

contributer is usually current, while members of the highly fluctuation force are wind gusts and waves. Furthermore, yaw amplitudes are normally small if the the vessel has the bow pointing towards the incoming waves. For this reason, yaw motion has been neglected.

The vessel block seen in Figure 4.4 consist of several sine input blocks. Each block represent one of the vessel's DOF and each has a fixed amplitude and period. Data about the amplitudes and periods are shown in Table 2.1. The vessel data is representative for a vessel similar to the Seven Sisters. Vessel data are presented for sea state 2, 4 and 6. Definitions of the three sea states are shown in Table 4.3.

Sea State	Description	Wave height [m]	Characteristics
2	Small waves becoming larger	0.1-0.5	Smooth
4	Moderate waves, taking a more pronounced long form	1.25-2.5	Moderate
6	White crests are more extensive everywhere	4-6	Rough

Table 4.3: Sea state definitions, inspired by Love et al. (2003).

Chapter 5

Verification of Crane Dynamics

The main goal of this chapter, is to check whether or not the Simulink model and the SimMechanics model produce the same output. First, the mathematical model that was derived in 4.2, will be compared to a Matlab generated model. Second, a SimMechanics model that was developed during the author's project assignment, will be used to verify the vessel-crane model. Last, there will be a discussion based on the results that have been presented.

5.1 Mathematical Verification

To verify the equations of motion in Chapter 4.2, a Matlab script has been made. The script generates the equations of motion using the symbolic math toolbox. It was developed by inspiration of Love et al. (2003). In Chapter 4.2, it shown that there is reason to believe that the calculation were correct, due to the proven skew symmetry of (4.36).

The Matlab script uses the Euler-Lagrange formulation, which was described in Chapter 2. The Matlab script produces the following elements of $\mathbf{g}(\mathbf{q})$ and $\mathbf{D}(\mathbf{q})$

$$\begin{aligned} g_1 &= 0, \\ g_2 &= \frac{1}{2} g \left(l_3 m_3 \cos\left(q_2 + q_3\right) + l_2 m_2 \cos\left(q_2\right) + 2 l_2 m_3 \cos\left(q_2\right) \right), \\ g_3 &= \frac{}{2} l_3 g m_3 \cos\left(q_2 + q_3\right), \end{aligned} \tag{5.1}$$

$$
\begin{aligned}
d_{11} &= I_{1y} + I_{2x} + I_{3x} - I_{3x}c_{23}^2 + I_{3y}c_{23}^2 - I_{2x}c_2^2 + I_{2y}c_2^2 \\
&\quad + m_3(\frac{l_3}{2})^2 c_{23}^2 + m_2(\frac{l_2}{2})^2(c_2)^2 + m_3 l_2^2 (c_2)^2 + 2m_3 l_2 \frac{l_3}{2} c_{23} c_2, \\
d_{12} &= d_{21} = 0, \\
d_{13} &= d_{31} = 0, \\
d_{22} &= m_3 l_2^2 + 2m_3 l_2 l_3 c_3 + m_2 {\frac{l_2}{2}}^2 + m_3 {\frac{l_3}{2}}^2 + I_{2z} + I_{3z}, \\
d_{23} &= d_{32} = m_3 {\frac{l_3}{2}}^2 + m_3 l_2 \frac{l_3}{2} c_3 + I_{3z}, \\
d_{33} &= m_3 {\frac{l_3}{2}}^2 + I_{3z}.
\end{aligned}
\tag{5.2}
$$

(5.3)

The Coriolis and centripetal contribution

$$
\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix},
\tag{5.4}
$$

are

$$
\begin{aligned}
c_1 &= c_{11}\dot{q}_1 + c_{12}\dot{q}_2 + c_{13}\dot{q}_3 \\
&= -\frac{1}{4}(\dot{q}_1(4I_{2y}\dot{q}_2 \sin(2q_2) - 4I_{2x}\dot{q}_2 \sin(2q_2) \\
&- 4I_{3x}\dot{q}_2 \sin(2q_2 + 2q_3) - 4I_{3x}\dot{q}_3 \sin(2q_2 + 2q_3) \\
&+ 4I_{3y}\dot{q}_2 \sin(2q_2 + 2q_3) + 4I_{3y}\dot{q}_3 \sin(2q_2 + 2q_3) \\
&+ {l_2}^2 m_2 \dot{q}_2 \sin(2q_2) + 4{l_2}^2 m_3 \dot{q}_2 \sin(2q_2) \\
&+ {l_3}^2 m_3 \dot{q}_2 \sin(2q_2 + 2q_3) + {l_3}^2 m_3 \dot{q}_3 \sin(2q_2 + 2q_3) \\
&+ 2l_2 l_3 m_3 \dot{q}_3 \sin(q_3) + 4l_2 l_3 m_3 \dot{q}_2 \sin(2q_2 + q_3) \\
&+ 2l_2 l_3 m_3 \dot{q}_3 \sin(2q_2 + q_3))),
\end{aligned}
\tag{5.5}
$$

$$\begin{aligned} c_2 &= c_{21}\dot{q}_1 + c_{22}\dot{q}_2 + c_{23}\dot{q}_3 \\ &= \frac{1}{2} I_{2y}\dot{q}_1^2 \sin\left(2q_2\right) - \frac{1}{2} I_{2x}\dot{q}_1^2 \sin\left(2q_2\right) \\ &- \frac{1}{2} I_3 x\dot{q}_1^2 \sin\left(2q_2 + 2q_3\right) + \frac{1}{2} I_{3y}\dot{q}_1^2 \sin\left(2q_2 + 2q_3\right) \\ &+ \frac{1}{8} {l_2}^2 m_2\dot{q}_1^2 \sin\left(2q_2\right) + \frac{1}{2} {l_2}^2 m_3\dot{q}_1^2 \sin\left(2q_2\right) \\ &+ \frac{1}{8} {l_3}^2 m_3\dot{q}_1^2 \sin\left(2q_2 + 2q_3\right) - \frac{1}{2} l_2 l_3 m_3\dot{q}_3^2 \sin\left(q_3\right) \\ &+ \frac{1}{2} l_2 l_3 m_3\dot{q}_1^2 \sin\left(2q_2 + q_3\right) - l_2 l_3 m_3\dot{q}_2\dot{q}_3 \sin\left(q_3\right), \end{aligned} \tag{5.6}$$

$$\begin{aligned} c_3 &= c_{31}\dot{q}_1 + c_{32}\dot{q}_2 + c_{33}\dot{q}_3 \\ &= \frac{1}{2} I_{3y}\dot{q}_1^2 \sin\left(2q_2 + 2q_3\right) - \frac{1}{2} I3x\dot{q}_1^2 \sin\left(2q_2 + 2q_3\right) \\ &+ \frac{1}{8} {l_3}^2 m_3\dot{q}_1^2 \sin\left(2q_2 + 2q_3\right) + \frac{1}{4} l_2 l_3 m_3\dot{q}_1^2 \sin\left(q_3\right) \\ &+ \frac{1}{2} l_2 l_3 m_3\dot{q}_2^2 \sin\left(q_3\right) + \frac{1}{4} l_2 l_3 m_3\dot{q}_1^2 \sin\left(2q_2 + q_3\right). \end{aligned} \tag{5.7}$$

Furthermore, it is possible to simplify (5.5) - (5.7) using

$$s_{12}c_{12} = \frac{1}{2}sin(2q_1 + 2q_2), \tag{5.8}$$

$$s_{12}c_1 = \frac{1}{2}(sin(2q_1 + q_2) + sin(q_2)), \tag{5.9}$$

$$c_{12}s_1 = \frac{1}{2}(sin(2q_1 + q_2) - sin(q_2)), \tag{5.10}$$

$$c_2 s_2 = \frac{1}{2}sin(2q_2). \tag{5.11}$$

By employing (5.8)-(5.11) on (5.5) - (5.7), the inertia matrix, Coriolis and centripetal matrix and gravity matrix becomes identical to the ones that were found in Chapter 4.2.

5.2 Simulation Verification

In this section, simulation results from the Simulink model will be compared to the simulation results from the SimMechanics model. The SimMechanics environment is a tool for modeling mechanical systems within the Simulink environment. Furthermore, the Simulink model that has been used, has incorporated vessel-crane interaction. This has been done by modifying the Matlab script to include (4.39 and (4.40). In addition, the vessel motion amplitudes, described in Section 4.3.3, has been set to zero. That means that the vessel has not been exposed to waves during simulations. This was done because the SimMechanics model does not include vessel crane interaction.

During simulations, both cranes were exposed to a step input after 20 seconds. The initial configuration of the crane was set to

$$\mathbf{q}_{initial} = \begin{bmatrix} 0 & 0 & -\frac{\pi}{2} \end{bmatrix}^T , \tag{5.12}$$

and the step input was

$$\mathbf{q}_{step} = \begin{bmatrix} \frac{\pi}{2} & \frac{\pi}{4} & -\frac{3\pi}{2} \end{bmatrix}^T . \tag{5.13}$$

Figures 5.1 - 5.3 show the joint variables for the Simulink model and the SimMechanics model. It can be seen that the models respond almost identical when they are exposed to the same step input. In fact, there is not possible to see any difference in orientation at all.

The torque that is fed into the crane model, are shown in Figures 5.4 - 5.6. The torque is calculated by the controllers, which for the SimMechanics model is a built-in controller. The Simulink model uses a PD controller with gravity compensation. The figures show that torque given from the controllers are almost identical. There can be seen some differences when the crane accelerates. These differences are small and appear in a short period of time.

Figures 5.7 and 5.8 show the difference between the Simulink and SimMechanics torque and orientation. Figure 5.7 shows that the maximum error in orientation is less than 0.01°, which is considered to be very accurate. For the torque, the biggest difference is approximately $5 \cdot 10^5$, which occurs after about 10 and 12 seconds. Besides this, the torque difference is small. The difference in torque is most likely an effect of using two different controllers.

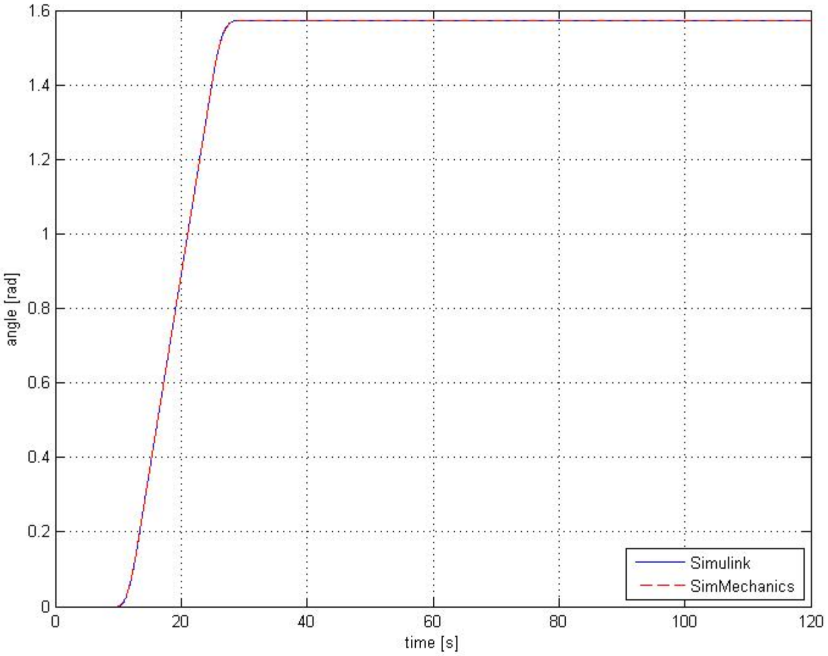


Figure 5.1: Simulink and SimMechanics joint variable q_1.

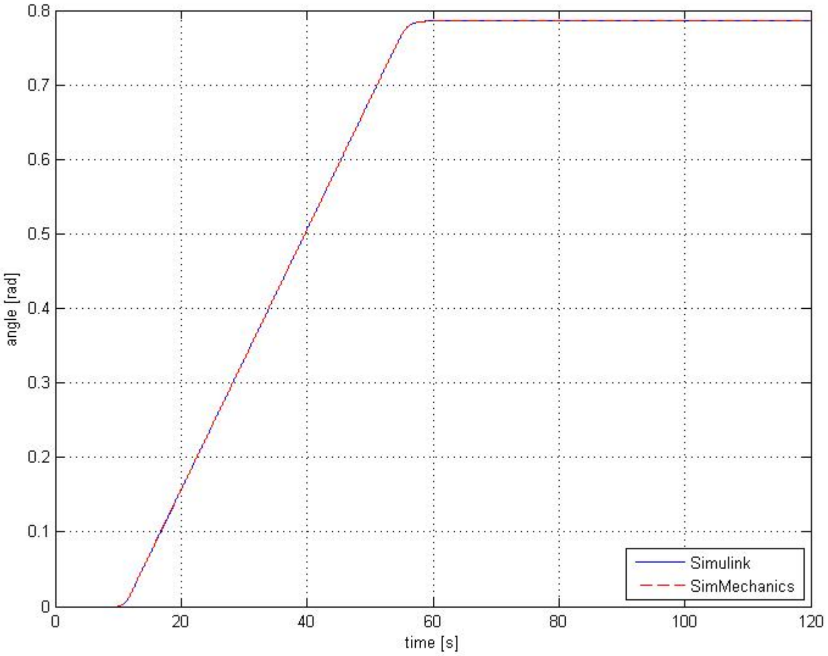


Figure 5.2: Simulink and SimMechanics joint variable q_2.

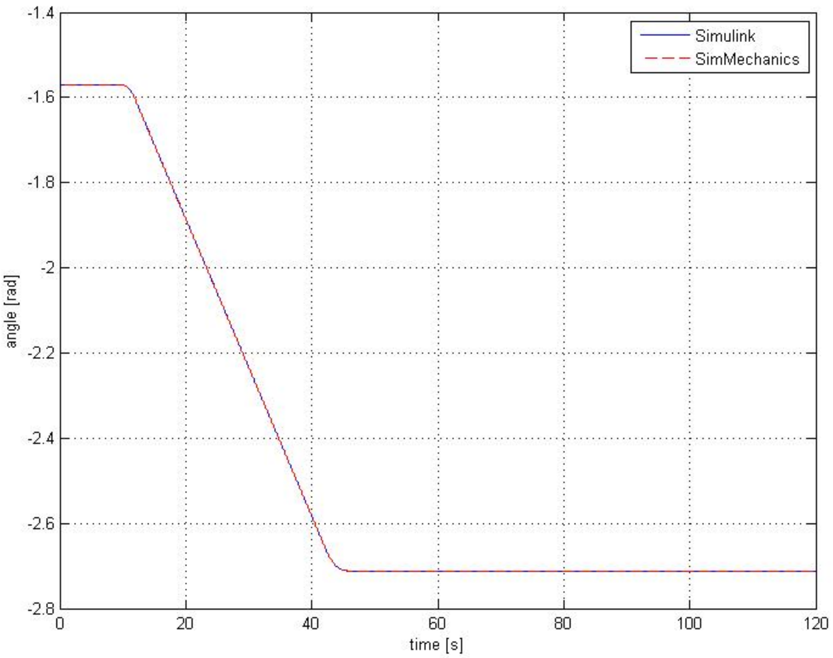


Figure 5.3: Simulink and SimMechanics joint variable q_3.

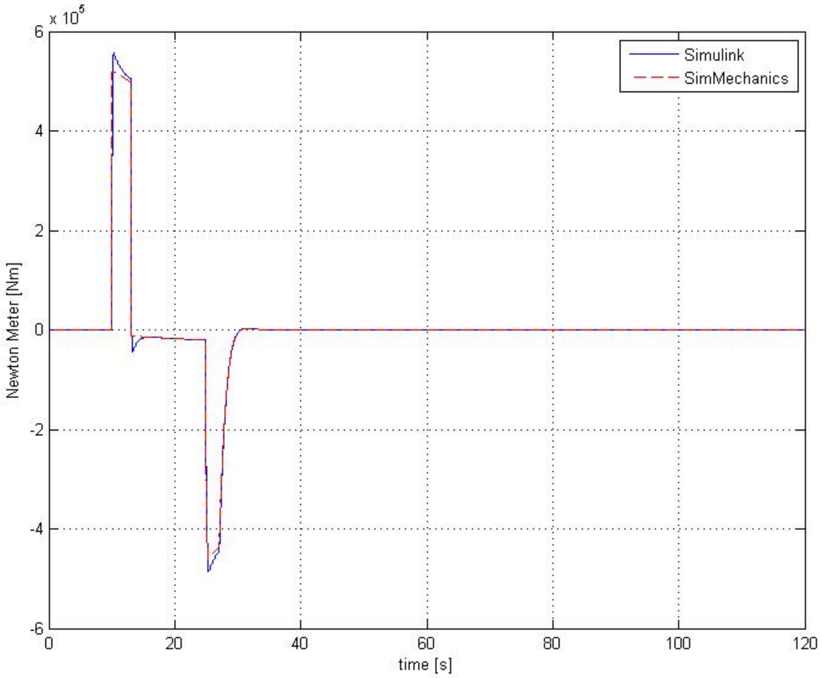


Figure 5.4: Simulink and SimMechanics torque τ_1.

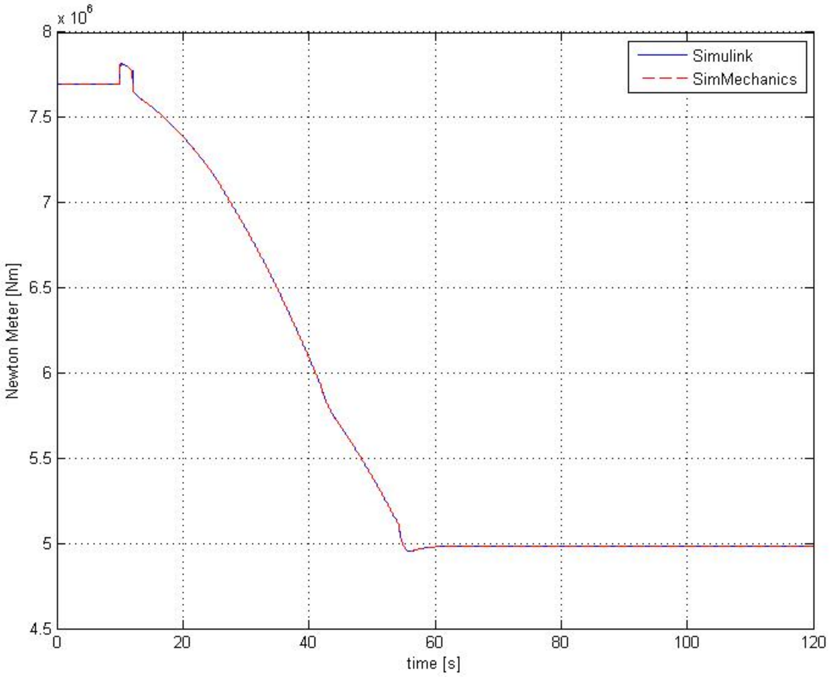


Figure 5.5: Simulink and SimMechanics torque τ_2.

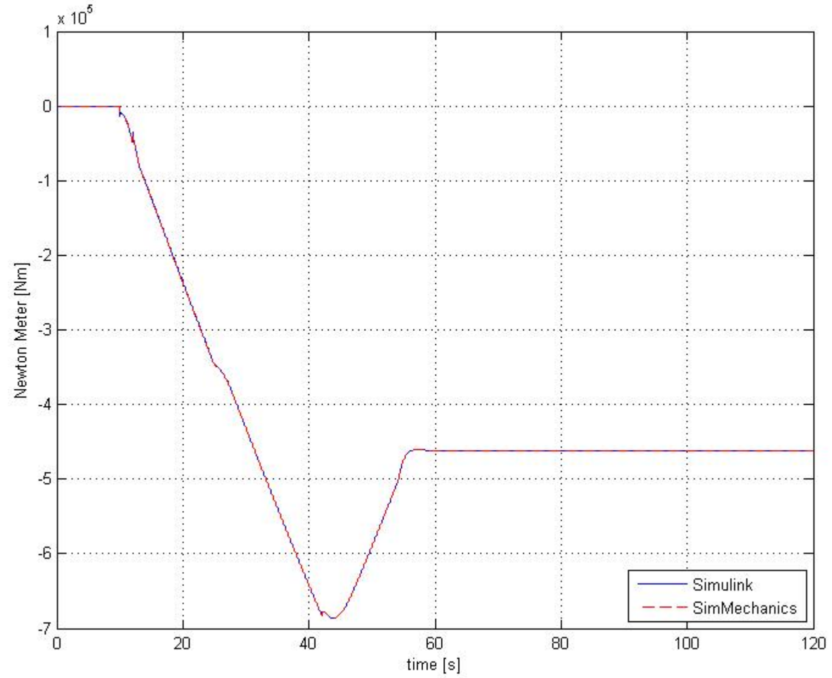


Figure 5.6: Simulink and SimMechanics torque τ_3.

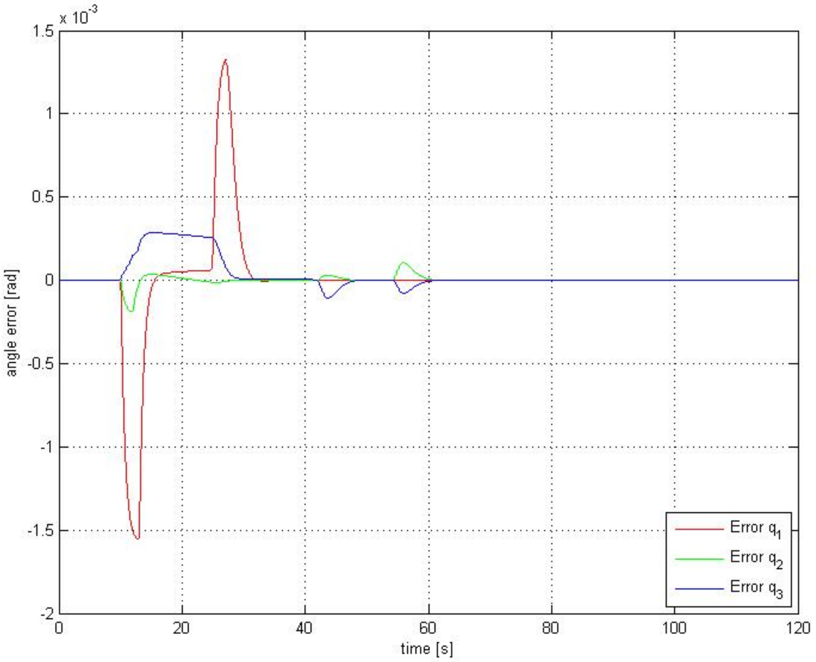


Figure 5.7: Simulink and SimMechanics configuration difference.

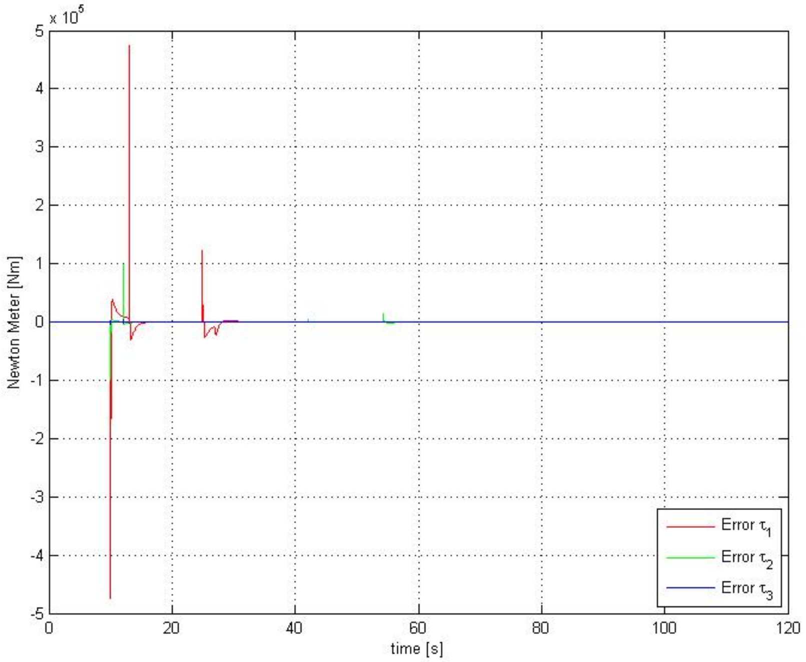


Figure 5.8: Simulink and SimMechanics torque difference.

5.3 Discussion

The mathematical model derived in Chapter 4.2 and the model derived by the Matlab script, turned out to identical. It is therefore reason to believe that the mathematical model, as well as the script generated model, are correct. Consequently, the reliability to vessel-crane dynamics, which is found using the same script (with a slight modification), is strengthened.

The vessel crane Simulink model and the SimMechanics model showed almost identical behavior when they were exposed to the same step input. There were some small differences in the generated torque. However, this was expected since the models use two different controllers.

In conclusion, the vessel-crane model has successfully been verified, and is now ready to perform vessel-crane-cable simulations.

Chapter 6

Control of Offshore Crane

This section describes how the Simulink model is controlled, and how it could be controlled to damp payload oscillations. First, todays's control modes will be explained. Second, today's and future control objectives are accounted for. In order to get the crane to work desirable, one need a reference model that takes the crane dynamics into account. This is described in the third section. The fourth section concerns a workspace control method, while the last section concerns a control shaping algorithm called input shaping.

6.1 Today's Solution

Today's cranes are controlled by the operator from the crane cabin, which is mounted on the left side of the King. The placement gives the operator a good view over the the crane and its surroundings. In addition, an LCD screen based control system gives the crane operator all required information for both crane system and its current operation status.

The operator cabin consists of a control system and several operating panels for easy control of all crane functions. The operator sits in a chair equipped with two joysticks, one on the left hand side, and one on the right hand side. Both joysticks are equipped with multiple buttons that utilize different functions. When the left hand side joystick is pushed out of its neutral position, the crane will either slew or raise/lower the main jib. The right hand side joystick control the knuckle jib and the winches.

The joysticks and the LCD screen have real time communication with the PLC. If a joystick is moved out of its neutral position, the PLC calculates the needed hydraulic flow and which valves that have to be opened or closed in the the main hydraulic valve system. Automatic control functions, such as AHC and AT, are started and stopped by the operator. If one of these functions are activated, algorithms in the PLC automatically calculates the needed winch in and out speed.

6.2 Control Objectives

Modern offshore cranes are able to switch between different control objectives. The most used control objective is the manual operator mode, also called the normal mode. In this mode the operator changes slew, main and knuckle jib angle, as well as he operates the winches. However, there are also more sophisticated control objectives like the AHC and AT. In addition to the existing functions, crane suppliers have an urge to develop new functions that keep their technology attractive. Future functions could be

- Workspace control,
- Control shaping,
- Payload swing compensation.

Instead of use the joysticks to control the king, main and knuckle jib separately, a workspace algorithm could calculate the needed crane movement in x or y direction based on the position of the joysticks. In order to do this, the inverse kinematics has to be calculated and applied. The workspace for the HMC3568 crane, was shown in Figure 2.1. Today, there exist various inverse kinematic algorithms with different pros and cons. Besides being useful for the operator, the algorithm could be optimized with regards to derating, shortest path and collision avoidance. However, one should be aware that singularities and computation delay could contribute to unwanted results.

If the vessel is exposed to waves and wind, there is a possibility that the payload will start to swing. The payload swing compensation algorithm could be split in three parts. Firstly, the pendulum compensation algorithm should remove kinetic and potential energy from the payload, thus reducing the payload oscillations. This could be done by measuring the cable angle and analysing pendulum energy equations. Secondly, the control loop should compensate for the ship induced crane motions. By applying kinematics to the MRU signals, one could calculate the crane tip motion. The resulting motion should be compensated by the control loop. The last part of the algorithm is the normal mode, with a sleight alteration. The joystick signals are modified through an control shaping algorithm such that the operator cannot cause payload swing to occur. The control shaping algorithm could be used in conjunction with the pendulum compensation algorithm, or alone. In brief, the three parts of the payload swing compensation algorithm are

- Remove payload swing using pendulum energy equations.
- Compensate for the ship induced crane motions using inverse kinematics.
- Remove operator induced payload swing using a control shaping technique.

This algorithm could be advantageous in many situations, especially in severe sea conditions. An increased weather operation window are of great interest, due to the high day rates on offshore support vessels. Algorithms like this also make the operations safer.

In the following sections two inverse kinematics algorithms, as well as a control shaping technique, will be described, implemented and tested in Simulink. The first part of the payload swing compensation algorithm, namely to measure the payload swing angle and analyzing pendulum energy equations, has not been studied in this thesis. The reason for this, is because the HMC3568 crane, as well as similar cranes, do not have any equipment for measuring the cable angle. As a result, new hardware would have to be installed in order for the algorithm to work on an actual crane. Instead, the following sections have focused on methods that can implemented without the need of any new hardware.

6.3 Automatic Control

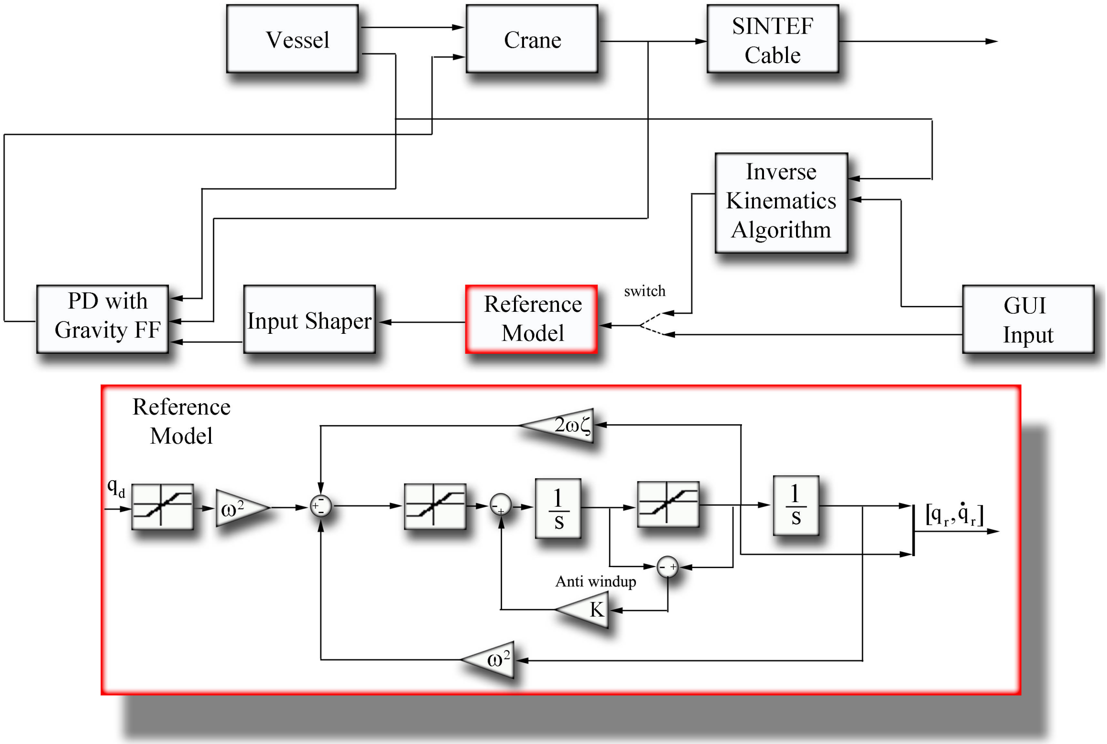


Figure 6.1: Overview of the Simulink model, with the reference model shown in detail.

This section describes how the reference signals are generated. Using the GUI input block, the operator are able to set the desired angles or the desired position. The block brings fourth a graphical user interface (GUI), which can be used to communicate with the Simulink model. The GUI replaces joysticks and panels, which would have been used by the operator to control the crane. Furthermore, if the operator sets a desired position (workspace control), the reference signal is first guided through the inverse kinematics block before it reaches the reference model. For convenience, Table 6.1 shows the four control inputs that have been used in this thesis. Two of them are specified in the configuration space, and two in the workspace. The workspace desired positions are given in $\{N\}$. In addition, the initial configuration for the three first control schemes are $\mathbf{q} = \begin{bmatrix} 0 & 0 & -\frac{\pi}{4} \end{bmatrix}^T$. The fourth control scheme has the same initial configuration as the desired one. It will be described later in this section what the two different modes means.

Control scheme number	Space	Desired	Reference model mode
1	Joint space	$\mathbf{q} = \begin{bmatrix} \frac{\pi}{2} & \frac{\pi}{4} & -\frac{\pi}{2} \end{bmatrix}^T$	Normal
2	Joint space	$\mathbf{q} = \begin{bmatrix} \frac{\pi}{4} & 0 & 0 \end{bmatrix}^T$	Normal
3	Workspace	$\mathbf{X} = \begin{bmatrix} -10 & 20 & 13 \end{bmatrix}^T$	Normal
4	Workspace	$\mathbf{X} = \begin{bmatrix} -10 & 20 & 13 \end{bmatrix}^T$	Position keeping

Table 6.1: Input schemes.

The objective of the reference model is to generate smooth trajectories for the crane to follow. It is important that the bandwidth of the reference model is chosen lower than the bandwidth of the motion control system in order to obtain satisfactory tracking performance. One way to generate smooth trajectories is to use a reference model inspired by a the dynamics of a mass-damper-spring system.

The reference model described in this section is motivated by a mass-dampen-spring model. It can be seen on block form in Figure 6.1, and on transfer function form, it can be written as

$$\frac{q_r}{q_d} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}. \tag{6.1}$$

Here, q_r is the reference joint angle and q_d is the desired joint angle, given from the GUI. ζ_i is the relative damping ratio and ω_{n_i} are the natural frequency. Both the relative dampening ratio and the natural frequency have to be chosen with regards to the desired crane performance. Due to the fact that the reference model is of second order, it is possible to saturate position, velocity and acceleration.

As mentioned, ζ is the relative dampening ratio. By choosing $\zeta = 0$ one obtain an undamped system, i.e. no energy is removed from the system. By choosing $0 < \zeta < 1$ one obtain an under damped system, while choosing $\zeta = 1$ a critical damped system is obtained. By choosing $\zeta > 1$, the system becomes over damped. Furthermore, a reference model with high natural frequency are able to track signals of higher frequency then a model with low natural frequency.

Industrial reference models usually have stored several sets of parameters. Each set of parameters are used in certain modes, like the AHC or AT for the crane, or at specific occasions. In our case, there are needed two pairs of parameters. One pair that is used when the crane is in normal mode, and one pair that is used when the crane counteracts the vessel induced crane motion, which is called position keeping mode. For the normal mode, the parameters that gives the most satisfactory tracking performance are

$$\boldsymbol{\omega}_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \qquad \boldsymbol{\zeta}_1 = \begin{bmatrix} 0.9 \\ 0.9 \\ 0.9 \end{bmatrix}. \tag{6.2}$$

These parameters make the reference model slightly under damped. The reference model also produces very similar q_r to what the real operator input would have produced.

When the crane counteracts the vessel induced crane motion, a reference model that is able track signals of higher frequency are needed. Therefore, the parameters for this mode have been chosen to

$$\boldsymbol{\omega}_2 = \begin{bmatrix} 6 \\ 4 \\ 6 \end{bmatrix}, \qquad \boldsymbol{\zeta}_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}. \tag{6.3}$$

As a result of increasing the natural frequency, the reference model able to track signals of higher frequency. The relative dampening has also been increased. The resulting reference model is a critical damped model, that is able to track signals of high frequency.

According to Wittenmark et al. (2003), a model with integral action combined with an signal that becomes saturated may lead some undesirable effects. Hence, an anti windup feature has been implemented in the reference model. The anti windup feature is marked in Figure 6.2. The model also contains saturation blocks. These make sure that the limitations in Table 3.2 are met.

6.4 PD Controller with Gravity Compensation

The problem of controlling the crane can be formulated as that to determine the torques and forces so as to guarantee that the crane stays on the desired path. In all it essence, the problem is to find $\boldsymbol{\tau}$ in (2.48) in a way that makes the system stable and work according to requirements.

In robot control theory, there are usually two kinds of controllers that are considered, namely joint space controllers and workspace controllers. Both have a closed loop structure in order to exploit the good features provided by the feedback. The resulting controllers are robust and able to suppress disturbance. Workspace controllers have a greater complexity due to its inverse kinematics algorithm in the feedback loop. Its main advantage is that this controller gives the possibility to act directly upon the work space variables. Joint space controllers however, are the way the operators control the crane today. This group of controllers can be further be divided into two control techniques. The

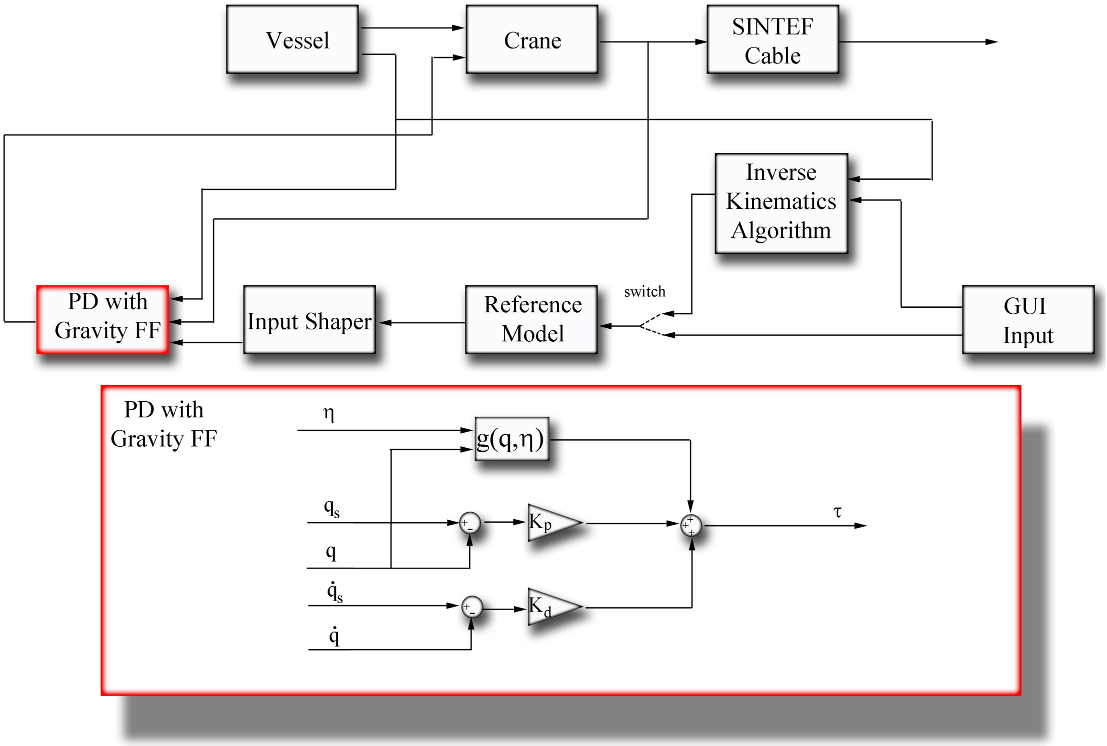


Figure 6.2: Overview of the Simulink model, with the reference model shown in detail.

first technique is called decentralized control. It considers a single manipulator joint independently of the others. The other technique is called centralized control. This control technique takes the dynamic interaction effects between the joints into account.

The controller used in the Simulink model shown in Figure 6.2 is a joint space centralized control technique. It is a PD controller that also feedforward the vessel-crane form of (2.45), which is the gravity term. The PD term is used for fast response, while the gravity term simply compensate for the gravitational influence on each structural element. Together they stabilizes the system, while also keeping the crane on it's desired path. See Sciavicco and Sciavicco (2000) for proof of global asymptotic stability of this controller. Stability is ensured for any choice of K_d and K_p, as long as these matrices are positive definite.

6.5 Inverse Kinematics

As mentioned, the forward kinematics problem is to calculate the end effector position given the joint angles. The end effector position given in $\{0\}$, is simply found by inserting the joint variables into (4.16). On the other hand, the inverse kinematic problem is much more complex. The problem is to determine the joint angles given the end effec-

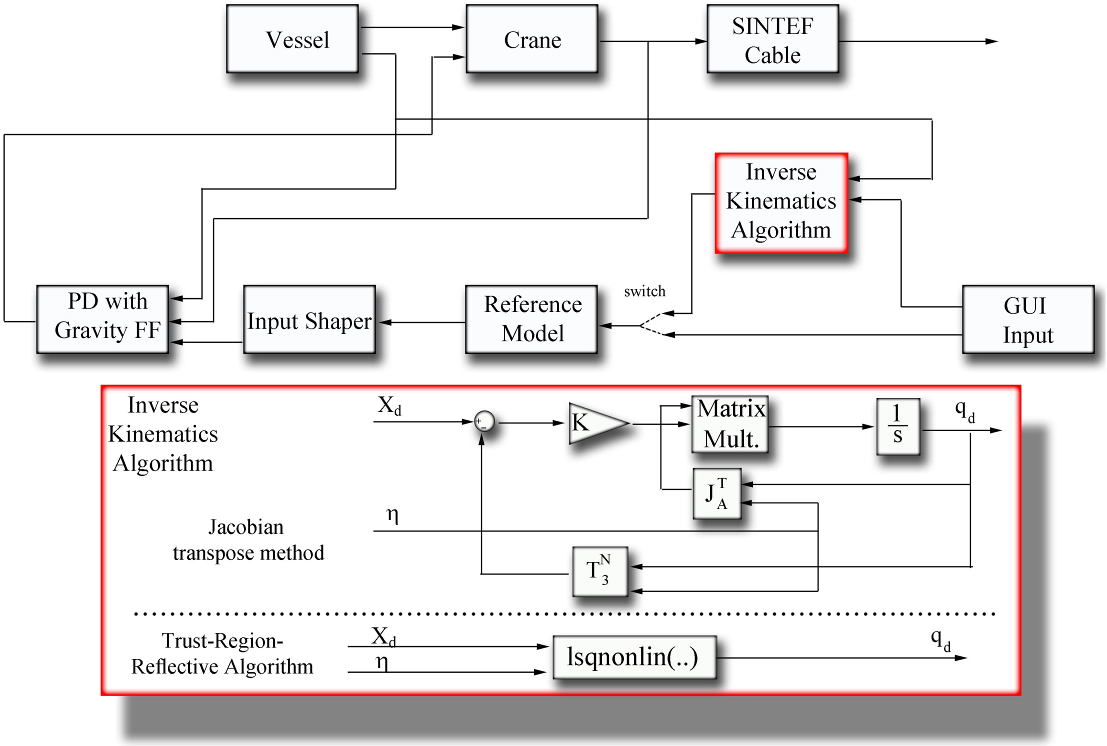


Figure 6.3: Overview of the Simulink model, with the inverse kinematics model shown in detail.

tor position. A solution to the inverse kinematic method could be advantageous in many situations. For instance, the algorithm could be used to keep a fixed crane tip position with regards to $\{N\}$. This means that the the algorithm could compensate for the vessel orientation and position. The method could also be used by the crane operator to control the crane tip in the workspace.

In this section, two inverse control methods will be described and tested. Both methods are simulated with the vessel-crane model, with and without vessel movement. For simulations without waves, control scheme number 3 has been used, while control scheme number 4 has been used when the vessel was exposed to waves of sea state 2, 4 and 6. The control schemes were sown in Table 6.1, and the sea states were shown in Table 4.2.

The inverse kinematics algorithms are able to calculate configurations that are needed to keep the crane tip fixed in $\{N\}$. However, during simulations it was seen that the slow jib and luff rate made it hard for the crane to keep the desired position. Therefore, both algorithms presented in this section calculates $\mathbf{q}$ based on the desired x and y position. The desired z position has been removed from the equations. In fact, it would be impractical to use the crane to compensate for the z position, when the winch, which has higher bandwidth, are able to do the job. If one want to keep a fixed x, y and z crane tip position, one could use the already existing ACH function together with workspace control method.

6.5.1 The General Inverse Kinematics Problem

The inverse kinematics problem is a much more complex problem to solve than the forward kinematics problem for the following reasons (Sciavicco and Sciavicco, 2000)

- The equations to solve are in most cases highly nonlinear, and thus it is not always be possible to find a closed form solution.
- Multiple solutions may exist.
- Infinite solutions or no admissible solutions may exist, in view of the manipulator kinematic structure.

Furthermore, the inverse problem can be stated as follows. Given the 4×4 homogeneous transformation matrix

$$\mathbf{H}_d = \begin{bmatrix} \mathbf{R}_d & \mathbf{p}_d \\ \mathbf{0} & 1 \end{bmatrix}, \tag{6.4}$$

with $\mathbf{R}_d$ as the 3×3 desired orientation matrix, and $\mathbf{p}_d$ as the 3×1 desired position vector, find joint variables q_1, q_2, q_3 so that

$$\mathbf{T}_B^N \mathbf{T}_0^B \mathbf{T}_3^0(q_1, q_2, q_3) = \mathbf{T}_3^N(q_1, q_2, q_3) = \mathbf{H}_d. \tag{6.5}$$

Here, $\mathbf{H}_d$ is expressed with regards to $\{N\}$. Furthermore, the inverse kinematics problem can be decomposed into two subproblems, namely the inverse position kinematics, and the inverse orientation kinematics. When the orientation is of no interest, as for the AHC crane, the problem is reduced to an inverse position problem, which can be written as

$$\mathbf{p}_3^N(q_1, q_2, q_3) = \mathbf{p}_d. \tag{6.6}$$

As a result, the problem is reduced to finding three unknowns, using three equations.

There exist two main categories of solution methods to the inverse kinematics problem. They can be split into the two following categories

Closed Form The forward kinematics may be rewritten in a manner that leads to a set of structured non-linear equations that may be solved explicitly for the joint variables. The closed form solution can be further divided into two approaches

1. **Geometric** Reduce the larger problem to a series of plane geometry problems.
2. **Algebraic** Rearrange the equations to find a trigonometric solution.

Numerical A numerical algorithm is applied and explicitly produces all feasible solutions.

For the vessel-crane inverse kinematics problem, the closed form solution approach would be unsuitable due to the overall complexity. For this reason the main focus has been on developing numerical algorithms.

6.5.2 Trust-Region-Reflective Algorithm

The trust-region-reflective algorithm is a subspace trust-region method, and it the standard solution method for the function *lsqnonlin* in Matlab. This method is based on the interior-reflective Newton method, and is described in (Coleman and Li, 1996) and (Coleman and Li, 1994). The function solves a system of non-linear equations, with or without constraints.

For the *lsqnonlin* function to solve the non-linear equations (6.6), they are rearranged to

$$\mathbf{p}_3^N(q_1, q_2, q_3) - \mathbf{p}_d = \mathbf{0}. \tag{6.7}$$

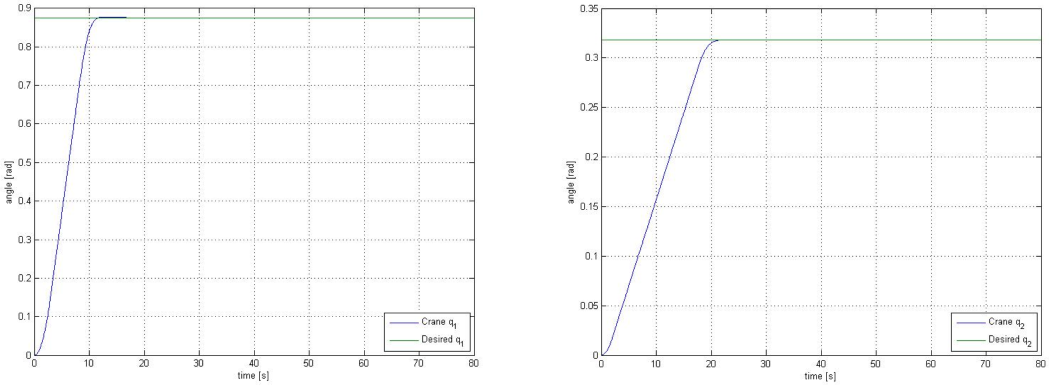
As mentioned, only the x and y part of (6.7) will be used to generate $\mathbf{q}_d$. In addition, there have been added constraints that makes sure that the configurations are within the joint space. Another nice feature, is that a cost function can be specified. As a result, the method can be optimized with regards to lift capacity (derating), shortest path etc.

Figure 6.4 present simulation results when there were no waves affecting the vessel. From the subfigures it can be seen that the algorithm is capable of getting the crane to the desired position. This means that the position error converge towards zero, which can be seen on the position error subfigure.

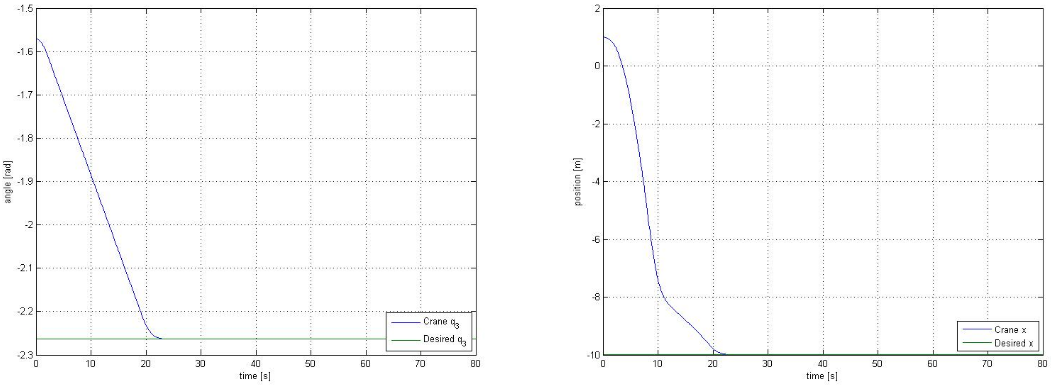
The resulting crane response when the vessel is exposed to waves of sea state 2 is shown in Figure 6.5. This sea state is characterized as smooth. From the joint variable subfigures it can be seen that the q_1 and q_3 are able to track the q_{1d} and q_{3d}. In contrast, q_2 has a small phase lag compared to q_{2d}. However, the crane is capable of keeping the position error within ± 6 cm. It should be noted that the jitter that is seen on the sub-figures, is filtered by the reference model before it reaches the controller. The jitter is produced by the *lsqnonlin* function.

Figure 6.6 presents the crane response for sea state 4, which is characterized as a moderate sea state. The sub-figures show similar results to what were seen in sea state 2. However, the position error has increased to ± 15 cm.

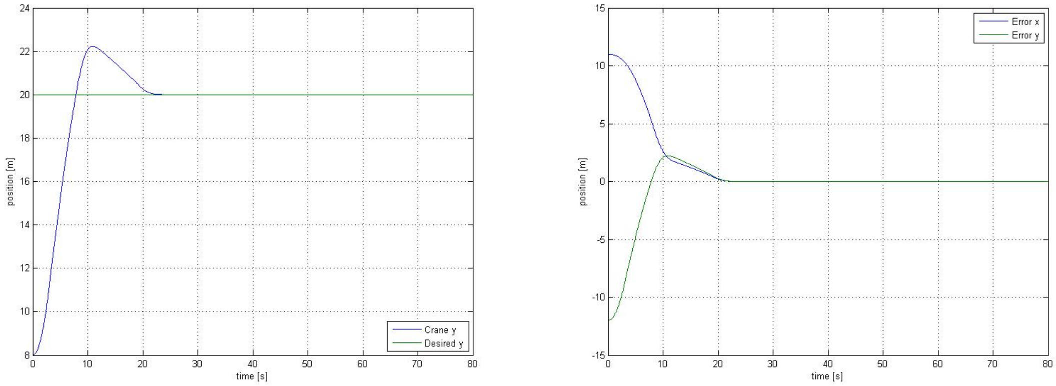
Simulation results for sea state 6, are shown in Figure 6.7. Still, the joint variable q_1 is able to track q_{2d}. In contrast, the two other joint variables are not able to track the desired joint variable. As a result, the position error oscillate around the desired position with approximate 2m in amplitude.



(a) Joint variable q_1 (b) Joint variable q_2

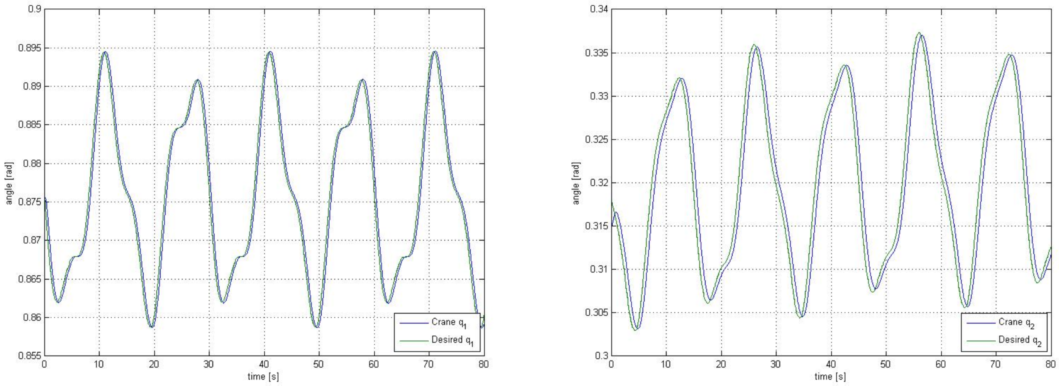


(c) Joint variable q_3 (d) Position x

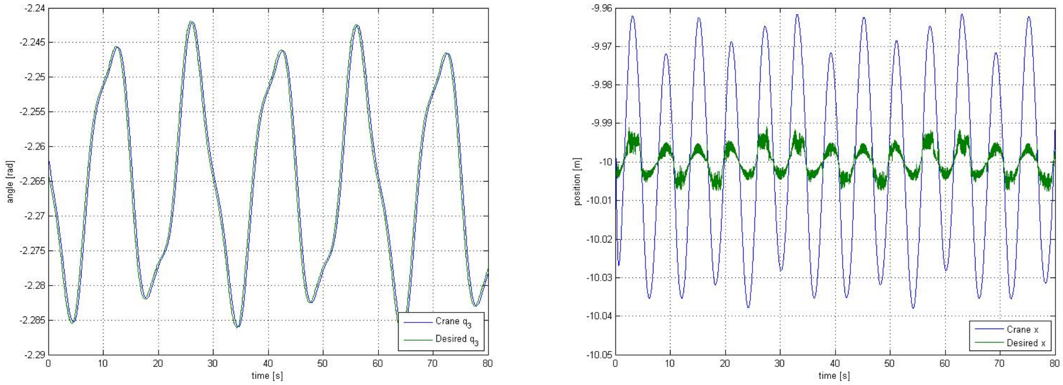


(e) Position y (f) Position error

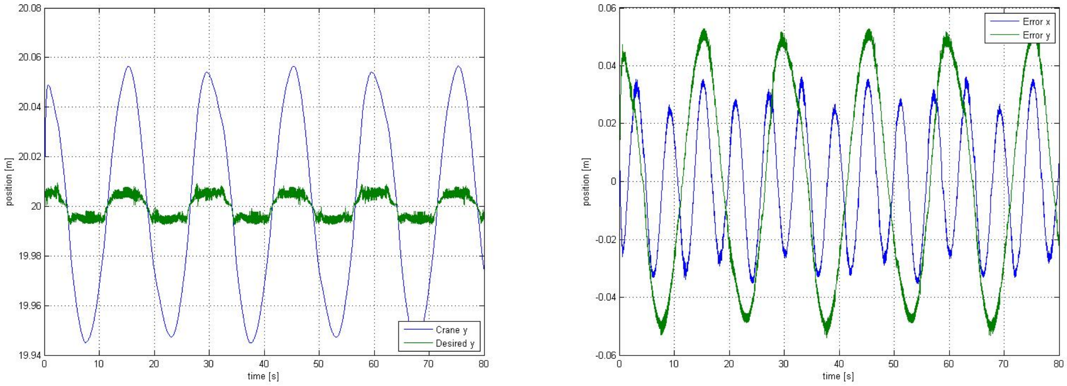
Figure 6.4: Without waves.



(a) Joint variable q_1 (b) Joint variable q_2

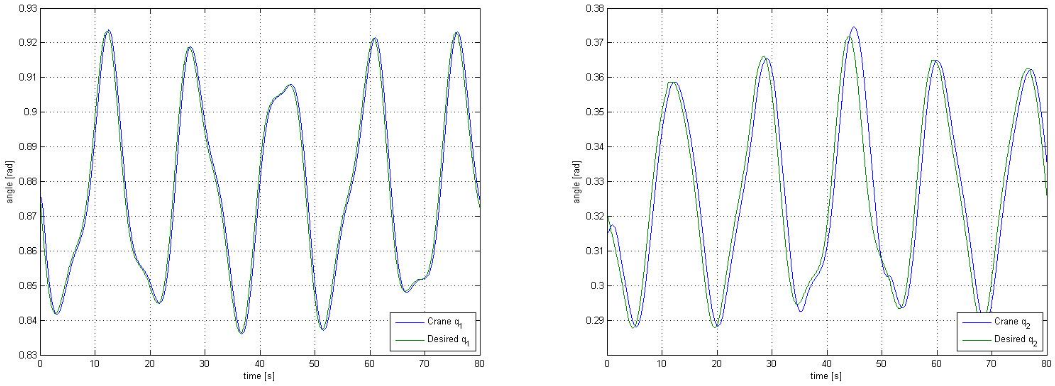


(c) Joint variable q_3 (d) Position x

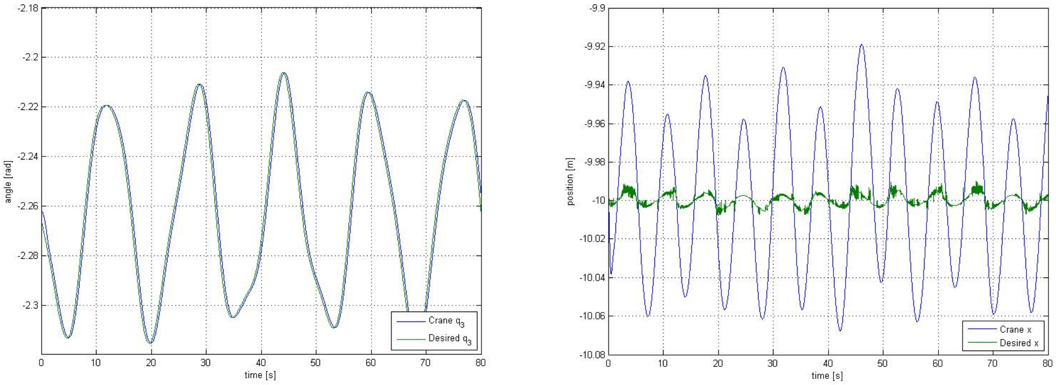


(e) Position y (f) Position error

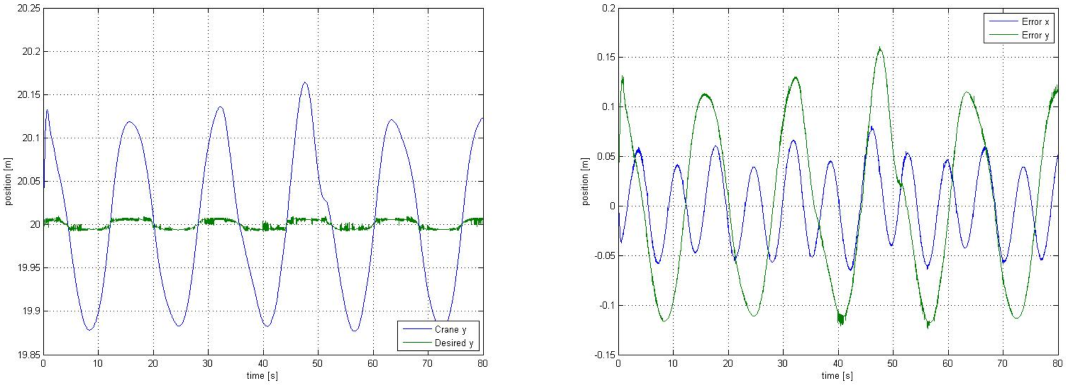
Figure 6.5: Sea state 2.



(a) Joint variable q_1 (b) Joint variable q_2

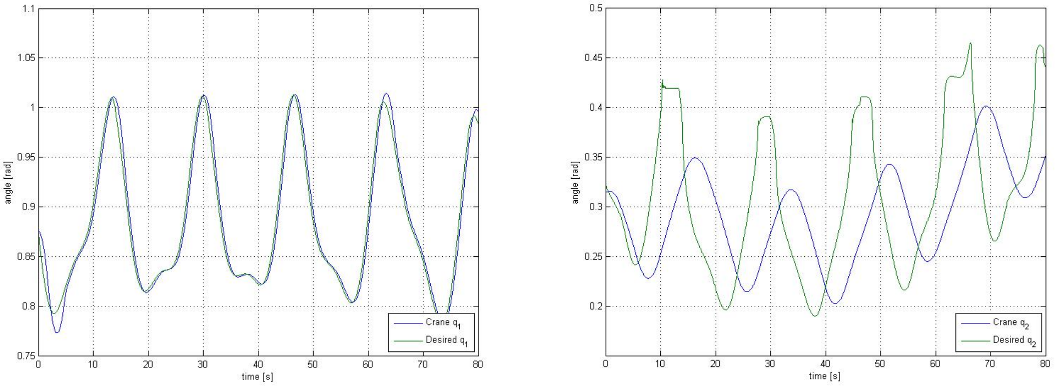


(c) Joint variable q_3 (d) Position x

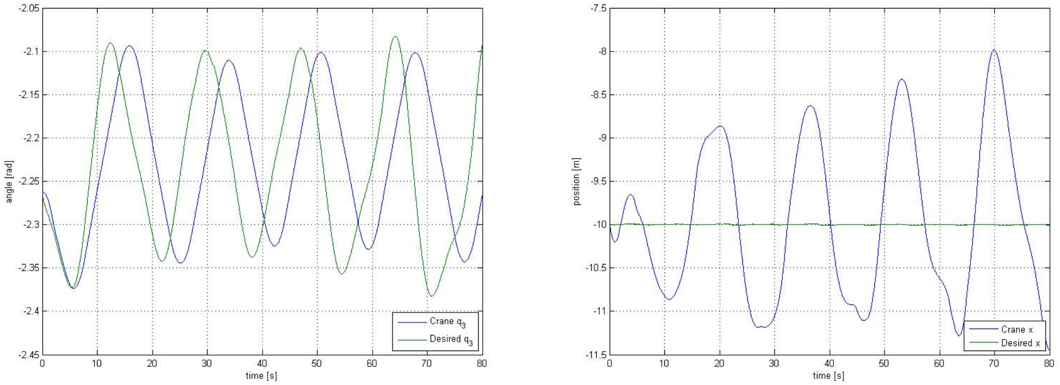


(e) Position y (f) Position error

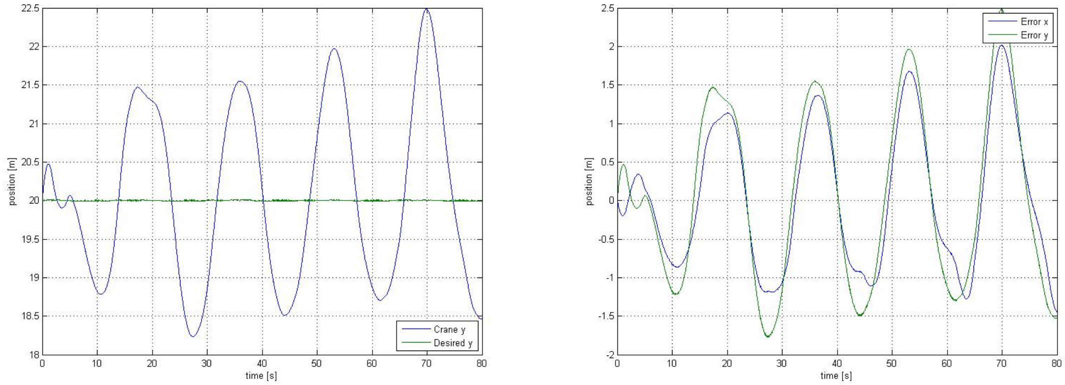
Figure 6.6: Sea state 4.



(a) Joint variable q_1 (b) Joint variable q_2



(c) Joint variable q_3 (d) Position x



(e) Position y (f) Position error

Figure 6.7: Sea state 6.

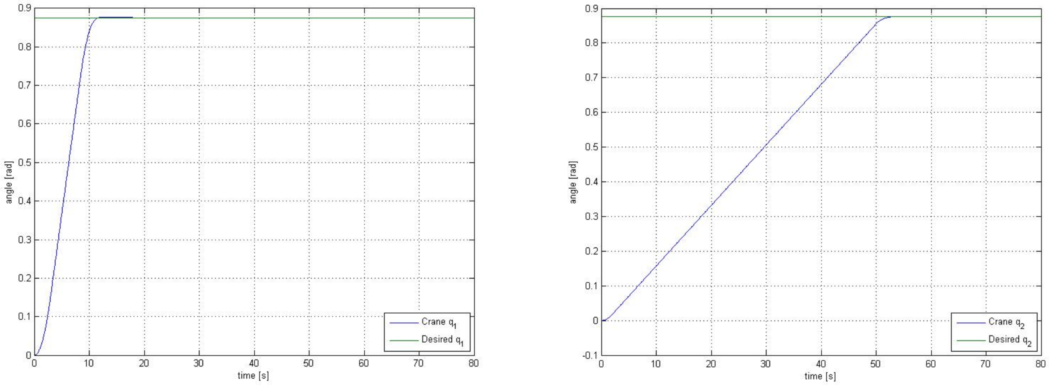
6.5.3 The Jacobian Transpose Method

The Jacobian transpose method is based on the computationally slower algorithm, the inverse Jacobian method. The only difference between these two methods is that the Jacobian transpose method use, as the name indicate, the transposed Jacobian instead of the inverse Jacobian. Compared to the Jacobian inverse, the Jacobian transpose method has cheaper step evaluation cost, as well as it avoids singularity problems. Figure 6.3 shows the block scheme of the Jacobian transpose method. Proof of convergence can be found in (Sciavicco and Sciavicco, 2000).

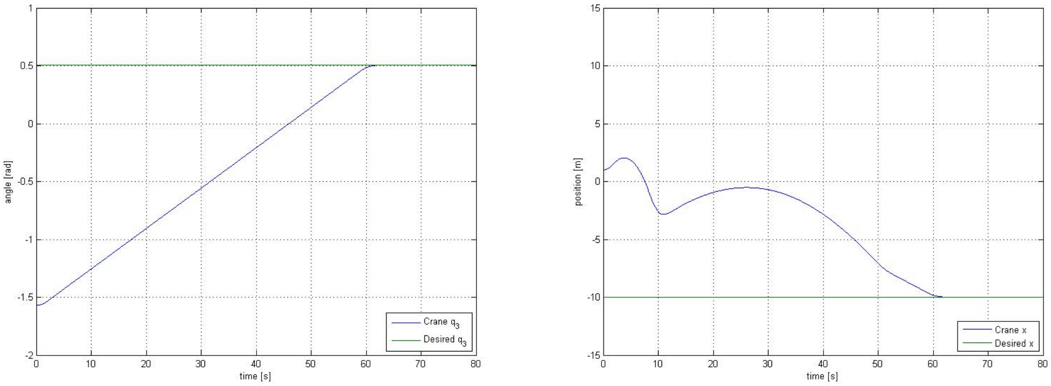
Figure 6.8 presents simulation results when there are no environmental disturbance. It can be seen that the method makes the position error converge towards 0. However, this method generate configurations that are not within the configuration space. In contrast to the *lsqnonlin* function, the Jacobian transpose method is not able to ensure that the calculated q_d is a feasible configuration. For this reason, the position saturation in the reference model had to be removed. The crane reaches its desired position after 60 seconds.

Figure 6.9 presents the simulation results for sea state 2. It can be seen that the Jacobian method generate q_{1d} and q_{3d} that the crane are able to track. Joint variable q_2 has problems tracking q_{2d}, which makes the position error oscillate at about ± 0.6 m.

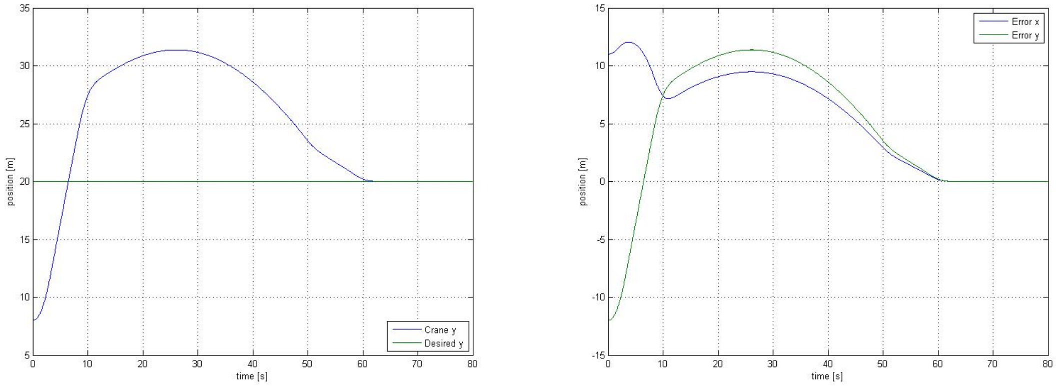
Figure 6.10 and Figure 6.11 present simulation results for sea state 4 and 6, respectively. In sea state 4, q_3 are the only joint variable that is able to track the desired value. Joint variable 1 and 2 are unable to follow the desired value. In sea state 6, all three joint variables are unable to track the desired value. This results in a position error of ± 5m in sea state 4 and ± 12m in sea state 6.



(a) Joint variable q_1 (b) Joint variable q_2

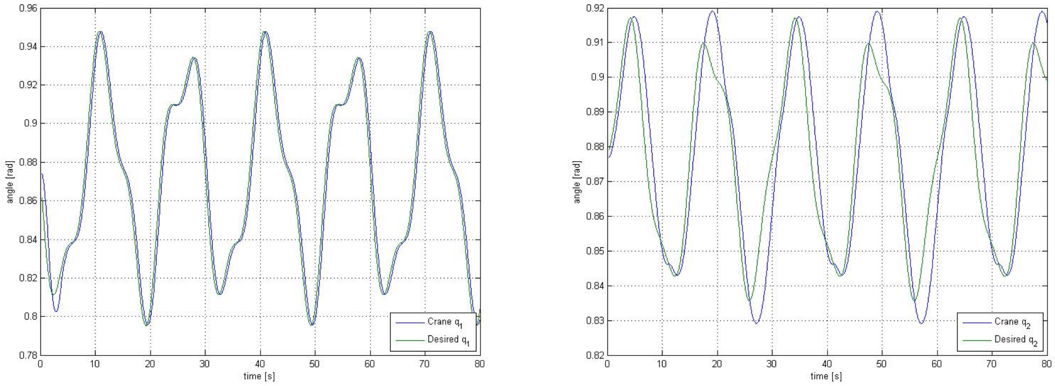


(c) Joint variable q_3 (d) Position x



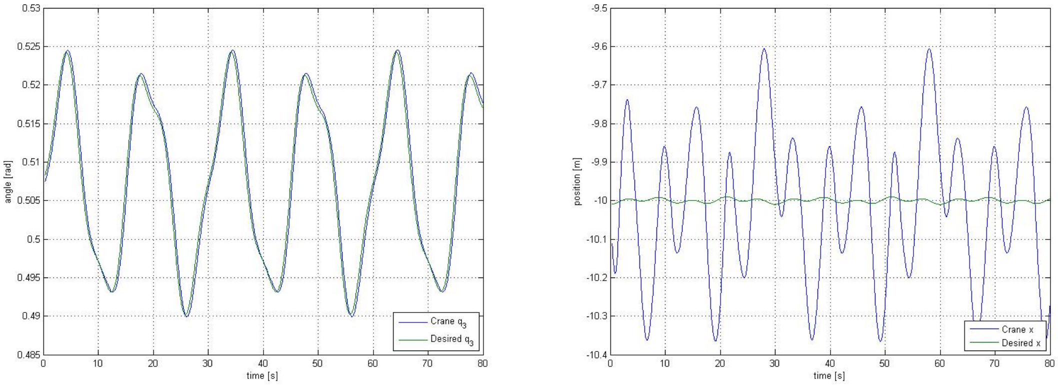
(e) Position y (f) Position error

Figure 6.8: Without waves.



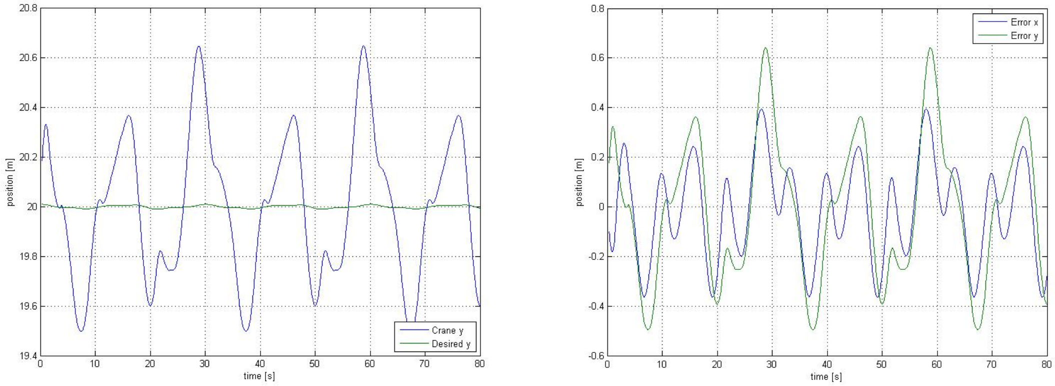
(a) Joint variable q_1

(b) Joint variable q_2



(c) Joint variable q_3

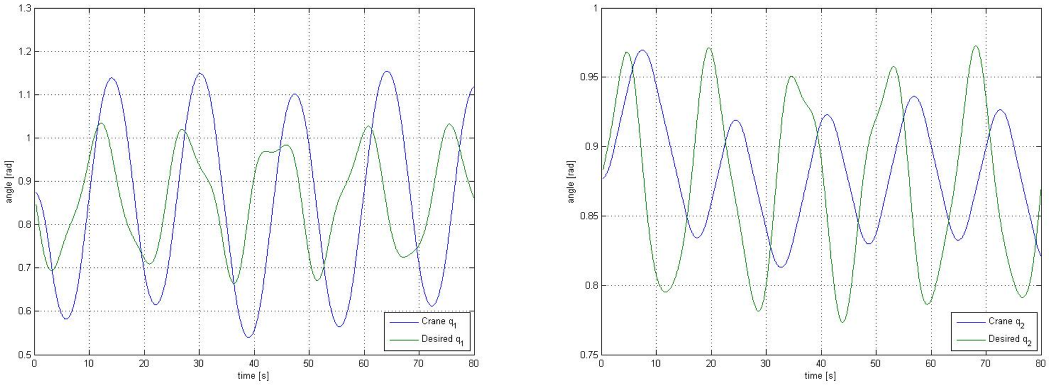
(d) Position x



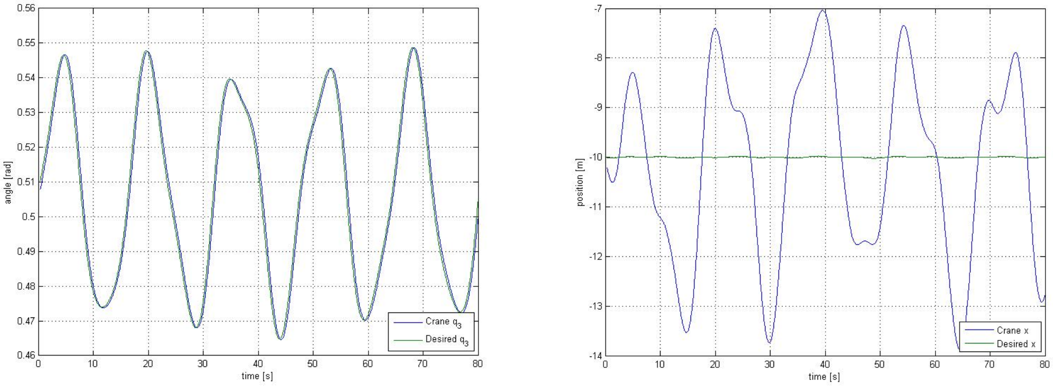
(e) Position y

(f) Position error

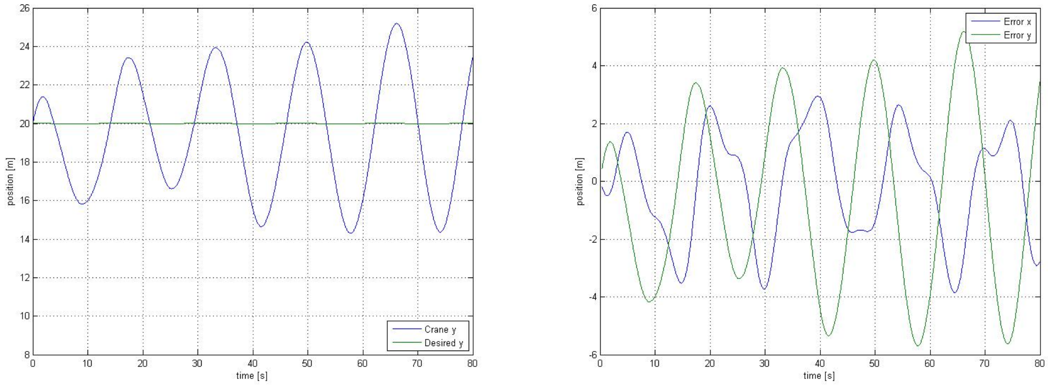
Figure 6.9: Sea state 2.



(a) Joint variable q_1 (b) Joint variable q_2

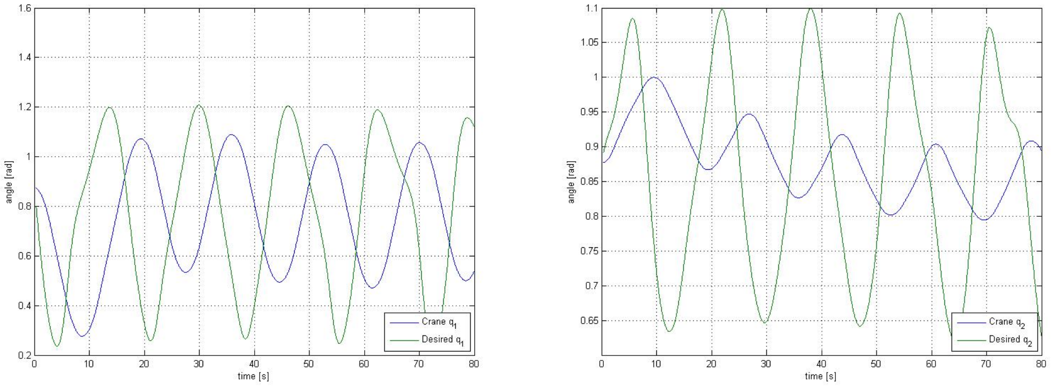


(c) Joint variable q_3 (d) Position x

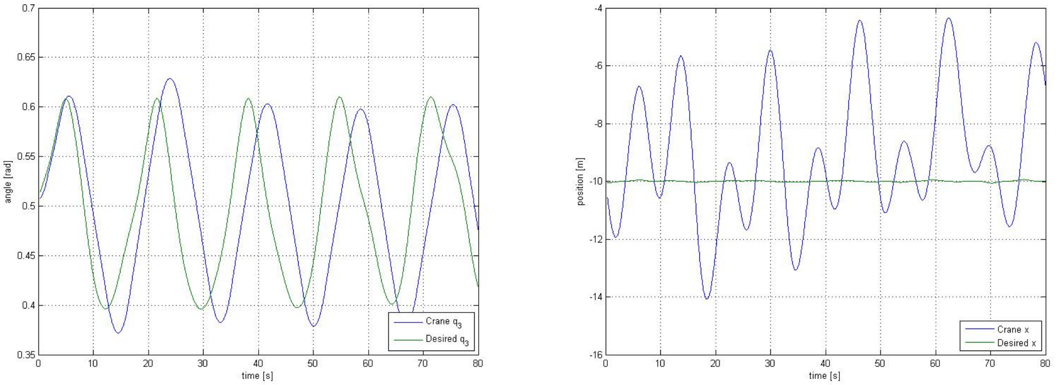


(e) Position y (f) Position error

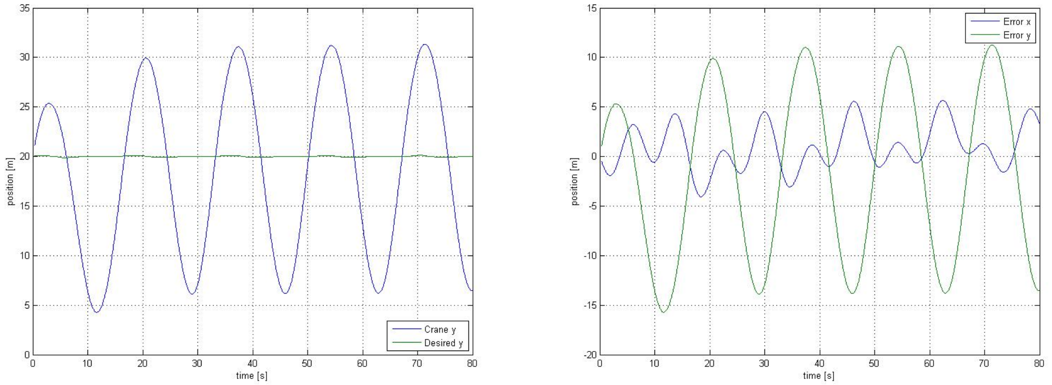
Figure 6.10: Sea state 4.



(a) Joint variable q_1 (b) Joint variable q_2



(c) Joint variable q_3 (d) Position x



(e) Position y (f) Position error

Figure 6.11: Sea state 6.

6.5.4 Discussion

Section 6.5 has presented results for both the transposed Jacobian method and the *lsqnonlin* method. The transposed Jacobian method generated an infeasible solution, which was not dealt with. Instead, the position saturation in the reference model was removed. This makes the Jacobian method not very interesting for crane suppliers because they have to be sure that feasible solutions are generated.

The *lsqnonlin* method proved to be way better than the transposed Jacobian. As well as the method produced desired joint values that the crane was able to track, the method also generated feasible configurations. The method successfully kept the crane tip position within ± 15cm of the desired values in moderate seas. The transposed Jacobian on the other hand, showed an position error of approximately 5m. In sea state 6 however, the crane became unable to follow the desired values generated by *lsqnonlin*. This inability to keep a fixed position in rough sea is most likely due to saturations in the crane.

All in all, the *lsqnonlin* proved to be superior to the transposed Jacobian. It showed better performance, as well as well as it generated feasible solutions.

6.6 Input Shaping

In this section, a control shaping technique called input shaping will be discussed. First, there will be given an introduction to the vibration control problem. Second, input shaping theory will be given, before three different shapers are formulated and implemented in the Simulink model. Last, there will be a discussion based on simulation results.

6.6.1 Introduction

Input Shaping is a feedforward control technique for reducing vibrations of oscillatory systems controlled by computers. The method works by creating command signals that cancels the vibration caused by the first part of the command signal. If the command signals in the shaper are chosen correctly, the system will respond without vibration. As an example, a ZV shaper response for an undamped system is illustrated in Figure 6.13. The figure shows how a second impulse can cancel the vibration caused by the first impulse.

For the computer controlled system to be able to utilize vibration control, the vibrations have to be of a finite frequency and amplitude. In addition, the vibration frequencies have to be within the control systems bandwidth. Input shaping is most often implemented on computers that control multi-body, or single-body, flexible systems. These systems may have resonance frequency that is within the control system's bandwidth, hence making it possible to reduce the vibrations.

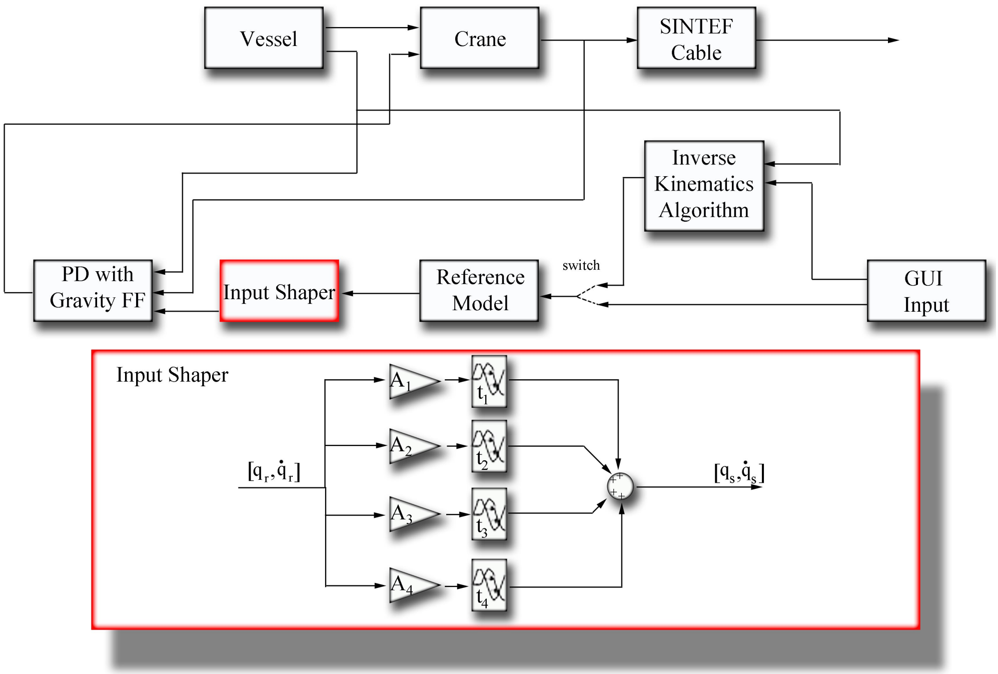


Figure 6.12: Overview of the Simulink model, with the input shaping block shown in detail.

Modern AHC cranes are controlled by a human operator who uses joysticks to alter joint angles. If the operator alter a joint for a finite time period, the crane tip will move a finite distance before it comes to rest. The payload however, might not come to rest. It may start to oscillate due to the change of distance. An experienced crane operator may be able to reduce these oscillations by alter the joint angles multiple times at proper instances. This process takes time, and even experienced operators may struggle to put the payload to rest.

For convenience, an overview of the input shaping implementation is shown in Figure 6.12. The reference signal is shaped by a number of number of gains and signal delays. These parameters are chosen in a way that reduces the residual vibration. For the reader to be fully aware of the shaping control problem, a simulation without any vibration control is shown in Figure 6.14. The figure shows that the difference between the crane tip position and the payload position oscillate at approximately 1m in amplitude in the x and y direction. This simulation, and all other simulations in this section have been done by rotating the king $45°$, i.e. control scheme number 2 in Table 6.1. The input shapers main objective is to alter the cranes configuration without inducing the oscillations seen in Figure 6.14.

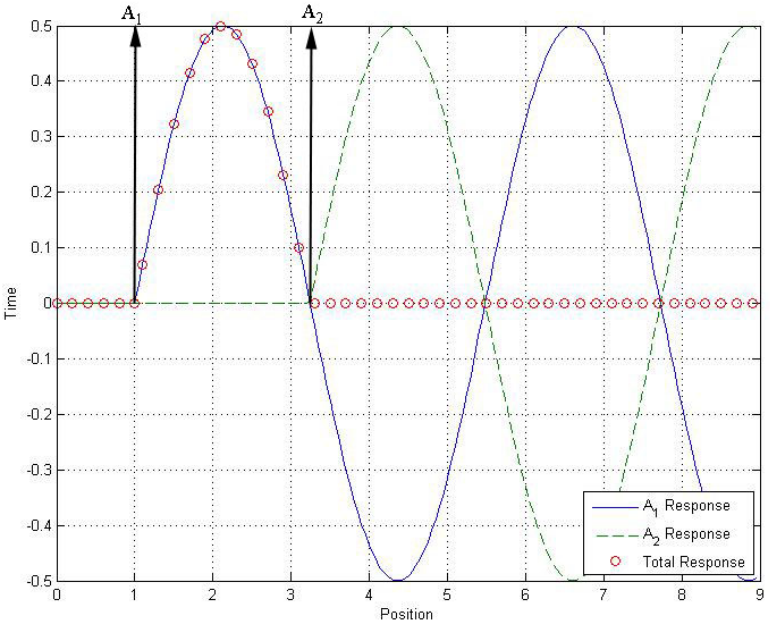


Figure 6.13: Two impulse response.

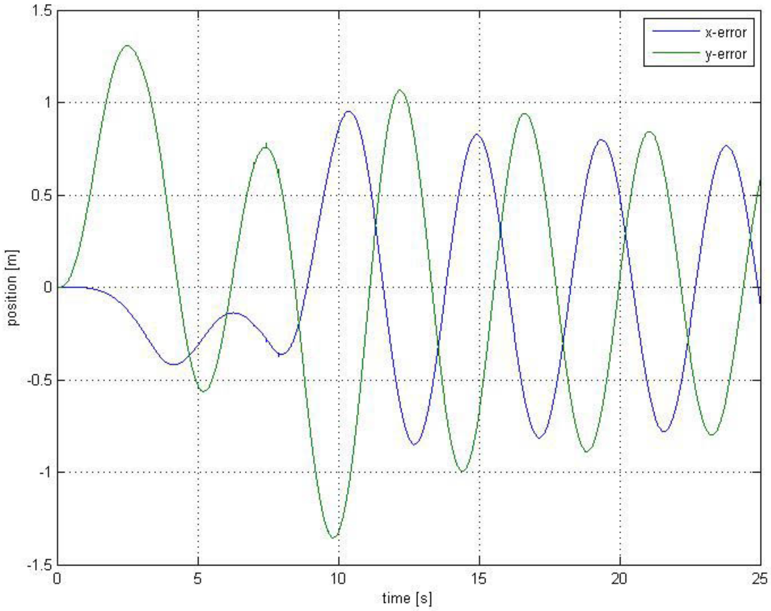


Figure 6.14: Payload oscillations without any input shaper.

6.6.2 Preliminaries on Input Shaping Theory

It may be possible to decompose a high order system to be a combination of a series of second order subsystems and design the corresponding shaper for each subsystem. It is feasible to discuss the system performance by assuming the closed-loop transfer function of the model as (Yuan and Chang, 2006)

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}. \tag{6.8}$$

In (6.8), ω_n represents the natural frequency, and ζ the dampening ratio of the system. The unit impulse response of (6.8) is given as

$$y(t) = \frac{\omega_n}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n(t-t_0)} \sin\left[(\omega_n\sqrt{1-\zeta^2})(t-t_0)\right] u(t-t_0), \tag{6.9}$$

where t_o is the impulse time instance and u(·) is the unit step function (Chang et al., 2006). The unit step function is split into a series of impulse inputs $A_i\delta(t-t_i)$. Here, A_i and t_i are the amplitudes and the time locations of the impulse inputs. The total response is

$$y(t) = \sum_{i=1}^{m} y_i(t), \tag{6.10}$$

where m is the number of impulses. The unit impulse response $y_i(t)$ is given as

$$y_i(t) = \frac{A_i\omega_n}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n(t-t_i)} \sin\left[(\omega_n\sqrt{1-\zeta^2})(t-t_i)\right] \delta(t-t_i), \tag{6.11}$$

The total response for the vibration system at settling time $t = t_N$ can be written as

$$y(t_N) = \sum_{i=1}^{m} \frac{A_i\omega_n}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t_N} e^{\zeta\omega_n t_i} \sin(\omega_n\sqrt{1-\zeta^2}(t_N - t_i)), \tag{6.12}$$

or alternatively,

$$y(t_N) = \sum_{i=1}^{m} B_i \sin(\omega_d t_N - \omega_d t_i), \tag{6.13}$$

where

$$\omega_d = \omega_n\sqrt{1-\zeta^2}, \tag{6.14}$$

$$B_i = \frac{A_i \omega_n}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t_N} e^{\zeta\omega_n t_i}. \tag{6.15}$$

Using the trigonometric identity (Gieck and Gieck, 1996)

$$K_1 \sin(\alpha t + \phi_1) + \ldots + K_n \sin(\alpha t + \phi_n) = A_{amp} \sin(\alpha t + \psi), \tag{6.16}$$

where

$$A_{amp} = \sqrt{\left(\sum_{i=1}^{n} K_i \cos(\phi_i)\right)^2 + \left(\sum_{i=1}^{n} K_i \sin(\phi_i)\right)^2}, \tag{6.17}$$

$$\psi = \tan^{-1} \frac{\sum_{i=1}^{n} K_i \cos(\phi_i)}{\sum_{i=1}^{n} K_i \sin(\phi_i)}, \tag{6.18}$$

makes it possible to write the vibration amplitude as

$$A_{\sum} = e^{-\zeta\omega_n t_N} \frac{\omega_n}{\sqrt{1-\zeta^2}} \sqrt{(\sum_{i=1}^{m} A_i e^{\zeta\omega_n t_i} \cos(\omega_d t_i))^2 + (\sum_{i=1}^{m} A_i e^{\zeta\omega_n t_i} \sin(\omega_d t_i))^2}. \tag{6.19}$$

To form a nondimensional vibration amplitude, (6.19) is divided by the amplitude of residual vibration from a single impulse of unity magnitude. The resulting expression gives the vibration in percentage of the unshaped vibration. The amplitude of residual vibration from a single unity-magnitude impulse applied at time zero is (Vaughan et al., 2008)

$$A_{single} = \frac{\omega_n}{\sqrt{1-\zeta^2}}. \tag{6.20}$$

Now, dividing (6.19) by (6.20) yields

$$V(\omega, \zeta, t_N) = e^{-\zeta\omega_n t_N} \sqrt{C(\omega,\zeta)^2 + S(\omega,\zeta)^2}, \tag{6.21}$$

where

$$C(\omega,\zeta) = \sum_{i=1}^{m} A_i e^{\zeta\omega_n t_i} \cos(\omega_d t_i), \tag{6.22}$$

$$S(\omega,\zeta) = \sum_{i=1}^{m} A_i e^{\zeta\omega_n t_i} \sin(\omega_d t_i). \tag{6.23}$$

Here, the two following trigonometric relations has been used:

$$\sin(-\phi) = -\sin(\phi), \tag{6.24}$$
$$\cos(-\phi) = \cos(\phi). \tag{6.25}$$

By setting (6.21) to zero, enables us to find the amplitudes and time locations that would result in zero vibration. But first, a few restrictions have to be placed on the impulse amplitude. The first constraint is that the sum of all amplitudes have to sum to one

$$\sum_{i=1}^{m} A_i = 1. \tag{6.26}$$

This restriction is placed in order to avoid zero-valued impulses, and to obtain a normalized result. Next, each impulse amplitude has to be restricted to take on only positive and finite values

$$A_i > 0 \quad i = 1, \ldots, m. \tag{6.27}$$

It is also possible to add another constraint that makes the shaper more robust to errors in the natural frequencies of the system. By setting the derivative of $V(\omega, \zeta, t_N)$ to zero makes the level of robustness higher, but it also makes the system response slower. The constraint is

$$\frac{d^i}{d\omega^i} V(\omega, \zeta, t_N) = 0, \tag{6.28}$$

where i is the level of robustness needed. The last constraint is the time optimality constraint which minimizes the response delay

$$\min t_m. \tag{6.29}$$

The problem is to find the unknown amplitudes and time locations that makes Equation (6.21) zero, while also satisfying the constraints.

6.6.3 Zero Vibration Shaper

The first shaper that will be described is the ZV Shaper. The ZV shaper constructs a second impulse that cancel the vibration induced by the first impulse. This means that the ZV Shaper is a two impulse shaper, which makes $m = 2$ in (6.22) and (6.23). For simplicity, and for avoiding response delay, t_1 is set to zero. The problem now consist of finding the three unknowns t_2, A_1 and A_2. Setting (6.21) to zero, and also ensuring that restriction (6.26), (6.27) and (6.29) are satisfied, results in three equations

$$A_1 + A_2 e^{\zeta \omega t_2} \cos(\omega_d t_2) = 0, \tag{6.30}$$

$$A_2 e^{\zeta \omega t_2} \sin(\omega_d t_2) = 0, \tag{6.31}$$

$$A_1 + A_2 = 1. \tag{6.32}$$

Equation (6.31) is satisfied when the sine term equals zero. This occurs at

$$\omega_d t_2 = n\pi \Rightarrow t_2 = \frac{n\pi}{\omega_d} \quad n = 1, 2, \ldots \tag{6.33}$$

In order to cancel the vibration in the shortest amount of time, hence satisfying (6.29), t_2 is chosen to

$$t_2 = \frac{\pi}{\omega_d}. \tag{6.34}$$

Now, substituting (6.34) and (6.32) into (6.30) yields

$$A_1 - (1 - A_1) e^{\zeta \omega \frac{\pi}{\omega_d}} = 0. \tag{6.35}$$

Rearranging (6.35) and inserting (6.14) gives

$$A_1 = \frac{e^{\frac{\zeta\pi}{\sqrt{1-\zeta^2}}}}{1 + e^{\frac{\zeta\pi}{\sqrt{1-\zeta^2}}}}. \tag{6.36}$$

All in all, the two impulses and time instances that leads to zero vibration are

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{1+K} & \frac{K}{1+K} \\ 0 & \frac{\pi}{\omega_d} \end{bmatrix}, \tag{6.37}$$

where

$$K = e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}}. \tag{6.38}$$

In our case, the natural frequency for the vibration system can be expressed as

$$\omega_n = \sqrt{\frac{g}{l}}, \tag{6.39}$$

where l is the length of the cable and g is the gravitational constant. The dampening ratio is set to zero, due to the fact that the density of air is low. Now, the two impulses and time instances are convolved with the initial command, thus creating the shaped command. The procedure is illustrated in Figure 6.15.

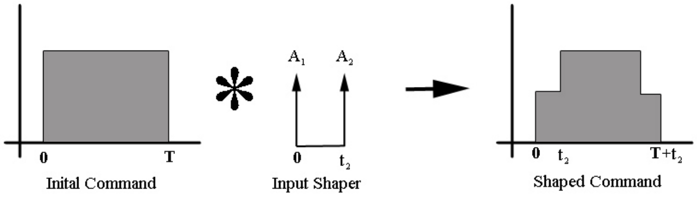


Figure 6.15: Continuous ZV shaper.

The input shaper, together with the rest of the system, are shown in Figure 6.12. Simulations of the input shaper are shown in Figures 6.16 and 6.17. The results show that the shaper is able to reduce almost all the payload oscillation, in both the x and y direction. The shaper introduces a 1.5 seconds delay compared to the unshaped command. However, the shaper would probably beat the human operator with regards to time used, and residual vibration.

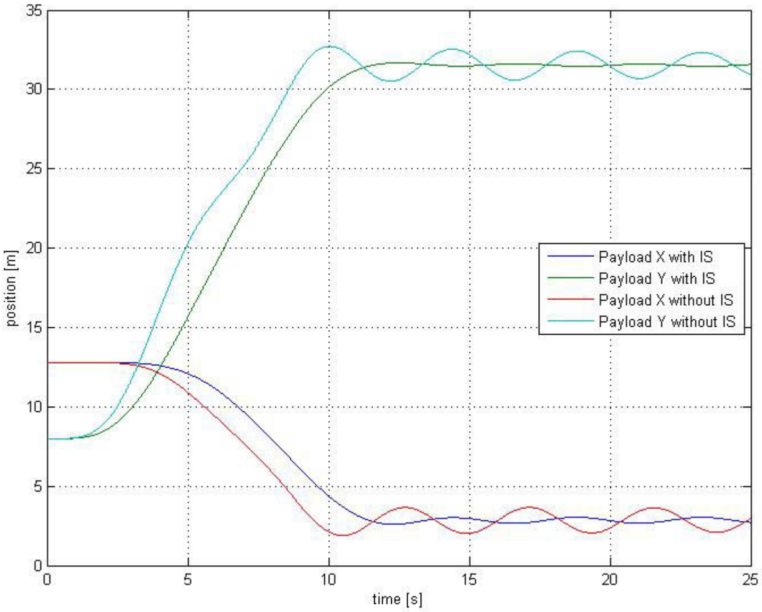


Figure 6.16: Payload position with, and without the continuous ZV shaper.

Figure 6.17 shows the difference between the crane tip position and the payload position. From Figure 6.14 it can be seen that the oscillations without an input shaper are approximately ±1m, whereas ± 0.2m for the ZV shaping case.

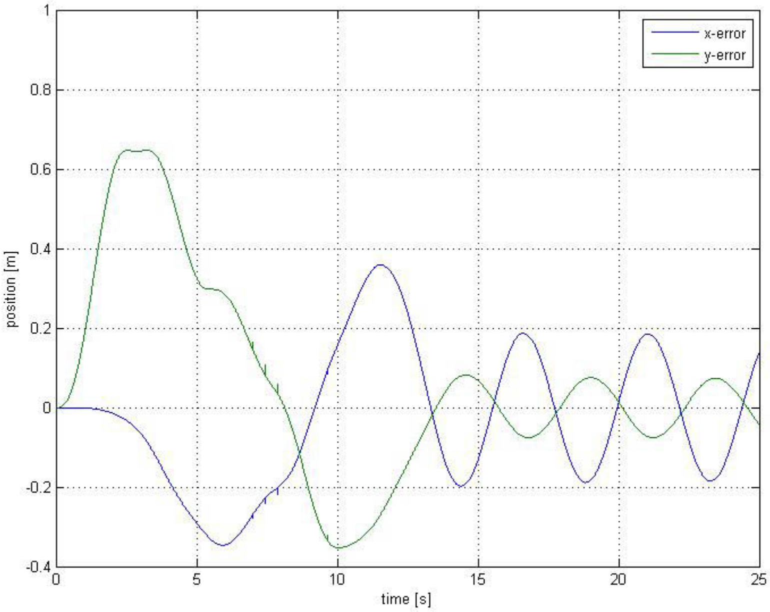


Figure 6.17: Difference between the crane tip and the payload position, using the continuous ZV shaper.

The ZV shaper is in theory able to suppress all vibration if the dampening and the natural frequency are chosen correct. But, unmodeled dynamics will affect the systems behavior. It is therefore necessary to discuss the robustness of the shaper. The robustness of an input shaper can be expressed as the level of the residual vibration (6.21), with respect to variation of the system parameters.

Figure 6.18 shows the three dimensional sensitivity curve for the ZV Shaper. This figure was produced by designing the input shaper with A_i and t_i according to (6.37) and (6.39). Then, a vector of natural frequencies $\boldsymbol{\omega}_{actual}$ and dampening ratios $\boldsymbol{\psi}_{actual}$ were inserted into (6.21) together with A_i and t_i. The actual dampening ratios and natural frequencies were varied according to

$$\omega_{actual} = \sqrt{\frac{g}{l}} - 0.5\sqrt{\frac{g}{l}}, \ldots, \sqrt{\frac{g}{l}} + 0.5\sqrt{\frac{g}{l}}, \tag{6.40}$$

$$\psi_{actual} = 0, \ldots, 0.2. \tag{6.41}$$

The result was a $\mathbf{V}(\omega, \zeta, t_N)$ matrix, where each element correspond to a natural frequency in (6.40), and a dampening ratio in (6.41). It can be seen in Figure 6.18 that even small changes in the natural frequency makes the residual vibration increase rapidly. This makes the ZV Shaper the first of choice when one has perfect knowledge of the system.

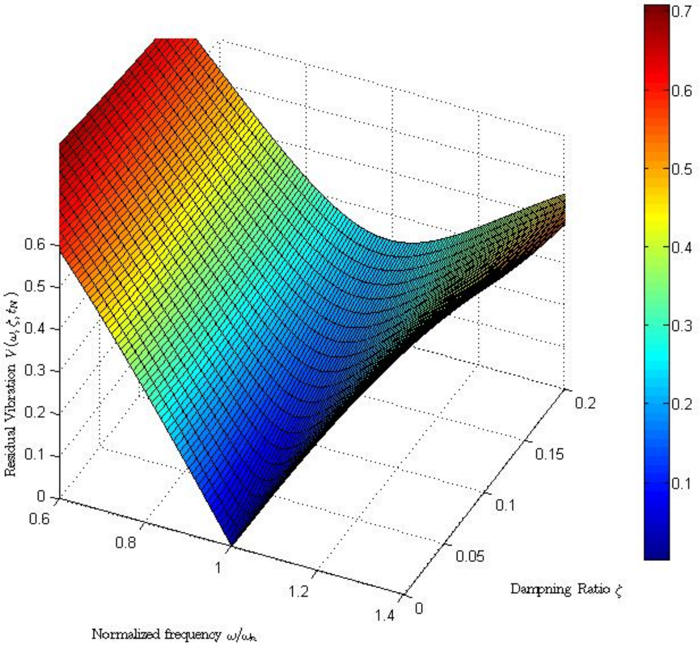


Figure 6.18: Three dimensional sensitivity curve for the ZV shaper.

6.6.4 Zero Vibration Derivative Shaper

The ZV Shaper that was described in the previous section, proved to be sensitive with regards to modeling error. In fact, in the two impulse ZV shaper discussed above, there can be a lot of vibration even for small modeling errors. This gave motivation to develop the more robust Zero Vibration Derivative (ZVD) Shaper. The three impulse ($m = 3$) ZVD shaper is a more robust shaper than the ZV shaper because it also satisfies the fist order robustness constraint

$$\frac{d}{d\omega}V(\omega,\zeta,t_N) = 0. \tag{6.42}$$

The physical meaning corresponds to zero amplitude and velocity of vibration at the end of the third impulse, whereas the ZV shaper only considered the amplitude of vibration.

The unknown variables t_2, t_3 A_1, A_2 and A_3 are found using the same approach as for the ZV shaper. The unknown variables are

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{1+2K+K^2} & \frac{2K}{1+2K+K^2} & \frac{K^2}{1+2K+K^2} \\ 0 & \frac{\pi}{\omega_d} & \frac{2\pi}{\omega_d} \end{bmatrix}, \tag{6.43}$$

where K is the same as for the ZV shaper. This shaper construct signals of the type shown in Figure 6.19.

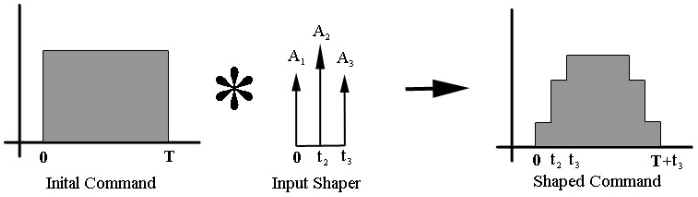


Figure 6.19: Continuous ZVD shaper.

Simulations show that this method reduces more vibration than the ZV Shaper, but it introduces an additional time delay. The payload x and y positions are shown in Figure 6.20, and the x and y errors are shown in Figure 6.21. The oscillations are now at approximately 3-4cm in amplitude.

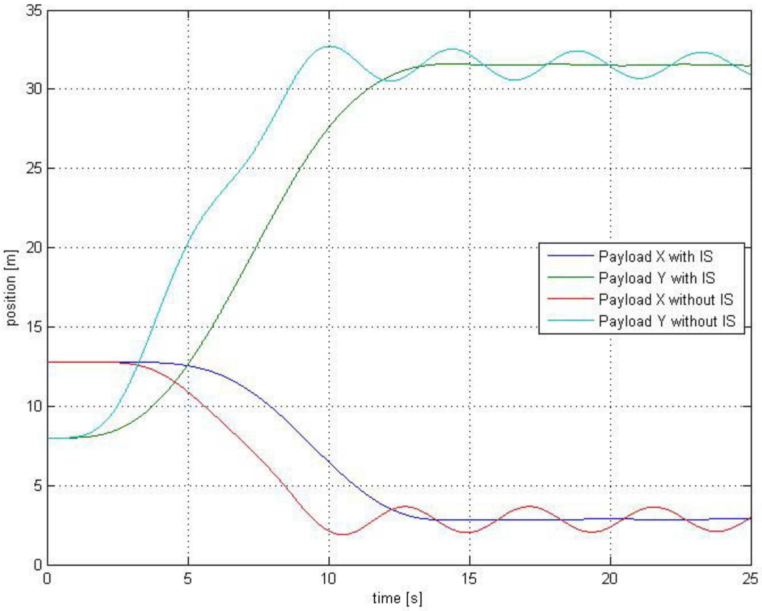


Figure 6.20: Payload position with, and without the continuous ZVD shaper.

Figure 6.22 shows the sensitivity curve for the ZVD Shaper. It is seen that the residual vibration is kept low, even when there are some modeling errors. This makes the ZVD Shaper suitable when one is not able to obtain perfect knowledge of the system.

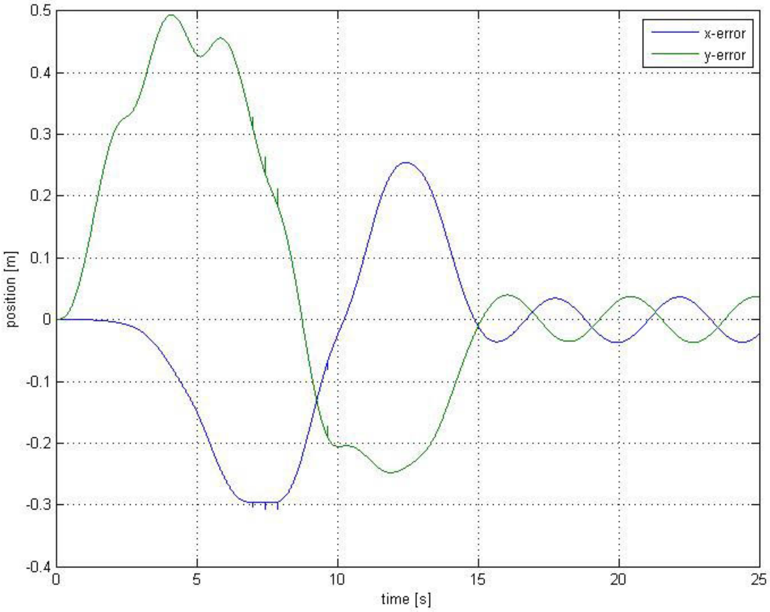


Figure 6.21: Difference between the crane tip and the payload position, using the continuous ZVD shaper.

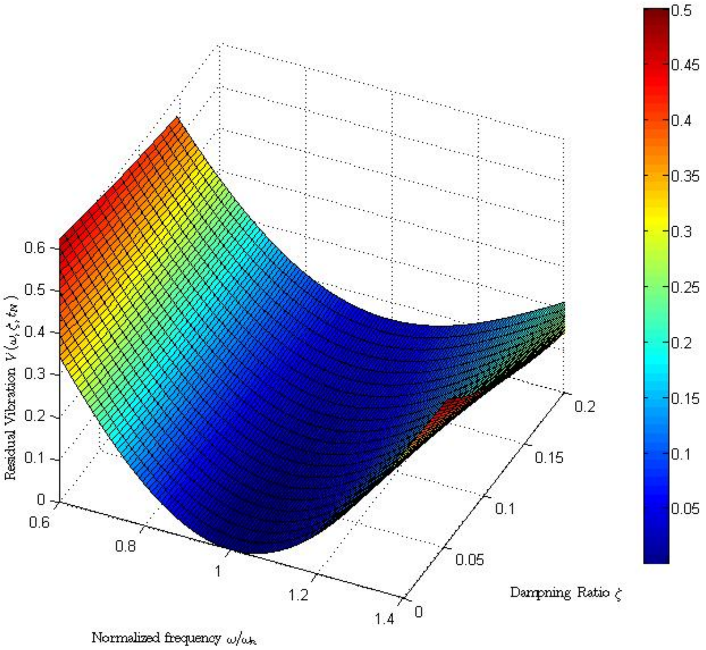


Figure 6.22: Three dimensional sensitivity curve for the ZVD shaper.

6.6.5 Zero Vibration Derivative Derivative Shaper

The Zero Vibration Derivative Derivative (ZVDD) Shaper is an even more robust shaper than the ZVD Shaper. It is a 4 impulse shaper, which makes m=4 in (6.22) and (6.23). The problem is therefore to find 7 unknowns. The only difference from the ZVD Shaper is that the second order robustness constraint is added

$$\frac{d^2}{d\omega^2}V(\omega,\zeta,t_N)=0. \tag{6.44}$$

The unknown variables are derived as for the ZV shaper. The ZVDD parameters are shown in (6.45), and the four impulse convolving process is illustrated in Figure 6.23. Simulation results for the ZVDD shaper are shown in Figure 6.24 and 6.25. These figures show that the shaper is able to reduce the vibration even more than the ZV and ZVD shaper. However, it also introduces more delay than the ZV and the ZVD shaper. The oscillations are now at approximately 2-3cm.

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{1+3K+3K^2+K^3} & \frac{3K}{1+3K+3K^2+3K^3} & \frac{3K^2}{1+3K+3K^2+K^3} & \frac{K^2}{1+3K+3K^2+K^3} \\ 0 & \frac{\pi}{\omega_d} & \frac{2\pi}{\omega_d} & \frac{3\pi}{\omega_d} \end{bmatrix}. \tag{6.45}$$

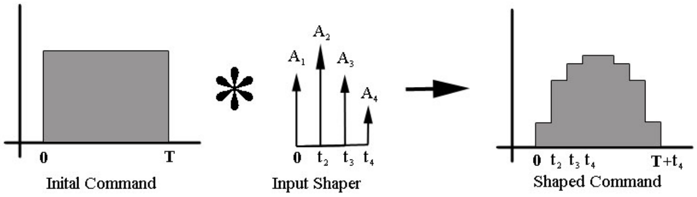


Figure 6.23: Continuous ZVDD shaper.

Figure 6.26 shows a 3 dimensional sensitivity curve for the ZVDD Shaper. The figure shows that the residual vibration is kept very low, even when there are modeling errors. The ZVDD Shaper is the most robust compared to the ZV and the ZVD. It is therefore the one that is able to keep the residual vibration at its lowest when there are modeling errors. If a robust shaper is needed, this shaper would be the best choice.

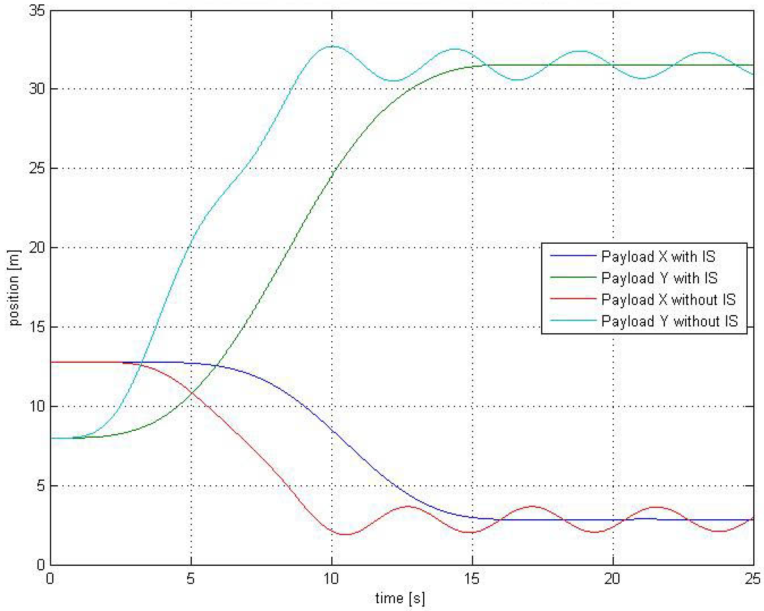


Figure 6.24: Payload position with, and without the continuous ZVDD shaper.

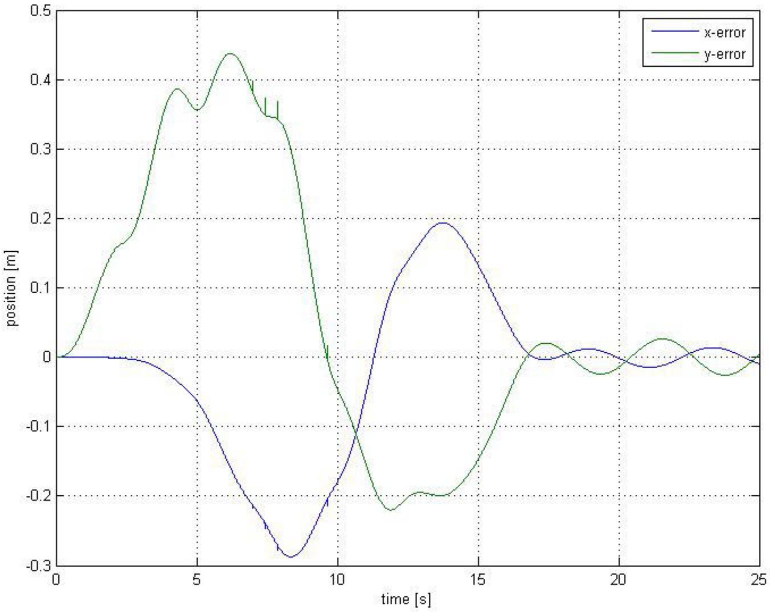


Figure 6.25: Difference between the crane tip and the payload position, using the continuous ZVDD shaper.

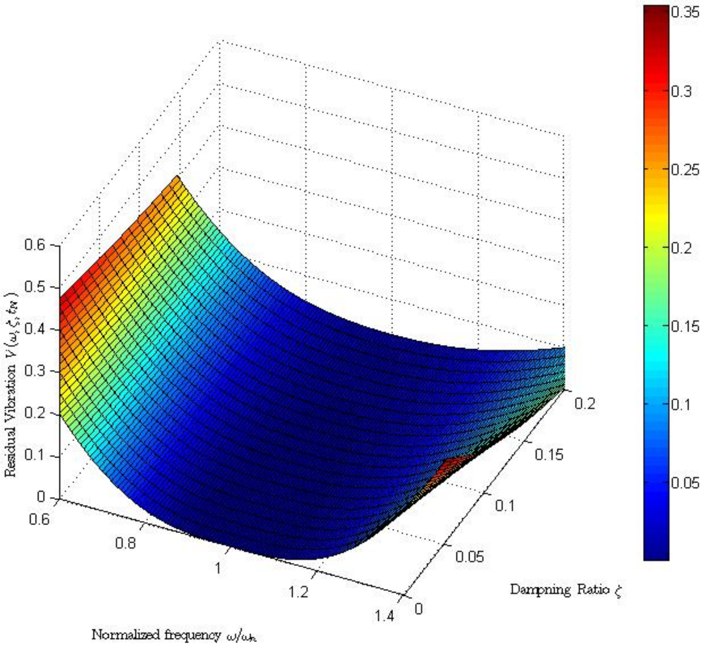


Figure 6.26: Three Dimensional Sensitivity Curve for the ZVDD shaper.

6.6.6 Discussion

The ZV shaper was able to reduce the vibrations to about 20cm, the ZVD to about 5cm and the ZVDD to about 2-3cm. For the unshaped case, the oscillation amplitude was approximately 1m. The three shapers have reduced the residual vibration significantly. Even better performance could possibly be achieved by tuning the damping and natural frequency.

As mentioned, more robustness leads to a greater delay in the operator input. The desirable robustness level has to be seen in connection with the operations the crane perform and the desired residual vibration. A figure that shows the three different shapers for the zero dampening case is shown in Figure 6.27. It is easy to see that the ZV vibration shaper is the least robust one, because it has a smaller width of frequencies of which it cancels the vibration. This was also seen from the simulation results and the 3 dimensional sensitivity curves.

It might also be possible to look at other types of shapers. For instance, the extra insensitive (EI) shaper, the unity magnitude zero vibration (UM-ZV) shaper and the specified intensity (SI), could be of interest. These shapers have not been considered due to similar delay and robustness properties.

All in all, the input shapers have proven to work very well on the vessel-crane-cable model. The input shaping technique has the potential to bring today's cranes towards a more demanding future.

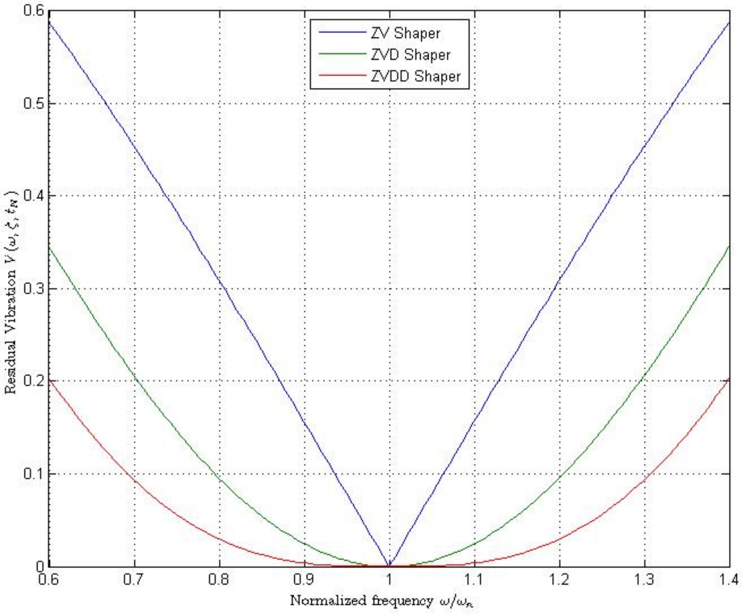


Figure 6.27: Two dimensional sensitivity curves for the three shapers.

Chapter 7

Conclusions and Further Work

7.1 Conclusions

The main focus of this thesis has been on developing an offshore crane model, both mathematically and through computer based software. Furthermore, three input shapers and two different workspace controllers have been developed in order to reduce residual vibration.

A mathematical model of a HMC3568 150t active boost crane was developed using robot modeling theory. The resulting equations and matrices were verified using Matlab's symbolic package. All matrices in the mathematical model were identical to the matrices Matlab generated. The vessel-crane model that was implemented in Simulink, was verified against a SimMechanics crane model. The two models showed almost identical behavior. However, small differences were found, which seemed to be a result of using two different controllers.

The Simulink model was later expanded to include a SINTEF cable model. The complete simulation tool consisted of a crane model, a vessel model and a cable model. During simulations, the total Simulink model responded according to what was expected.

A second order reference model were implemented in order to generate smooth reference signals. Furthermore, the reference model had incorporated position, velocity and acceleration saturations. Together with the PD controller with gravity compensation, the system became stable.

Two inverse kinematics algorithms were implemented and tested. Both the Jacobian transpose method, and the *lsqnonlin* method, were able to position the crane tip in $\{N\}$. The Jacobian Transposed method however, was not able to constrain the solution to a feasible area. This resulted in crane configurations that were invalid on the HMC3568 crane. Furthermore, the *lsqnonlin* method turned out to be superior to the Jacobian transposed method. The *lsqnonlin* method showed better performance without waves, and in sea state 2, 4 and 6. In sea state 6, the vessel motion became more extensive, making the crane unable to keep a fixed position. This was most likely an effect of saturations on the velocity and acceleration.

Three input shapers have successfully reduced the payload swing. Without the shapers, the vibration were approximately 1m in amplitude. When the operator input was shaped by the ZVD shaper, it was reduced to about 5cm. This is a significant change in payload swing, and is expected to be of great interest to crane suppliers. The down side of it, is that the shaped command is delayed compared to the unshaped one. However, it would be safer and more efficient to use the shapers than to manually damp the payload swing. As a result, the input shaping technique has the potential bring the crane industry one step closer to what is needed in the future.

7.2 Recommendations for Further Work

Further work on this subject should include testing of the input shapers. They have proven to work in the Simulink environment which makes the next natural step to perform experiments and full scale tests. For this reason, a set up that is able to measure the residual vibration has to be installed. For the time being, cranes of this type do not have a device that measures the payload position or the cable angle.

In the future, the crane system might be able to communicate with the vessel system. Therefore, the crane system could be extended to automatically set the desired vessel position and orientation. As a result, the vessel, instead of the crane, could reduce residual vibration. This could be advantageous when the residual vibration periods are large. The crane are be unable to reduce the residual vibration when the oscillation amplitudes are larger than the crane's operating radius.

Another area of improvement, is the inverse kinematics algorithms. The Jacobian transpose method produces infeasible configurations, which is unwanted. This has to be dealt with. In addition, simulations with the cable model should be performed and analysed.

Another area of improvement, is how reference signals are generated. The reference model is tuned to generate similar inputs as what the crane operator would have generated. However, further work should be done by making the reference signals more realistic. For this purpose, the GUI input block could be replaced with joysticks.

Bibliography

Chang, T., Jaroonsiriphan, P., Bernhardt, M. and Ludden, P. (2006), ‘Web-based command shaping of cobra 600 robot with a swinging load’, *Industrial Informatics, IEEE Transactions on* **2**(1), 59 – 69.

Coleman, T. F. and Li, Y. (1994), ‘On the convergence of interior-reflective Newton methods for nonlinear minimization subject to bounds’, *Mathematical Programming* **67**(1), 189–224.

Coleman, T. F. and Li, Y. (1996), ‘An Interior Trust Region Approach for Nonlinear Minimization Subject to Bounds’, *SIAM Journal on Optimization* **6**(2), 418–445.

Garrido, S., Abderrahim, M., Giménez, A., Diez, R. and Balaguer, C. (2008), ‘Anti-swinging input shaping control of an automatic construction crane’, *IEEE T. Automation Science and Engineering* **5**(3), 549–557.

Gieck, K. and Gieck, R. (1996), *A collection of technical formulae*, 8th english ed. edn, Gieck Verlag.

Khalid, A., Singhose, W., Huey, J., Lawrence, J. and Frakes, D. (2004), Study of operator behavior, learning, and performance using an input-shaped bridge crane, *in* ‘Control Applications, 2004. Proceedings of the 2004 IEEE International Conference on’, Vol. 1, pp. 759 – 764 Vol.1.

Love, L. J., Jansen, J. F. and Pin, F. G. (2003), Compensation of Wave-Induced Motion and Force Phenomena for Ship-Based High Performance Robotic and Human Amplifying Systems, Technical report, Oak Ridge National Laboratory.

MacGregor (2010), *Course Compendium.* Confidential.

Mark W. Spong, Seth Hutchinson and Vidyasagar, M. (2006), *Robot Modeling and Control*, John Wiley and Sons, Inc.

Mitchell, S. O. J. (1958), *Feedback control systems*, McGraw-Hill, New York :.

Sciavicco, L. and Sciavicco, B. (2000), *Modelling and Control of Robot Manipulators*, 2nd edn, Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Singhose, W. E., Kim, D. and Kenison, M. (2008), 'Input shaping control of double-pendulum bridge crane oscillations', *Journal of Dynamic Systems Measurement and Control* **130**(3).

SINTEF (2010), *FhSim Overview*, 1.0 edn. Confidential.

Skagestad, R., Viõarsson, r. and Vilbergsson, K. S. (2008), Simulation and control of active heave compensated crane, Master's thesis, Aalborg University, Department of Control Engeneering. Confidential.

Sorensen, K. L., Singhose, W. and Dickerson, S. (2007), 'A controller enabling precise positioning and sway reduction in bridge and gantry cranes', *Control Engineering Practice* **15**(7), 825 – 837.

Subsea7 (2011), 'Seven sister', `http://www.subsea7.com/files/docs/Datasheets/Vessels/Seven_Sisters.pdf`.

Thor I. Fossen (2010), *Guidance and Control of Marine Craft*, Norwegian University of Science and Technology.

Tuttle, T. and Seering, W. (1995), Vibration reduction in 0-g using input shaping on the mit middeck active control experiment, *in* 'American Control Conference, 1995. Proceedings of the', Vol. 1, pp. 919 –923 vol.1.

Vaughan, J., Yano, A. and Singhose, W. (2008), 'Comparison of robust input shapers', *Journal of Sound and Vibration* **315**(4-5), 797 – 815.

Wittenmark, B., Åström, K. J. and Årzén, K.-E. (2003), Computer control: An overview, Technical report, IFAC Professional Brief.

Yuan, D. and Chang, T. (2006), Trajectory control of robot with model reference zero vibration shaper, *in* 'Mechatronics, 2006 IEEE International Conference on', pp. 636 –641.