Model Predictive Control of 3-DOF Articulated Robotic Manipulator

Endris Muhammed Hassen

A Thesis submitted to

The department of Electrical Power and Control Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Electrical Power and Control engineering (Control Engineering)

Office of Graduate Studies

Adama Science and Technology University

June, 2022

Adama, Ethiopia

Model Predictive Control of 3-DOF Articulated Robotic Manipulator

Endris Muhammed Hassen

Major Advisor: Prof. Samsun Ma

Co-Advisor: Mr. Endrias Alemayehu

A Thesis submitted to

The department of Electrical Power and Control Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Electrical Power and Control engineering (Control Engineering)

Office of Graduate Studies

Adama Science and Technology University

June, 2022

Adama, Ethiopia

DECLARATION

I hereby declare that this Master Thesis entitled "Model Predictive Control of 3-DOF Articulated Robotic Manipulator" is my original work. That is, it has not been submitted for the award of any academic degree, diploma or certificate in any other university. All sources of materials that are used for this thesis have been duly acknowledged through citation.

Name of student	Signature	Date

RECOMMENDATION OF ADVISORS

We, the advisors of this thesis, hereby certify that I have read the revised version of the thesis entitled "Model Predictive Control of 3-DOF Articulated Robotic Manipulator" prepared under our guidance by Endris Muhammed Hassen submitted in partial of the requirements for the degree of Masters of Science in Electrical Power and Control Engineering. Therefore, we recommend the submission of revised version of the thesis to the department following the applicable procedures.

________________________	________________________	________________________
Major Advisor	Signature	Date
________________________	________________________	________________________
Co-Advisor	Signature	Date

APPROVAL PAGE

We, the advisors of the thesis entitled "Model Predictive Control of 3-DOF Articulated Robotic Manipulator" and developed by Endris Muhammed Hassen, hereby certify that the recommendations and suggestions made by the board of examiners are appropriately incorporated in to the final version of the thesis.

Major Advisor	Signature	Date
Co-Advisor	Signature	Date

We, the undersigned, members of the Board of Examiners of the thesis by Endris Muhammed Hassen have read and evaluated the thesis entitled "Model Predictive Control of 3-DOF Articulated Robotic Manipulator" and examined the candidate during open defense. This is, therefore, to certify that the thesis is accepted for partial fulfilment of the requirement of the degree of Master of Science in control engineering.

Chairperson	Signature	Date
Internal Examiner	Signature	Date
External Examiner	Signature	Date

Finally, approval and acceptance of the thesis is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate committee (SGC).

Department Head	Signature	Date
School Dean	Signature	Date
Office of Postgraduate Studies, Dean	Signature	Date

ACKNOWLEDGEMENT

First and foremost, I want to express my gratitude to the Almighty Allah for providing me with the strength to continue through this difficult thesis journey. Next my gratitude goes to my Major advisor, for prof. Samsun Ma and co-advisor Mr. Endrias Alemayehu for their inspiration, motivation, invaluable support and constant encouragement during my thesis work. I want to say thank you to prof. Kim and prof. Jin for their open mind and helping me with giving an idea and motivational word in order to accomplish my thesis. Finally, I want to express and send my gratitude and blessing to all of my family and others those assisted me for the accomplishing of thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBRIVATIONS

3-D	Three dimension
AMPC	Adaptive Model Predictive Control
CTC	Computed Torque Control
D-H	Denavit-Hartenberg
DOF	Degree of Freedom
FBL	Feedback linearization
FSMC	Fuzzy Sliding Mode Control
GA	Genetic Algorithm
GUI	Graphical User Interface
$H\infty$	H-infinity
HT	Homogenous Transformation
LPV	Linear parameter varying
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regulator
MATLAB	Matrix Laboratory
MIMO	Multi-Input Multi-Output
MPC	Model Predictive Control
PID	Proportional Integral Derivative
PMDC	Permanent Magnet DC Motor
SMC	Sliding Mode Control
ZN	Ziegler Nicholas

LIST OF SYMBOLS

$C(\theta)$	Centripetal
F_s	Columb friction coefficient
$B(\theta)$	Coriolis
ξ	Damping ratio
α	First gain of MPC
G	Gravity
$M(\theta)$	Inertia
ω_0	Natural frequency
p	Prediction horizon
β	Second gain of MPC
χ	Synthetic control
τ_d	Torque disturbance
F_v	Viscous friction coefficient
w	weight factor

ABSTRACT

Many researches has been done on robotic manipulator. Robotic manipulators are used for jobs that are too dirty, dangerous, repetitive for humans to perform. Due to these reasons, robotic manipulator research is one of the most fascinating and exiting fields in industrial technology. Robotic manipulators are nonlinear systems with many inputs and outputs, parametric variation, external disturbance, and a dynamic capability that changes rapidly. The most difficult difficulties to control and track in robotic trajectory tracking are uncertainty and time-varying dynamic systems. Denavit-Hartenberg parameters and rotation matrix around axis's are employed for forward kinematics in this thesis, whereas algebraic technique is used for inverse kinematics of three degree of freedom (3-DOF) robots. The non-linear dynamics equation of motion of the 3-DOF industrial articulated robot is examined using the Euler-Lagrange equation of motion. Because of the system's dynamic model is nonlinear, the feedback linearization approach is used to create a linear system. A predictive control law with a minimizing objection function is then built for the obtained linear model. The settings of the model predictive control (MPC) controller were fine-tuned utilizing torque constraint analytic studies, and a second order system was considered for it. The controller obtained the intended outcome in MATLAB/Simulink software. Step, ramp and sine wave tracking are discussed for proposed MPC controller. The suggested controller is an improved approach of process control that uses little effort to track a process with optimum parameter while satisfying input constraints and reduced tracking error. To evaluate the applicability of the recommended control technique with proportional integral derivative genetic algorithm (PIDGA) controller, a simulation and results analysis are provided. For the 1^{st}, 2^{nd} and 3^{rd} joint, the maximum energy required to track the desired signal for PIDGA and MPC is 24.8649 Nm, -8.8917 Nm, -7.3159 Nm and 22.9040 Nm, -8.4805 Nm, -6.3049 Nm respectively. The maximum overshoot happened to control the 1^{st}, 2^{nd} and 3^{rd} joint articulation for PIDGA and MPC controller is 3.7214%, 5.9657%, 4.0252% and 0.275%, 0.275%, 0.275% respectively. According to the results, the proposed MPC controller outperforms the PIDGA controller in terms of performance tracking and energy used.

Keywords: 3-DOF Robotic arm, Feedback linearization, Kinematic, Dynamic, MPC

CHAPTER ONE

INTRODUCTION

1.1 Background

Now days, the majority of our world's industries use robots to enhance output and profit. Robots are any autonomously operated machine that takes the places of human being effort. Thus, they have become increasingly popular in recent years as a means of doing tasks faster than humans. Robotics' purpose is to create intelligent machines that can help and assist humans in their daily lives while also keeping everyone safe. Robotic applications are divided into two categories: industrial and service. In an industrial setting, a robot has a lot of appealing qualities. Among those features, Robots are being used in almost every industry to boost productivity, precision, flexibility and quality (Mahil & Al-durra, 2016). An electronically operated arm link serial manipulator with numerous joints makes up the robotic manipulator. In the realm of robotics, robot arm control is one of the most active research fields, which is owing to the high command of the industrial field in order to enhance production as well as make the difficult and repetitive work for the field of operator easier (Raza et al., 2018).

Robotic manipulators are used for jobs that are too dirty, dangerous, repetitive for humans to perform (Ghaleb & Aly, 2018). Due to these reasons, robotic manipulator research is one of the most fascinating and exiting fields in industrial technology. An industrial articulated robot is industrial legged robot with rotary joint. It can range from simple two-joint constructions to complex multi-joint ones and it is the most commonly industrial robot used in the world wide (Jin et al., 2018). Therefore, cutting, spraying, painting, material handling, picking, welding, painting, assembling, and sealing are just a few of the applications for articulated robots used in the manufacturing industry (Agbaraji et al., 2017). Therefore, one of challenging problem to do the above applications area task is controlling robot. In order to control robot, controller is the most important component for complex and sophisticated plant dynamic system. The presence of uncertainty and a highly time shifting dynamic system are the most difficult problems to regulate when tracking robotic trajectory (Piltan et al., 2012).

In (Elkhateeb & Badr, 2017), developed a unique PID controller tuning mechanism for manipulator robot trajectory tracking control. Using a dynamic inertia weight artificial bee colony optimization approach, the best gains of a proportional integral derivative controller are realized.

MPC is an optimum best control technique that anticipates the system's future behavior based on its existing states and responses. The optimum control signal is subsequently created, taking into account both the forecast results and previous behavior. An in-depth analysis of MPC-based control techniques is offered (Bemporad, 2006). The proposed MPC technique possessed several advantages like, good transient response, fast response, almost zero overshoot and insensitive to external disturbance. So, the proposed model predictive control for 3-DOF robotic manipulator considering disturbance and joint friction is used to improve also for constraint problem. Since trajectory tracking control of robotic manipulator is a challenging task due to its complexity of dynamics and uncertainty. External disturbance on the system model cause unstable performance of the control system. In addition to this, most research works have been observed to neglect disturbance and joint friction. Thus, reduced the trajectory tracking control performance of the system. Therefore, this proposed controller would have addressed the loss of system stability due to neglected disturbance and friction by taking in to account disturbance and joint friction in the system to improve the trajectory tracking control performance of the system (Ehsan, 2010).

The majority of research have used diverse controllers to construct and control 2-link and 3-link robotic manipulators without disturbance and joint friction. PID controllers are used to regulate the majority of linear and nonlinear systems in industry. Thus, resulted reduced trajectory performance control of robotic manipulator. This thesis makes a contribution by ensuring and overcoming the difficulty of PID trajectory tracking performance of the robotic manipulator by taking into account disturbance torque due to payload variation and vibration, joint friction, and satisfying input limitations on the system's dynamic model. Moreover, the other contribution is filling the following research gap that seen and observed in the previous journal paper. Firstly, when researchers used PID approach the constraint problem was not included this makes the system out of control and comes out high overshoot and oscillation result due to small adjustment of controller. Second, many researchers failed to address both friction and disturbance, despite the fact that these two elements affect the real-time plant system. Third, many studies were solely interested in tracking the step set point tracking. Finally, the performance of the time domain specification for the resulting robotic manipulator system approach technique should be enhanced, as discussed in the previous review.

In this thesis model predictive control is chosen and the dynamic of three degree of freedom robot is derived considering friction and disturbance. The construction of MPC control rules for a three-degree-of-freedom articulated robotic arm with serially connected links and joints was reported in this study. The resulting kinematics and dynamics of the robotic arm were used to develop control rules. In order to observe the efficiency of the proposed control approach, PIDGA is deigned and finally compared through tracking results.

1.2 Statement of the problem

Robotic manipulators are created from a sequence of links and joints. They are very complex multi-input multi-output nonlinear systems with uncertainty and dynamic capabilities that change over time. Gravitational forces that are depending on position, carioles, and centrifugal force are the result of nonlinearity and coupled dynamicity of system. These qualities make controlling the trajectory of a robotic manipulator difficult. Additionally, there are several uncertainties in the system model, such as external disturbance, which cause the control system's performance to be unstable. In a system model, introducing joint friction can add complexity to the model's overall dynamics. Most research works have been designed and control without having disturbance and joint friction for 2-link and 3-link of robotic manipulator using PID, Fuzzy sliding mode control (FSMC), MPC and others. Especially, in industry most of linear and nonlinear systems are controlled by PID controller. Thus, resulted reduced trajectory performance control of robotic manipulator. The contribution of this thesis is that to ensured and overcome the problem of PID trajectory tracking performance of 3-DOF robotic manipulator by considering disturbance, joint friction, and satisfying input constraints on the dynamic model of the system. Model predictive control proposed to handle the challenge of handling uncertainties and time variable dynamics with the goal of boosting system stability to produce a better result.

1.3 Objectives of thesis

1.3.1 General objectives

The general objective of this thesis is to design MPC control law to get the robotic arm manipulator to track the intended trajectory input in MATLAB/Simulink.

1.3.2 Specific objectives

- Formulation of kinematics and dynamics modeling of 3-DOF articulated robotic arm
- Interface and simulate model predictive control law with plants in MATLAB/Simulink
- Handling disturbance uncertainty, instability and checking disturbance capability of MPC controller
- Evaluate performance of the developed MPC control with PIDGA in MATLAB/Simulink
- Based on the system's output, making appropriate conclusions and recommendations

1.4 Scope of the study

The thesis would have involved controlling 3-DOF articulated robotic manipulator using model predictive control in MATLAB/Simulink. The simulation result is evaluated using efficiency performance criteria. The system is considered by taking in to account disturbance, friction joint model and constraint problem to improve the desired trajectory tracking control performance of approximate real system. Here, the thesis focusses on the dynamic modelling and simulation design for 3-DOF robotic arm. Depending on the dynamic modelling the MPC control approach is designed. Finally, PIDGA is created to test the efficiency of the proposed controller.

1.5 Limitations of the study

The thesis is limited to consideration of actuators of dynamics. So, it may affect the performances of the system in real time applications. Formulating the 3-DOF dynamic model analytically is complicated, which means the calculation will take a lengthy time and difficult. As a result, an error in the system's model will possible. That makes reduced the performance of system. Therefore, computerized system should be developing for overcoming the model error by finding dynamics equation and resulted enhancing the performance. Implementation of the system will not have performed practically which is the limitation of the study.

1.6 Motivation of the study

The main reason for choosing this thesis over others is that enjoying on robots. Because of the dynamics element is so complicated, a very little study has been done on 3-DOF robotic manipulators. Many of the studies had two degrees of freedom and a straight forward mathematical model. Another reason for the study's motivation is that it has a high degree of freedom, which implies the end effector can choose from a larger variety of alternatives. Robots can work in dangerous environments, freeing humans from dangerous jobs. They can move and lift heavy objects, handle hazardous materials, and perform repetitive tasks. As a result, optimal MPC controller is developed for this electromechanical system such that it can travel in any direction to complete a given task.

1.7 Significance of the Study

Aside from being accurate and consistent, robots may operate in any setting, increasing their versatility. Robots can work in hazardous areas and eliminate risky employment for humans. They are capable of lifting big items, handling dangerous substances, and doing repeated jobs. This has helped businesses avoid numerous accidents while also saving time and money. Robots can reach and fit where humans can't, allowing for greater precision. Industrial robots have improved productivity, safety, and efficiency. Robots can provide extremely accurate, consistent, and high-quality work without taking vacations or breaks. As a result, the proposed controller controlled the trajectory control of the robotic manipulator in order to provide the robot with the desired functionality.

1.8 Thesis organization

This thesis is organized in to five chapters.

Chapter 1: - The research work's history, statement of the problems, study aims, scope, limitations, and importance are all described.

Chapter 2: - The literature on 3-DOF articulated robotic manipulators and related research is reviewed in this chapter.

Chapter 3: - The kinematic and dynamic models of a 3-DOF articulated robotic arm are described, as well as the materials and methods used.

Chapter 4: - The control design is also discussed in this chapter.

Chapter 5: - The simulation result for proposed MPC controller are analyzed and discussed with PIDGA controller.

6. Conclusions and suggestions are included in this chapter.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

A robot is a multifunctional machine that performs many tasks by moving parts, materials, tools, or specialized devices through varied programmed motion. It takes the right combination of actuator, manipulator, and robot controller to complete a range of tasks. To properly track the desired input, the robot end-effecter manipulator requires an adequate selection of best controller (Kim et al., 2020).

2.2 Related work on robotic manipulator

Various designing, modeling and control strategies have been presented in different researches for trajectory tracking control of robotic manipulator for n-degree of freedom. In this section a review of the researches which focus on trajectory tracking control of robotic manipulators using different controllers are presented. In general, the studies in the literature review focused on controlling and designing of industrial robotic manipulator either in classical or modern control.

In (Bouzoualegh, 2019), the authors developed model predictive control of three degree of freedom manipulator robot. First, the end effector manipulator robot's coordinates were determined using a geometric model. The kinematic model of the planar two-link robot was the derivative of the direct geometric model. The concept involved using a feedback linearization method for the robot's nonlinear dynamic model, and then developing an MPC control based on the linear model created by including the predictive controller output as a constant within the prediction horizon. The authors presented an MPC control of a three-DOF manipulator planar two-link robot with a new cost function and a new term to reduce energy consumption. A second order closed loop system was studied in order to acquire a better tune of an MPC control parameter based on torques analytical analysis. Thus, they got good tracking result for step desired set point tracking using better optimal gain of MPC controller. Finally, the authors proposed that future research should focus on applying this predictive control law to more degrees of freedom and challenging the acquired rule to follow the set point trajectory.

In (Ghaleb & Aly, 2018), the authors presented a modelling, simulation and control of 2-DOF robot arm. The kinematics of a robot arm with two degrees of freedom were calculated using

Denavit-Hartenberg parameters, and the dynamics equation of motion for two degrees of freedom was determined using the Euler-Lagrange method. For the modeling and controlling technique, a control design was created using a traditional linear PID controller. For the sake of simplicity, the authors assumed that an arbitrary torque cannot be commanded at the robotic manipulator's joint and external disturbance. Finally, the system rapidly stabilized, however a small adjustment in the controller parameters resulted in further overshoot and oscillation due to the very sensitive initial and final locations. In the future, the authors intend to tweak and turn the gains of the PID controller at every instant to prevent overshoot and oscillation caused by changes in parameter values.

The author created model predictive rules for an articulated robotic arm in (Iqbal, 2019). The Euler-Lagrange equation was utilized to solve the dynamics of the robotic arm manipulator, and Denavit-Hartenberg parameters were employed to derive kinematic modeling. To construct the control legislation, the author looked into two recent control strategies (MPC and H∞). The designed optimal control laws were critical for transient metrics such as steady state error and overshoot, as well as settling time, to be examined. Although both designed optimal control laws provide adequate tracking performance, a comparison of system performance by both controllers demonstrates that MPC outperforms than H∞. The size of the prediction window was raised to reduce overshoot in MPC response. The author expects to implement both control law approaches on a genuine robotic platform in the future.

In (Naing et al., 2018), authors used PID controller to control the position of a 3-DOF articulated robot arm. The Arduino microcontroller was used to generate the required angular position of the robot joint, which was used to choose and place an object. The Denavit-Hartenberg parameter and analytical method were used to create a robotic forward and inverse kinematic model. The results of inverse and forward kinematics were displayed in the MATLAB GUI The scientists used a PID controller and a link length arm to compute the precision of a motor in each joint in order to enable transporting the desired object that a robot can hoist. The Ziegler-Nicholas (ZN) approach was used to tune the gain of the PID controller, and this controller was used to find the real position of the robot arm.

A comparative study on MPC control and LQ optimal control of 2-link robot arm was presented in (Guechi et al., 2018). MPC and LQ control trials were conducted in this study. The Euler-Lagrange method was used to develop the dynamics equation of motion with two degrees of freedom. A control strategy for a plant model is designed using a dynamic robot model. Minimizing the cost function yielded the MPC parameters furthermore, a simulation study was conducted to compare the suggested MPC and LQ control approaches, which are both based on the same feedback linearization, with a control strategy provided in the literature for the same problem. Finally, based on the findings, MPC control outperforms LQ control in terms of system performance. The authors intend to test the proposed MPC control on a real robot in the future to ensure the method's robustness.

In (Cisneros et al., 2018), the authors developed constrained predictive control of 2-DOF robotic manipulator using quasi-LPV representations. Nonlinear Model Predictive Analysis was used in this study. The control of a robotic manipulator in a realistic way is provided, with viscous friction torque on the system taken into consideration. The authors employed quasi-LPV modeling and velocity-based linearization to place efficient quadratic programming quasi-linearity. Finally, the experimental results revealed that the built integral action of velocity increased the effectiveness of the control approach algorithm and overcame the robustness of the robotic manipulator's unmolded dynamics impact. The authors concluded that a viable approach control technique was implemented successfully.

In (Moe Myint et al., 2016), the authors demonstrated a method for controlling the position of a robotic manipulator arm in an object sorting system. This research study presented an embedded control strategy for robotic arm systems that pick and arrange objects to reduce manual control. Denavit-Hartenberg parameters were used to calculate the kinematics of a robot arm. The end effector location of a robotic manipulator arm was constructed using forward kinematics in this study. The scientists predicted that in the future, it may be engineered to provide the best grip and a material with image processing capabilities to determine the object's location. Finally, the results of the experimental tests and the analytical calculations were identical.

In (Önkol & Kasnakoğlu, 2018), the authors developed adaptive model predictive control approach for a two-wheeled robot manipulator with varying mass. The Euler-Lagrange equation is used to calculate the dynamics of a two-wheeled robotic arm manipulator. The linear parameter varying model was presented at various values of variable parameter included with linear local model. In the inner loop, an adaptive model predictive controller is used to regulate the center of gravity angle, while in the outer loop, Linear Quadratic Gaussian is used to control linear position. The suggested approach's closed loop response performance is compared to PID/PID, FBL/LQG, and LQG/LQG with inner/outer loop controllers, and they discovered a good and superior result in terms of reference tracking, as well as good robustness to mass variation. The metrics performance in the AMPC control approach method was excellent.

In (Obasi et al., 2020), the authors developed a computational model dynamic of 3-links articulated robotic manipulator. The Euler-Lagrange formulation approach was used to create the dynamics equation for a 3-link articulated robotic manipulator. The energy and torque that produce robot motion at each joint were estimated independently in this study. Finally, the computed value specifies the motor rating that will be used in the real-world application. It was not adequate to describe the plant by expressing and computing system dynamics. As a result, the kinematics section should be included. The goal of this study is to use the MPC control approach to control a 3-DOF articulated robotic manipulator.

By looking and observing the above related work, I have got some gaps of research. Finding and defining problem is one of methodology that we pass in engineering research. One way of obtaining statement of the problem is the effectiveness and robustness of the controller. Moreover, the trouble came under the idea that has not been done in the past and adding consideration of uncertainty that includes in real time application. The research gap that obtained from previous review is generally defined in this thesis.

The research gap that observed and solved in this thesis are: firstly, when researchers used PID approach the constraint problem was not solved this makes the system out of control and comes out high overshoot and oscillation result due to small adjustment of controller. Secondly, many researchers did not have considered both friction and disturbance, but in real time application these two are the factors that affect real system of plant. Thirdly, many researchers were track only the step set point tracking. Finally, in previous review the time domain specification performance for the resulted robotic manipulator system should to be improved and it is more flexible when the robot arm has more degree of freedom. These gaps are planned, and statement of the problems are solved in this thesis. In table 2.1 below, the summary of the literature review in short and compact table format is presented to understand the previous overview related work in what way they passed the approach method, achievement and gaps of research.

Table 2.1 Summery of literature review

Authors	Title	Methodology	Achieve	Gap
(Ghaleb & Aly, 2018)	Modelling and control of 2-DOF robot arm	Traditional PID controller	System finally Stabilized	-overshoot, due to sensitivity places Initial and final Location -Initially oscillation (instability)
(Bouzoualegh, 2019)	3-DOF manipulator robot	MPC	Good set point tracking result	-did not have Consider both Friction and Disturbance (uncertainty)
(Iqbal, 2019)	Control of articulated robotic arm	Model predictive rules	MPC outperforms than H∞.	-only tracking step input
(Naing et al., 2018)	control the position of 3-DOF articulated robot arm	PID controller	Obtained the real position of robotic arm	-Less performance in time domain -constraint problem (instability of input energy)
(Cisneros et al., 2018)	constrained predictive control of 2-DOF robotic manipulator	Using quasi-LPV representations	overcame the robustness problem	-did not have consider disturbance uncertainty and coulomb friction -simple DOF

CHAPTER THREE

SYSTEM MODELLING

3.1 Overview

In this chapter, the material and system methodology of the 3-DOF articulated robotic manipulator is presented. The model predictive control is proposed to track the desired set point tracking of robotic arm.

3.2 Materials

This thesis makes use of MATLAB/2021a, Microsoft Office Word 2016, and Math Type 6.0 equation editor. MATLAB is a computer application that incorporates Simulink and technical toolboxes. Microsoft Office Word is used to write and revise a thesis documentation. In Microsoft Office Word, Math Type 6.0 is used to write mathematical formulas and equations.

3.3 Method/Procedure

This thesis' technique relies primarily on mathematical and software algorithms. A flowchart will be used to create the thesis, which will identify all of the steps that must be completed. The first procedure for doing this thesis was reading literature review from different journal paper on robotic manipulator. The literature reviews are discussed in chapter two and they were previous related work. After that, by calculating the kinetic and potential energy for each link of robot arm, the mathematical model is formulated based on Euler-Lagrange method. Therefore, the resulting dynamic model for 3-DOF articulated robot arm is non-linear system. For obtaining information about the system on the behavior of equilibrium point, the feedback linearization is applied for the robotic manipulator. Then, the model predictive control law is designed for decoupled linear system and perform simulation MATLAB/SIMULINK. For different value of natural frequency, the torque is computed and getting prediction horizon and weighting factor for the system. Finally, by finding the optimal parameter of model predictive control using the prediction horizon and weighting factor, the simulation result is analyzed and compared it with proportional integral derivative genetic algorithm to looked the efficiency of the proposed controller. in Figure 3.1 below, the work flow chart of this thesis is presented.

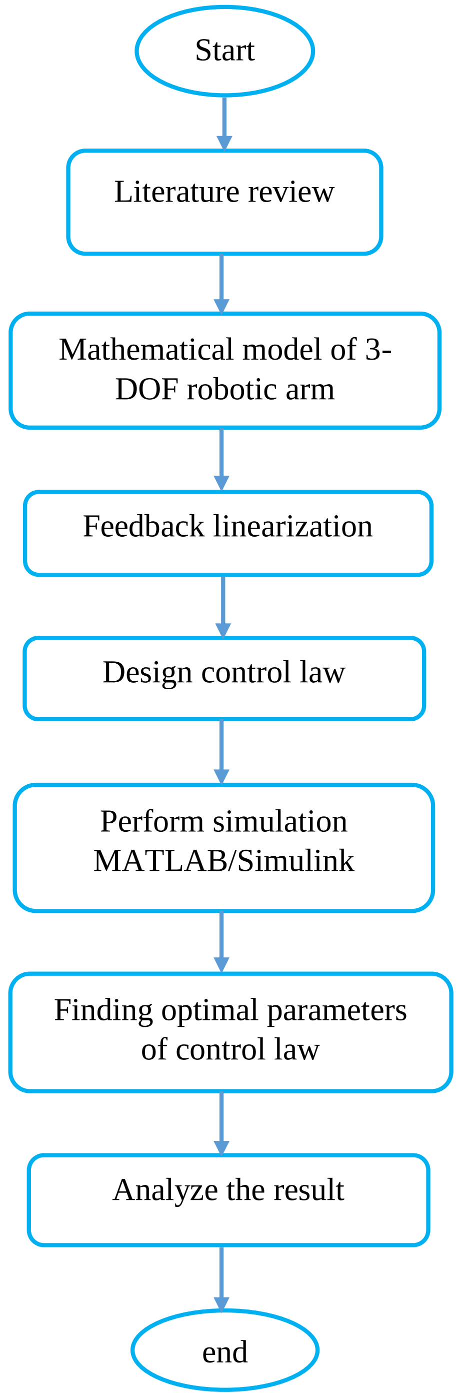


Figure 3.1 Work flow

3.4 Description of 3-DOF robotic arm

A rigid body is a collection of particles in which the distance between any two points of a particle remains constant and independent of the external force acting on the body. The rigid body is the link in robotics, representing an idealized solid body that cannot deform (Fateh, 2008). The rigid body coordinate system end effector travels in two directions: translational and rotational. The positioning of objects in three-dimensional space is a focus of robotics research. Position and orientation properties are used to define rigid body items. We always attach the frame rigidly to the object in order to indicate the orientation and location of the body in space. Based on the structure of connectivity, robots are either serial or parallel (Voisembert et al., 2012). One of the most fascinating fields of industrial technology study is robotic manipulator research. The six basic types of industrial robots are cylindrical, SCARA, delta, Cartesian, polar, and articulated robots. Articulated robots are the most prevalent industrial robot in use across the world. An articulated robot is an industrial robot with linkages and a rotating joint. It can range from simple two-joint constructions to complex multi-joint ones. Cutting, spraying, painting, material handling, picking, welding, assembling, and sealing are all applications for articulated robots in the manufacturing industry. It has a large work area and functions in three dimensions. Industrial articulated robots are made up of several serially connected links with joints that allow any link connected to nearby bodies to be flexible (Tourassis & Ang, 1992). The component is moved linearly in prismatic and rotated in a revolute method by the robot joint. The number of degrees of freedom is the number of independent variables that can be used to move the end effector location. Each joint position in industrial robotics is defined by a single variable that indicates the number of degrees of freedom. As a result, in robotics, the number of degrees of freedom is the most significant quantity to consider while analyzing kinematic and dynamic equations of motion(Yadmellat et al., 2013). In general, the position of the manipulator is described by the tool frame, which is attached to the end-effectors, in relation to the base frame, which is attached to the manipulator's nonmoving base. Industrial robots are typically controlled by electricity. By installing an actuator at each joint of the robot, the manipulating system can be derived. The input control unit for the robotic manipulator system is the actuator. Servo and PMDC motors are the most often utilized electrical actuators in robotics (Ashi, 2014).

In this chapter 3-DOF industrial articulated robot arm are used and the kinematic and dynamic equation of motion for proposed robot arm are modeled. model predictive control is designed to follow the set point tracking.

With an electrical motor at each joint, the robotic arm may move freely in any degree of freedom. The robot properties are specified by the robot object parameter, and the robot joint can be placed on the robot physical system's base, wrist, shoulder, elbow, and slider. The total number of degrees of freedom of rigid bodies minus the number of constraints on their mobility equals the number of degrees of freedom of robots. Joints impose restrictions on movement. The revolute joint is the most frequent type of joint (Mahil & Al-durra, 2016).

Let say the robot has N links and j joint. By historical convention, N includes ground as a link and I defined m to be the degree of freedom of a single body.

We can write equation and determine number of degrees of freedom using Grumbler's formula in terms of these variable.

$N-1$: - is the number of total links other than ground &

$m(N-1)$: - is the total numbers of freedom of bodies (not constrained by joints). Then subtract off the constraints provided by the joints. Drive the equation

DOF= $\sum$(freedoms of bodies) - # of independent constraint

$$\text{DOF}= \underbrace{m(N-1)}_{\text{rigid body of freedoms}} - \underbrace{\sum_{i=1}^{j} c_i}_{\text{joint constraint}} \tag{3.1}$$

Since the number off constraints provided by joint i is equal to m minus the number of freedoms allowed by joint i. We can replace c_i by m minus an independent constraint f_i and rewrite equation the above (3.1).

$$\text{DOF}=m(N-1)-\sum_{i=1}^{j}(m-f_i) \tag{3.2}$$

After rearranging, the obtained Grumbler's formula is:

$$\mathrm{DOF}=m(N-1-j)+\sum_{i=1}^{j} f_i) \tag{3.3}$$

The above (3.3) formula assumes that the joints' constraints are all independent. For various robot mechanisms, we can use Grumbler's formula.

3.4.1 Position description

A 3x1 position vector can be used to locate any point in space, but it must be coupled to a coordinate system with three mutually orthogonal unit vectors in which it is defined by superscript. The position represents the vector's projection on each of the coordinate system's axes. Let us consider a point p represented on the three-dimensional Cartesian vector space p_x, p_y, and p_z in the fixed frame O (Incremona et al., 2019). A representation of vector $\vec{r}$ in fixed frame O of rigid body can be written as:

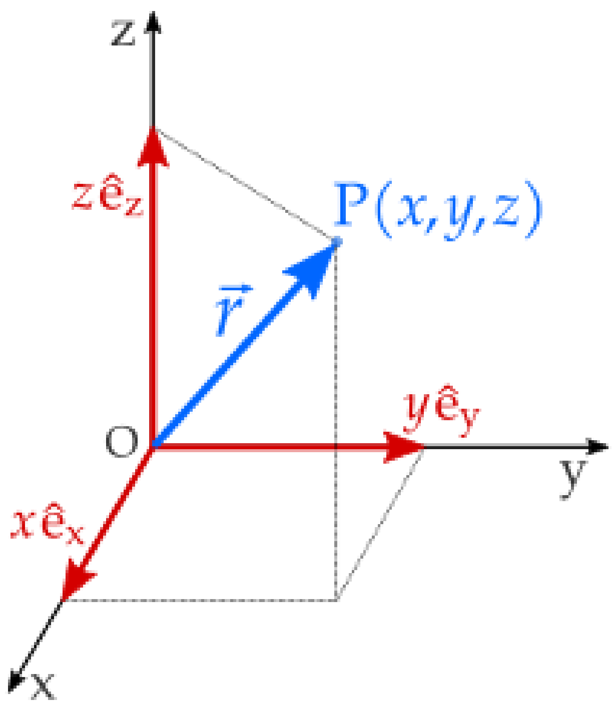


Figure 3.2 Position geometry radius vector (Incremona et al., 2019)

A Euclidean location or radius vector is a view position vector in geometry space. It represents the position of point P in space with respect to frame reference O. In short, a position vector is a translation that maps from original point to P.

$$\vec{r} = \overrightarrow{OP} \tag{3.4}$$

$$[p]_O = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \tag{3.5}$$

Radius vector $\vec{r}$ represents the position of a point $P(x, y, z)$ with respect to origin. In Cartesian coordinate system

$$\vec{r} = P_x\hat{e}_x + P_y\hat{e}_y + P_z\hat{e}_y \tag{3.6}$$

Where $\hat{e}_x,\ \hat{e}_y\ and\ \hat{e}_z$ represented the unit vector along three-dimension space of the frame O.

3.4.2 Orientation description

A vector describes the position of a point, whereas an attached coordinate system describes the orientation of a body. To represent and characterize the orientation of a body in Figure 3.2 example below, the coordinate system must connect the red line {B} to it and define it in relation to the reference coordinate system blue line {A}. Then, in terms of {A}, write 3-unit vectors of {B} and stack as columns of a 3 by 3 rotation matrix. in Figure 3.3, object orientation expressed with rotation matrix. The unit vectors of the element frame with respect to the reference system are the columns of the rotation matrix (Johansson, 1990).

$${}^{A}R_{B} = \begin{bmatrix} {}^{A}x'_B & {}^{A}y'_B & {}^{A}z'_B \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}_{3x3} \tag{3.7}$$

$$\begin{bmatrix} R_{11} \\ R_{21} \\ R_{31} \end{bmatrix} = \begin{bmatrix} x'_B.x_A \\ x'_B.y_A \\ x'_B.z_A \end{bmatrix}, \qquad \begin{bmatrix} R_{12} \\ R_{22} \\ R_{32} \end{bmatrix} = \begin{bmatrix} y'_B.x_A \\ y'_B.y_A \\ y'_B.z_A \end{bmatrix}, \qquad \begin{bmatrix} R_{13} \\ R_{23} \\ R_{33} \end{bmatrix} = \begin{bmatrix} z'_B.x_A \\ z'_B.y_A \\ z'_B.z_A \end{bmatrix} \tag{3.8}$$

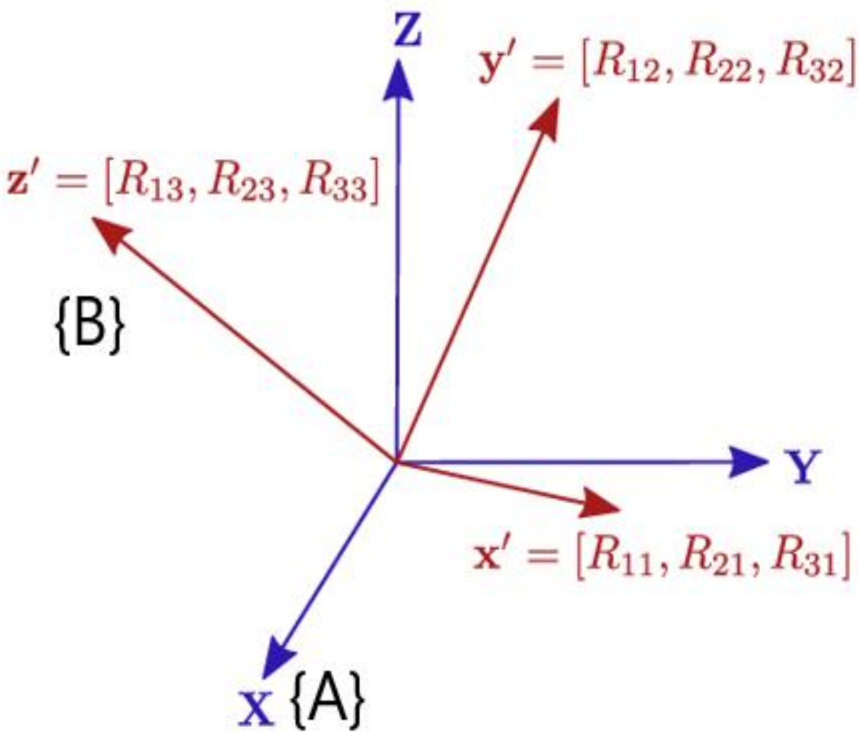


Figure 3.3 Object orientation expressed with rotation matrix (Chang et al., 2008)

$$^{A}R_{B} = \begin{bmatrix} ^{A}x'_{B} & ^{A}y'_{B} & ^{A}z'_{B} \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}_{3x3} \tag{3.9}$$

Cosine of the angle is the dot product of two-unit vectors. Therefore, new vector orientation object with respect to reference system {A} is

$$\begin{bmatrix} ^{A}x'_{B} \end{bmatrix} = \begin{bmatrix} R_{11} \\ R_{21} \\ R_{31} \end{bmatrix}, \qquad \begin{bmatrix} ^{A}y'_{B} \end{bmatrix} = \begin{bmatrix} R_{12} \\ R_{22} \\ R_{32} \end{bmatrix}, \qquad \begin{bmatrix} ^{A}z'_{B} \end{bmatrix} = \begin{bmatrix} R_{13} \\ R_{23} \\ R_{33} \end{bmatrix} \tag{3.10}$$

3.4.3 Rotation matrix

Representation and explanation of rotation matrix depend on the concept of projection. Here, suppose that XYZ plane frame as the Figure 3.4.

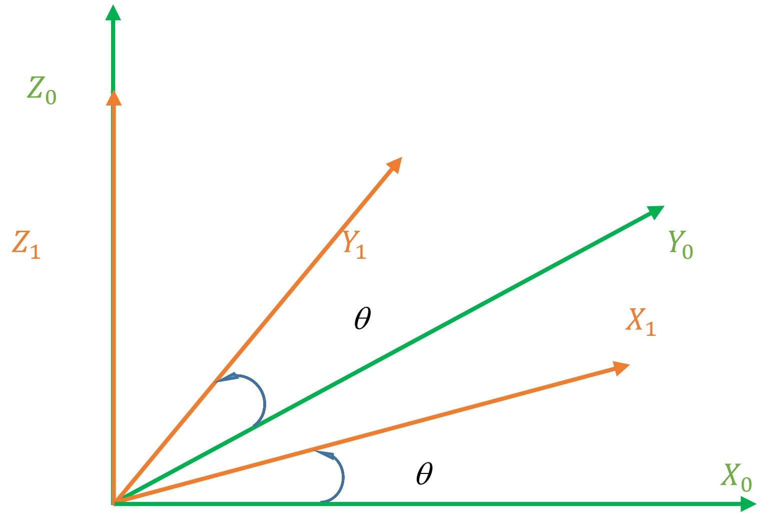


Figure 3.4 rotation matrix projection vector around Z-axis

The rotation of a matrix about any vector axis is the projection of a new vector on to an old vector. From the physical appearance of my system structure, the rotation of joint variable is around z-axis. Thus, a vector graph around the z-axis is chosen to obtain rotation matrix values using trigonometric values.

The projection of old vector of z on new vector of z is the same. Thus, the value is 1. But, for x and y, using trigonometric ratio, the adjacent and opposite sides are $\cos\theta$ and $\sin\theta$.only the direction of x old and y new is opposite means the projection of two vectors are negative.

$$R_z = \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.11}$$

Where, $C\theta$ is $\cos\theta$ and $S\theta$ is $\sin\theta$

Properties of rotation matrix

1. A rotation matrix will always be a square matrix and determinant is always equal to one.
2. Because of rotation matrix is always an orthogonal matrix, the inverse of the matrix is the transpose.

3.5 Robot kinematics

Kinematics describes body of motion without taking into account input control unit forces or any motion-causing moments. By making one variable independent and vice versa, robot kinematics analyzes the relationship between its spatial layout, robot coordinate joint orientation, position, acceleration, and velocity of each link of body in a robotic manipulator system. In general robot kinematics is the numerical and analytical study of robot arm manipulators (L1, 1988). In analytical kinematics, exact solution is obtained by creating a well-understood technique of the assignment frame issue, whereas numerical kinematics relies on guess work, until sufficient results are obtained. As a result, for a 3-DOF articulated robotic manipulator system, the analytical method approach is chosen. The Cartesian coordinate system of a robot can be separated into two types of movement: translational and rotational. Denavit-Hartenberg parameters are used to describe robotic kinematics, which can be represented by a homogeneous square matrix transformation (Kim et al., 2020). Forward and backward kinematics are the two types of kinematics in robotics. Forward kinematics is the technique of determining end effector position given a link's joint angle. It is a straightforward method of determining the link in advance employing an analytical method investigation. Inverse kinematics is the technique of determining the angle joint of a robot given the position vector of the end effector, and it is the inverse of forward kinematics. When kinematic equations have several solutions, inverse kinematics is difficult to calculate and costs time rather than forward motion. For the analytical method of robot kinematics, there are two basic solution techniques. Geometrical and algebraic equations of solution are the two methods. A geometric model for forward and algebraic for inverse kinematics of a robotic manipulator system is used to solve this challenge(Incremona et al., 2017). The graphic Figure 3.5 depicts a schematic illustration of forward and inverse kinematics.

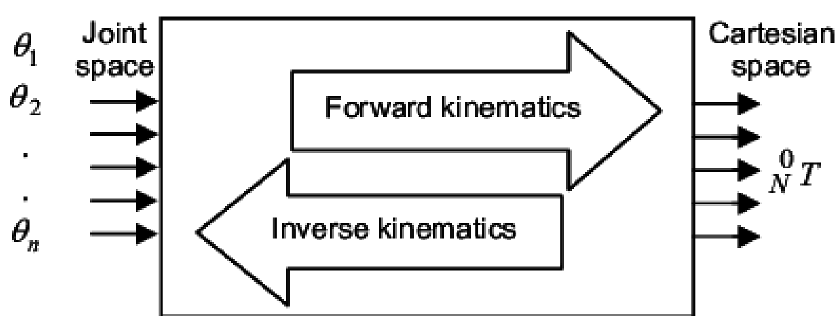


Figure 3.5 Schematic presentations of forward and inverse kinematics (K. Kawamura, 2007)

3.5.1 Forward kinematics

Forward kinematics is the technique of applying a robot's kinematic equation to compute the end effector position in Cartesian space given the joint angle of the link. Furthermore, forward kinematics is the technique of calculating end effector velocity given known joint angles and angular velocities (Fateh, 2008). The purpose is to obtain a Cartesian coordinates position of space, if the first and second joint angles for an arm are given in Cartesian coordinate plane.

Forward kinematics is a method of computing the relationship between location, velocity, acceleration, and orientation end-effector frame and known joint variables using a transformation homogeneous matrix. The forward and inverse kinematics of the three DOF articulated robotic arm analysis problem is solved in this study.

Forward kinematics is the process of calculating the position and orientation of an end effector in terms of a joint variable. Peter Cork using Denavit-Hartenberg parameters is the most frequent way for assessing and describing the robot forward kinematics in this study. In MATLAB peter-corker, a graphical series connection of a 3-DOF articulated robotic manipulator with forward kinematics is presented in this section (K. Kawamura, 2007).

These are the parameters: α is the twist angle: this is the angle required to map the previous joint's z-axis to the current joint's z-axis, θ is the rotation angle: this is always along the z-axis, a is the distance between previous and next joint centers in x-direction, and d is the offset distance between previous and next joint centers in z-direction. Figure 3.1 shows the frame assignments for the 3-DOF articulated robotic arm, as well as the numerous joints and their linkages (Incremona et al., 2017). Table 3.1 shows the DH parameters for the 3-DOF robotic manipulator.

Table 3.1 DH parameter for 3-DOF articulated robotic arm

Link	a_{i-1}	α_{i-1}	d_i	θ_i
1	0	90^0	d_1	θ_1
2	a_2	0	0	θ_2
3	a_3	0	0	θ_3

But here, rotational matrix and displacement column matrix vector are used using transformation homogenous matrix for describing a forward kinematics. Therefore, Homogenous transformation matrix is obtained based on number of degrees of freedom. It is expressed in terms of rotation matrix, displacement vector from old to new one. The frame assignments for 3-DOF articulated robotic manipulator is shown in Figure 3.6 below.

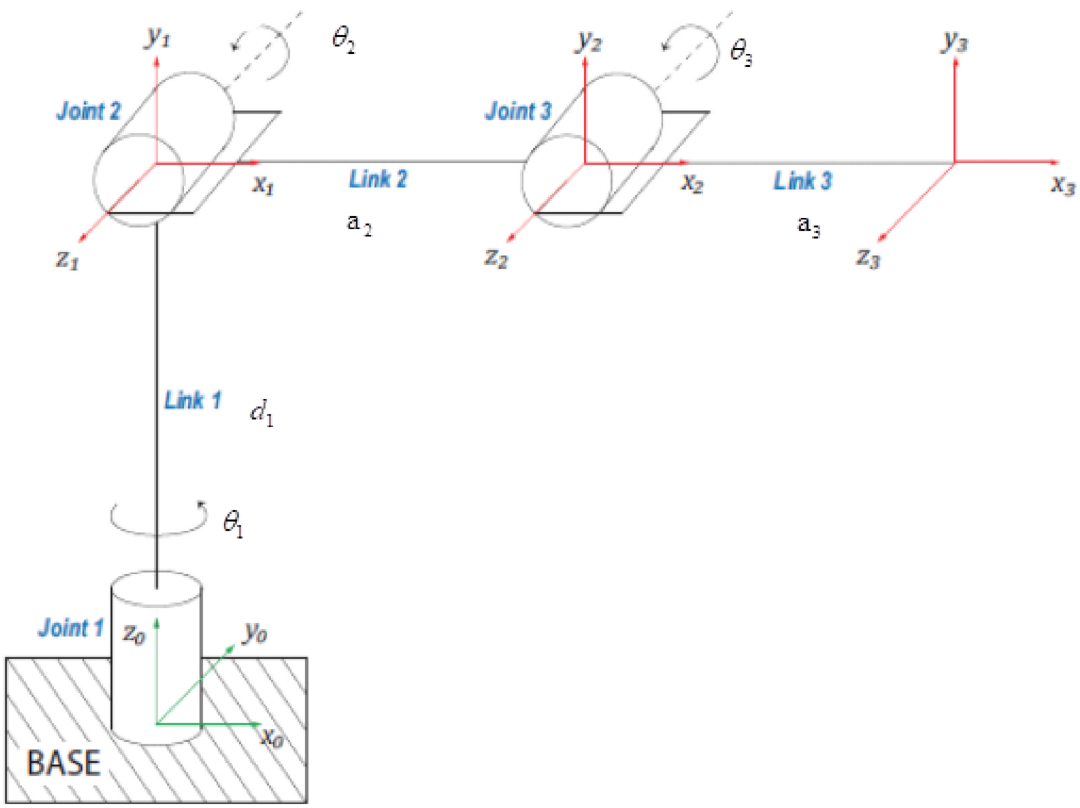


Figure 3.6 Frame assignments for the 3-DOF articulated robotic arm (Ashagrie et al., 2021)

From the Figure 3.6, the angle of rotation is around z-axis, therefore here, rotation matrix around z is used. Homogenous transformation (HT) matrix from frame 0 to frame 1 is:

$$\mathrm{H}_1^0 = \begin{bmatrix} R_1^0 & & & d_1^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.12}$$

R_1^0 and d_1^0 are rotation matrix and displacement vector from frame 0 to 1 respectively. R_1^0 is equal to the multiplication of rotation matrix around z and the projection matrix of new x, y and z vector on old Cartesian vector from frame 0 to 1. If the rigid body move without rotation, then the projection matrix new on old one is identity. Rotation around z-axis is:

$$R_Z = \begin{bmatrix} C_1 & -S_1 & 0 \\ S_1 & C_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.13}$$

The projection matrix of new vector on old vector from frame 0 to 1 is:

$$P_1^0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \tag{3.14}$$

$$R_1^0 = R_Z * P_1^0$$

$$R_1^0 = \begin{bmatrix} C_1 & 0 & S_1 \\ S_1 & 0 & -C_1 \\ 0 & 1 & 0 \end{bmatrix} \tag{3.15}$$

The displacement vector from frame 0 to frame 1 is:

$$d_1^0 = \begin{bmatrix} 0 \\ 0 \\ d_1 \end{bmatrix} \tag{3.16}$$

The homogenous transformation matrix from frame 0 to frame 1 (H_1^0) is:

$$\mathrm{H}_1^0 = \begin{bmatrix} R_1^0 & & & d_1^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & d1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.17}$$

From frame 1 to frame 2, a homogeneous transformation matrix is computed and displayed. The axis of rotation of the robotic manipulator's second link is identical to that of the robot's first link. From frame 1 to 2, the projection vector matrix is changed and homogenous transformation matrix is:

$$\mathrm{H}_2^1 = \begin{bmatrix} R_2^1 & & & d_2^1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.18}$$

The projection matrix of new vector on old vector from frame 1 to 2 is:

$$P_2^1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.19}$$

$$R_2^1 = R_Z * P_2^1$$

$$R_2^1 = \begin{bmatrix} C_2 & -S_2 & 0 \\ S_2 & C_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.20}$$

The displacement vector from frame 1 to frame 2 is:

$$d_2^1 = \begin{bmatrix} \mathrm{a}_2 C_2 \\ \mathrm{a}_2 S_2 \\ 0 \end{bmatrix} \tag{3.21}$$

The homogenous transformation matrix from frame 1 to frame 2 (H_2^1) is:

$$\mathrm{H}_2^1 = \begin{bmatrix} R_2^1 & & & d_2^1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_2 & -S_2 & 0 & \mathrm{a}_2 C_2 \\ S_2 & C_2 & 0 & \mathrm{a}_2 S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.22}$$

Homogenous transformation matrix from frame 2 to frame 3 is computed and represented. In the third link of robotic manipulator the axis of rotation is the same as the first and second link of the robot and projection vector direction matrix from frame 2 to 3 is the same with second joint of robot. Homogenous transformation matrix from frame 2 to frame 3 is:

$$\mathrm{H}_3^2 = \begin{bmatrix} R_3^2 & & & d_3^2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.23}$$

The projection direction matrix of new vector on old vector from frame 2 to 3 is:

$$P_3^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.24}$$

$$R_3^2 = R_Z * P_3^2$$

$$R_3^2 = \begin{bmatrix} C_3 & -S_3 & 0 \\ S_3 & C_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.25}$$

The displacement vector from frame 2 to frame 3 is:

$$d_3^2 = \begin{bmatrix} \mathrm{a}_3 C_3 \\ \mathrm{a}_3 S_3 \\ 0 \end{bmatrix} \tag{3.26}$$

The homogenous transformation matrix from frame 2 to frame 3 (H_3^2) is:

$$\mathrm{H}_3^2 = \begin{bmatrix} R_3^2 & & & d_3^2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_3 & -S_3 & 0 & \mathrm{a}_3 C_3 \\ S_3 & C_3 & 0 & \mathrm{a}_3 S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.27}$$

Therefore, a general homogenous transformation matrix from frame 0 to frame 3 is presented. It is written in compact form.

$$\mathrm{H}_3^0 = \mathrm{H}_1^0 \mathrm{H}_2^1 \mathrm{H}_3^2 \tag{3.28}$$

$$\mathrm{H}_3^0 = \begin{bmatrix} C_1 C_{23} & -C_1 S_{23} & S_1 & \mathrm{a}_3 C_1 C_{23} + \mathrm{a}_2 C_1 C_2 \\ S_1 C_{23} & -S_1 S_{23} & -C_1 & \mathrm{a}_3 S_1 C_{23} + \mathrm{a}_2 S_1 C_2 \\ S_{23} & C_{23} & 0 & \mathrm{a}_3 S_{23} + \mathrm{a}_2 S_2 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.29}$$

From the formula of homogenous transformation matrix

$$\mathrm{H}_3^0 = \begin{bmatrix} R_3^0 & & & d_3^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.30}$$

d_3^0: -is the displacement vector from frame 0 to frame 3, which is the forward kinematic end effector Cartesian space position of robotic arm manipulator.

$$d_3^0 = \begin{bmatrix} \mathrm{a}_3 C_1 C_{23} + \mathrm{a}_2 C_1 C_2 \\ \mathrm{a}_3 S_1 C_{23} + \mathrm{a}_2 S_1 C_2 \\ \mathrm{a}_3 S_{23} + \mathrm{a}_2 S_2 + d_1 \end{bmatrix} \tag{3.31}$$

The end effector coordinates are:

$$\begin{aligned} x_3 &= \mathrm{a}_3 C_1 C_{23} + \mathrm{a}_2 C_1 C_2 \\ y_3 &= \mathrm{a}_3 S_1 C_{23} + \mathrm{a}_2 S_1 C_2 \\ z_3 &= \mathrm{a}_3 S_{23} + \mathrm{a}_2 S_2 + d_1 \end{aligned} \tag{3.32}$$

The end-effector orientation matrix is represented by the first three rows and columns of the transformation matrix:

$$R_3^0 = \begin{bmatrix} C_1 C_{23} & -C_1 S_{23} & S_1 \\ S_1 C_{23} & -S_1 S_{23} & -C_1 \\ S_{23} & C_{23} & 0 \end{bmatrix} \tag{3.33}$$

In order to get the end effector robot linear velocities articulation, derivation is applied on (3.32) with respect to time.

$$\begin{cases} \dot{x}_2 = -\mathrm{a}_2 C_1 S_2 \dot{\theta}_2 - \mathrm{a}_3 C_1 S_{23} (\dot{\theta}_2 + \dot{\theta}_3) - (\mathrm{a}_2 S_1 C_2 + \mathrm{a}_3 S_1 C_{23}) \dot{\theta}_1 \\ \dot{y}_2 = -\mathrm{a}_2 S_1 S_2 \dot{\theta}_2 - \mathrm{a}_2 S_1 S_{23} (\dot{\theta}_2 + \dot{\theta}_3) + (\mathrm{a}_2 C_1 C_2 + \mathrm{a}_3 C_1 C_{23}) \dot{\theta}_1 \\ \dot{z}_2 = \mathrm{a}_2 C_2 \dot{\theta}_2 + \mathrm{a}_3 C_{23} (\dot{\theta}_2 + \dot{\theta}_3) \end{cases} \tag{3.34}$$

In Figure 3.7, the 3-DOF articulated robotic manipulator at starting time. The angular position for this physical system is at 000 rad. The initial and final end effector coordinate for this physical system is obtained using forward petercorke method. The rotation of the angle is around z-axis and the others are expressed in terms articulation of it.

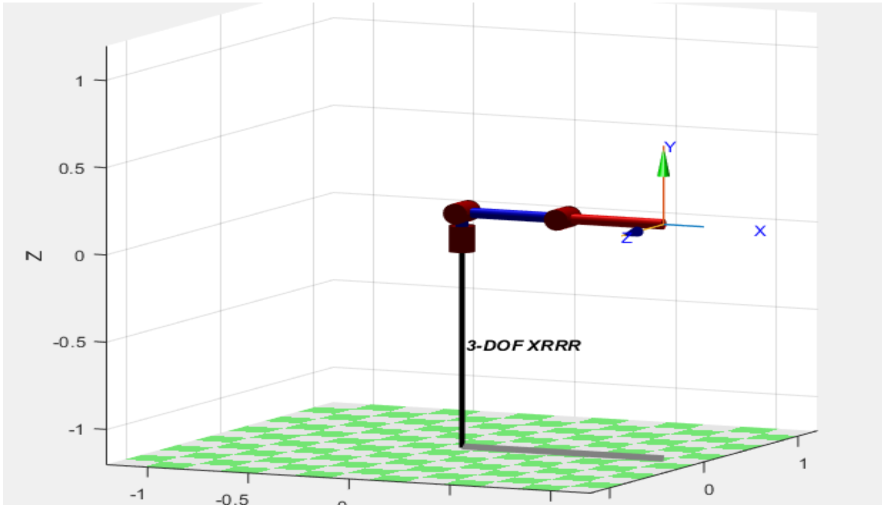


Figure 3.7-DOF articulated robotic manipulator at angular position of 000

3.5.2 Inverse kinematics

Inverse kinematics refers to the process of finding angle joint of robot given known positon vector of end effector Cartesian coordinate space. Algebraic approach top and side view of frame assignments of 3-DOF articulated robotic manipulator is used for solving inverse kinematics problem. In general, it is a mathematical process used to compute the joint position at specific position and orientation that are needed to put robotic end effector at the desired place(Darajat & Istiqphara, 2020).

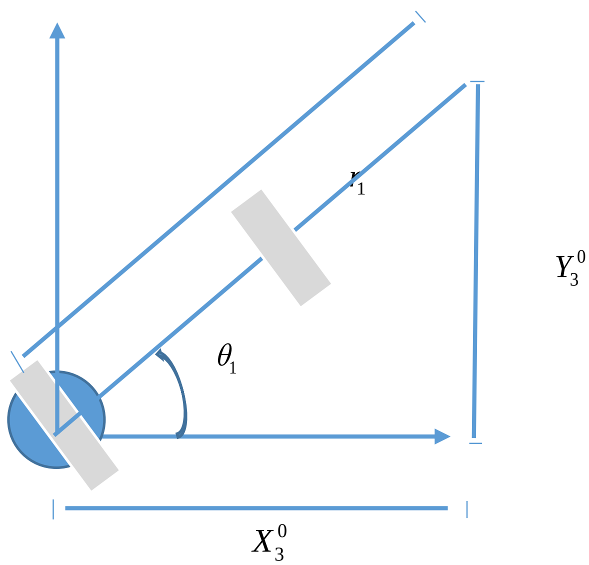


Figure 3.8 Frame assignment 3-DOF robot from top view

The solving problem in inverse kinematics is more difficult than the forward kinematics problem. Here, it is desired to find the angles θ_1, θ_2 and θ_3 corresponding to the given end effector positions and orientations for the articulated 3-DOF robotic arm motion. The Figure 3.8 is illustrated, when we see the frame assignment from the top view

$$\theta_1 = \tan^{-1}\left(\frac{Y_3^0}{X_3^0}\right) \tag{3.35}$$

$$r_1 = \sqrt{\left(X_3^0\right)^2 + \left(Y_3^0\right)^2} \tag{3.36}$$

When we see the frame assignment representation of 3-DOF articulated robotic manipulator from side view, the representation of the Figure 3.9 is obtained. When θ_1 and θ_2 are not at position zero.

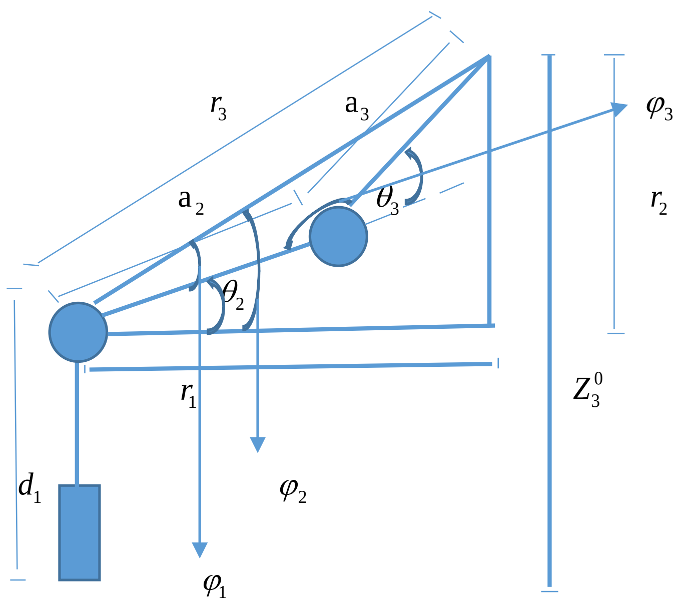


Figure 3.9 frame assignment representation of 3-DOF articulated robotic manipulator from side view

From the above figure

$$\theta_2 = \varphi_2 - \varphi_1 \tag{3.37}$$

$$\varphi_2 = \tan^{-1}\left(\frac{r_2}{r_1}\right), \text{ where } r_2 = Z_3^0 - d_1, \text{ and } r_1 = \sqrt{\left(X_3^0\right)^2 + \left(Y_3^0\right)^2} \tag{3.38}$$

In trigonometric, for not right angle triangle the equation becomes:

$$a_3^2 = a_2^2 + r_3^2 - 2a_2 r_3 C\varphi_1 \tag{3.39}$$

$$\varphi_1 = \cos^{-1}\left(\frac{a_3^2 - a_2^2 - r_3^2}{-2a_2 r_3}\right), \text{ where } r_3 = \sqrt{r_1^2 + r_2^2} \tag{3.40}$$

$$\theta_3 = pi - \varphi_3 \tag{3.41}$$

$$r_3^2 = a_2^2 + a_3^2 - 2a_2 a_3 C\varphi_3 \tag{3.42}$$

$$\varphi_3 = \cos^{-1}\left(\frac{r_3^2 - a_2^2 - a_3^2}{-2a_2 a_3}\right) \tag{3.43}$$

Assumptions that take for making dynamic model of 3-DOF articulated robotic manipulator

1. The center mass of an object is considered at the center of each link of robotic manipulator
2. Initially the radian for each joint are at 000 rad and gravity used 9.81 m/s^2
3. At the first link of the robot potential energy is zero
4. Based on torque calculation from the specification of robot manipulator the constraint is 40 Nm
5. The moment of inertia is symmetric and positive defined matrix

3.6 Dynamic modelling of 3-DOF articulated robot

Dynamics of robot refers to the studies of describing and analyzing motion of a body with consideration of input force that cause to motion. Dynamics of robot related to position, velocity, acceleration, inertia and mass of a body by applying input control unit force and torque for both translational and rotational body respectively. Robot dynamics is mainly concerned on the relationship between the input control force acting on robot mechanism and the acceleration produced by force. It is necessary to have actuator capable of exerting enough force in order to accelerate the robot link, joint move to the desired position and angular velocity. In this research, PMDC motor is used to give input control torque for 3-DOF articulated robotic manipulator but not consider its actuator dynamics. The dynamical performance of robot manipulators is highly affected by varying load and parameter change of link of rigid body. In general, dynamics of robot in this research considers input torque unit for controlling the acceleration, velocity and position of joint and link of robot manipulator system. Therefore, the mathematical equations that describe the relationship between the joint torques and articulation motion of the robot is derived (Agbaraji et al., 2017).

By using the Euler lagrangian method, the mathematical dynamic equation of motion for the three degree of freedom articulated robotic arm by considering that the center of mass for each link is at the center of link. For the three degree of freedom robotic arm link 1, 2 and 3, The joint variables are $\theta_1,\ \theta_2$ and θ_3, link lengths are $d_1,\ a_2,$ and a_3. The mass of the link for three degrees of freedom are $m_1,\ m_2,$ and m_3 respectively. In this study, the datum potential energy is chosen at the axis of rotation or origin.

The control joint input torque vector is:

$\tau = [\tau_1\ \ \tau_2\ \ \tau_3]^{T}$ where, $\tau_1,\ \tau_2,$ and τ_3 are the joint torques supplied by the motor in each joint for each link respectively.

$\theta = [\theta_1\ \ \theta_2\ \ \theta_3]^{T}$ where $\theta_1,\ \theta_2$ and θ_3, are joint variable angle for each joint respectively. The values of the physical parameters of industrial 3-DOF articulated robotic manipulator are presented in Table 3.2 (Pezeshki et al., 2012).

Table 3.2 List of parameters of robot manipulator

Parameter	**Notation**	**Value**
Length of link 1	d_1	0.15m
Length of link 2	a_2	0.5m
Length of link 3	a_3	0.5m
Mass of link 1	m_1	0.5kg
Mass of link 2	m_2	0.5kg
Mass of link 3	m_3	0.5kg
Gravity	G	$9.81 m/s^2$

3.6.1 Kinetic and Potential Energy

The kinetic and potential energy of three degree of freedom industrial articulated robotic arm for each link of robot by using physics law is derived. Kinetic energy is expressed on both joint variable and angular joint velocity, but potential energy is only in joint variable. In this thesis, the center of mass of an object is considered at the center of each link of robotic manipulator(Dereje, 2018).

For link 1 the kinetic is only rotational kinetic energy. Therefore, the kinetic and potential energy for link 1 are expressed as:

$$K_1 = I_1 \frac{\dot{\theta}_1^2}{2} \qquad \text{and} \qquad U_1 = 0 \tag{3.44}$$

For dynamic part, where

$C_i = \cos\theta_i$, i=1,2,3 $\quad \mathrm{S}_i = \sin\theta_i, i = 1, 2, 3$

$C_{ij} = \cos(i+j),\ \mathrm{S}_{ij} = \sin(i+j),\ \mathrm{C}_{2i+j} = \cos(2i+j),\ \mathrm{S}_{2i+j} = \sin(2i+j)$

$\mathrm{C}_{2(i+j)} = \cos(2(i+j)),\ \mathrm{S}_{2(i+j)} = \sin(2(i+j))$

By Multiplying (3.17) and (3.22), the homogenous transformation from frame 0 to frame 2 is obtained.

$$H_2^0 = H_1^0 H_2^1 \tag{3.45}$$

$$H_2^0 = \begin{bmatrix} C_1C_2 & -C_1S_2 & S_1 & \mathrm{a}_2C_1C_2 \\ S_1C_2 & -S_1S_2 & -C_1 & \mathrm{a}_2S_1C_2 \\ S_2 & C_2 & 0 & \mathrm{a}_2S_1 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_2^0 & & & d_2^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.46}$$

From the above the displacement vector from frame 0 to frame 2 is the positions of link 2 end effector Cartesian space

$$d_2^0 = \begin{bmatrix} \mathrm{a}_2C_1C_2 \\ \mathrm{a}_2S_1C_2 \\ \mathrm{a}_2S_2 + d_1 \end{bmatrix} \tag{3.47}$$

For link 2, the position is equal to (3.47) and velocity are obtained

$$\begin{aligned} x_2 &= \mathrm{a}_2 C_1 C_2 \\ y_2 &= \mathrm{a}_2 S_1 C_2 \\ z_2 &= \mathrm{a}_2 S_2 + d_1 \end{aligned} \tag{3.48}$$

$$\dot{x}_2 = -\mathrm{a}_2 S_1 C_2 \dot{\theta}_1 - \mathrm{a}_2 C_1 S_2 \dot{\theta}_2 \tag{3.49}$$

$$\dot{y}_2 = \mathrm{a}_2 C_1 C_2 \dot{\theta}_1 - \mathrm{a}_2 S_1 S_2 \dot{\theta}_2 \tag{3.50}$$

$$\dot{z}_2 = \mathrm{a}_2 C_2 \dot{\theta}_2 \tag{3.51}$$

Therefore, velocity squared in kinetic energy is:

$$v_2^2 = \dot{x}_2^2 + \dot{y}_2^2 + \dot{z}_2^2 \tag{3.52}$$

Then the kinetic and potential energy for the second mass is:

$$K_2 = \frac{1}{2} m_2 \left(\mathrm{a}_2^2 C_2^2 \dot{\theta}_1^2 + \mathrm{a}_2 \dot{\theta}_2^2 \right) \tag{3.53}$$

$$U_2 = m_2 \mathrm{g} \mathrm{a}_2 S_2 \tag{3.54}$$

For link 3 the position and velocity are obtained

$$x_3 = \mathrm{a}_3 C_1 C_{23} + \mathrm{a}_2 C_1 C_2 \tag{3.55}$$

$$y_3 = \mathrm{a}_3 S_1 C_{23} + \mathrm{a}_2 S_1 C_2 \tag{3.56}$$

$$z_3 = \mathrm{a}_3 S_{23} + \mathrm{a}_2 S_2 + d_1 \tag{3.57}$$

$$\begin{cases} \dot{x}_3 = [-\mathrm{a}_2 S_2 \dot{\theta}_2 - \mathrm{a}_3 S_{23} (\dot{\theta}_2 + \dot{\theta}_3)] C_1 - (\mathrm{a}_2 C_2 + \mathrm{a}_3 C_{23}) S_1 \dot{\theta}_1 \\ \dot{y}_3 = [-\mathrm{a}_2 S_2 \dot{\theta}_2 - \mathrm{a}_3 S_{23} (\dot{\theta}_2 + \dot{\theta}_3)] S_1 + (\mathrm{a}_2 C_2 + \mathrm{a}_3 C_{23}) C_1 \dot{\theta}_1 \\ \dot{z}_3 = \mathrm{a}_2 C_2 \dot{\theta}_2 - [\mathrm{a}_3 (\dot{\theta}_2 + \dot{\theta}_3)] C_{23} \end{cases} \tag{3.58}$$

Therefore, velocity squared in kinetic energy is:

$$v_3^2 = \dot{x}_3^2 + \dot{y}_3^2 + \dot{z}_3^2 \tag{3.59}$$

Then the kinetic and potential energy for the third mass is:

$$K_3 = \frac{1}{2} m_3 \Big(a_2 \dot{\theta}_2^2 + 2a_2 a_3 C_2 C_{23} \dot{\theta}_2^2 + 2a_2 a_3 S_2 S_{23} \dot{\theta}_2^2 + 2a_2 a_3 C_2 C_{23} \dot{\theta}_2 \dot{\theta}_3 + 2a_2 a_3 S_2 S_{23} \dot{\theta}_2 \dot{\theta}_3 + \ldots \\ 2a_2 a_3 C_2 C_{23} \dot{\theta}_1^2 + a_3^2 \dot{\theta}_2^2 + a_3^2 \dot{\theta}_3^2 + 2a_3^2 \dot{\theta}_2 \dot{\theta}_3 + a_2^2 C_2^2 \dot{\theta}_1^2 + a_3^2 C_{23}^2 \dot{\theta}_1^2 \Big) \tag{3.60}$$

$$U_3 = m_3 g (a_2 S_2 + a_3 S_{23}) \tag{3.61}$$

3.6.2 Euler Lagrangian Equation

The general lagrangian equation formula is expressed by the following formula

$$L = K - U \tag{3.62}$$

Before going to lagrangian formulation, firstly the total kinetic energy of three degree of freedom robot arm is calculated as

$$K = K_1 + K_2 + K_3 \tag{3.63}$$

$$K = I_1 \frac{\dot{\theta}_1^2}{2} + \frac{1}{2} m_2 \Big(a_2^2 C_2^2 \dot{\theta}_1^2 + a_2 \dot{\theta}_2^2 \Big) + \frac{1}{2} m_3 \Big(a_2 \dot{\theta}_2^2 + 2a_2 a_3 C_2 C_{23} \dot{\theta}_2^2 + 2a_2 a_3 S_2 S_{23} \dot{\theta}_2^2 + \ldots \\ 2a_2 a_3 C_2 C_{23} \dot{\theta}_2 \dot{\theta}_3 + 2a_2 a_3 S_2 S_{23} \dot{\theta}_2 \dot{\theta}_3 + 2a_2 a_3 C_2 C_{23} \dot{\theta}_1^2 + a_3^2 \dot{\theta}_2^2 + a_3^2 \dot{\theta}_3^2 + 2a_3^2 \dot{\theta}_2 \dot{\theta}_3 \ldots \\ + a_2^2 C_2^2 \dot{\theta}_1^2 + a_3^2 C_{23}^2 \dot{\theta}_1^2 \Big) \tag{3.64}$$

And the total potential energy of three link robot arm can be calculated as

$$U = U_1 + U_2 + U_3 \tag{3.65}$$

$$U = 0 + m_2 g a_2 S_2 + m_3 g (a_2 S_2 + a_3 S_{23}) \tag{3.66}$$

Therefore, by using (3.62), the Euler-lagrangian equation of the system is derived.

$$L = K - U$$

$$L = I_1 \frac{\dot{\theta}_1^2}{2} + \frac{1}{2} m_2 \Big(a_2^2 C_2^2 \dot{\theta}_1^2 + a_2 \dot{\theta}_2^2 \Big) + \frac{1}{2} m_3 \Big(a_2 \dot{\theta}_2^2 + 2a_2 a_3 C_2 C_{23} \dot{\theta}_2^2 + 2a_2 a_3 S_2 S_{23} \dot{\theta}_2^2 + \ldots \\ 2a_2 a_3 C_2 C_{23} \dot{\theta}_2 \dot{\theta}_3 + 2a_2 a_3 S_2 S_{23} \dot{\theta}_2 \dot{\theta}_3 + 2a_2 a_3 C_2 C_{23} \dot{\theta}_1^2 + a_3^2 \dot{\theta}_2^2 + a_3^2 \dot{\theta}_3^2 + 2a_3^2 \dot{\theta}_2 \dot{\theta}_3 + \ldots \\ a_2^2 C_2^2 \dot{\theta}_1^2 + a_3^2 C_{23}^2 \dot{\theta}_1^2 \Big) - m_2 g a_2 S_2 - m_3 g (a_2 S_2 + a_3 S_{23}) \tag{3.67}$$

For any input torque, the general formula of equation motion of a system is

$$\tau_i = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \frac{\partial L}{\partial \theta_i} \tag{3.68}$$

Therefore, the first equation of motion is

$$\tau_1 = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_1}\right) - \frac{\partial L}{\partial \theta_1} \tag{3.69}$$

$$\begin{aligned}
\tau_1 = {} & \left(m_1 r_1^2 + \frac{1}{4} m_2 \mathrm{a}_2 C_2^2 + \frac{1}{4} m_3 \mathrm{a}_3 C_{23}^2 + m_3 \mathrm{a}_2^2 C_2^2 + \frac{1}{2} m_2 \mathrm{a}_2 \mathrm{a}_3 C_2 C_{23} \right) \ddot{\theta}_1 + \ldots \\
& \left(\frac{-1}{4} m_2 \mathrm{a}_2^2 S_{22} - m_3 \mathrm{a}_2^2 S_{22} - m_3 \mathrm{a}_2 \mathrm{a}_3 S_{22+3} - \frac{1}{4} m_3 \mathrm{a}_2^2 S_{2(2+3)} \right) \dot{\theta}_1 \dot{\theta}_2 + \ldots \\
& \left(-m_3 \mathrm{a}_2 \mathrm{a}_3 C_2 S_{23} - \frac{1}{4} m_3 \mathrm{a}_3^2 S_{2(2+3)} \right) \dot{\theta}_1 \dot{\theta}_3
\end{aligned} \tag{3.70}$$

$$\begin{aligned}
\tau_2 = {} & \left(\frac{1}{4} m_2 \mathrm{a}_2^2 + m_3 \left(\mathrm{a}_2^2 + \frac{1}{4} \mathrm{a}_3^2 + \mathrm{a}_2 \mathrm{a}_3 C_3 \right) \right) \ddot{\theta}_2 + \ldots \\
& m_3 \left(\frac{1}{2} \mathrm{a}_2 \mathrm{a}_3 C_3 + \frac{1}{4} \mathrm{a}_3^2 \right) \ddot{\theta}_3 - m_2 \mathrm{a}_2 \mathrm{a}_3 S_3 \dot{\theta}_2 \dot{\theta}_3 - \frac{1}{2} m_3 \mathrm{a}_2 \mathrm{a}_3 S_3 \dot{\theta}_3 \ldots \\
& + \left(\frac{1}{8} m_2 \mathrm{a}_2^2 S_{22} + \frac{1}{2} m_3 \mathrm{a}_2^2 S_{22} + \frac{1}{2} \mathrm{a}_2 \mathrm{a}_3 m_3 S_{22+3} + \frac{1}{8} \mathrm{a}_3^2 S_{2(2+3)} \right) \dot{\theta}_1^2 \ldots \\
& + \frac{1}{2} \mathrm{a}_2 m_2 g C_2 + \mathrm{a}_2 m_3 g C_2 + \frac{1}{2} \mathrm{a}_3 m_2 g C_{23}
\end{aligned} \tag{3.71}$$

$$\begin{aligned}
\tau_3 = {} & -m_3 \left(\frac{1}{2} \mathrm{a}_2 \mathrm{a}_3 C_3 + \frac{1}{4} \mathrm{a}_3^2 \right) \ddot{\theta}_2 + \frac{1}{3} m_3 \mathrm{a}_3^2 \ddot{\theta}_3 + \frac{1}{2} m_3 \mathrm{a}_3 S_{23} \left(\mathrm{a}_2 C_2 + \frac{1}{2} \mathrm{a}_3 C_{23} \right) \dot{\theta}_2^2 + \ldots \\
& m_3 \left(\frac{1}{2} \mathrm{a}_2 \mathrm{a}_3 S_3 \dot{\theta}_2 \dot{\theta}_3 + \frac{1}{2} g \mathrm{a}_3 C_{23} \right)
\end{aligned} \tag{3.72}$$

From above derived equation, the vector form yields dynamic equation for three degree of freedom industrial articulated robotic manipulators arm is given by three coupled nonlinear differential equation of motion. Robotic manipulator with n rigid degree of freedom is characterized by a set of n generalized coordinates $\theta = [\theta_1 \ \theta_2 \ \theta_3 \ldots \theta_n]^{\mathrm{T}}$ called the joint space (K. Kawamura, 2007). By presenting the above three equations, (3.73) is obtained.

$$\tau = M(\theta)\ddot{\theta} + B(\theta)[\dot{\theta}, \dot{\theta}] + C(\theta)[\dot{\theta}]^2 + G(\theta) \tag{3.73}$$

Where, $\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix}$ is the actuator joint torques vector. Where,

$$M(\theta) = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \quad \text{is moment of inertia matrix}$$

$$\text{Where,} \begin{cases} m_{11} = m_1 r_1^2 + \frac{1}{4} m_2 \mathrm{a}_2 C_2^2 + \frac{1}{4} m_3 \mathrm{a}_3 C_{23}^2 + m_3 \mathrm{a}_2^2 C_2^2 + \frac{1}{2} m_2 \mathrm{a}_2 \mathrm{a}_3 C_2 C_{23} \\ m_{12} = m_{13} = m_{21} = m_{31} = 0 \\ m_{22} = \frac{1}{4} m_2 \mathrm{a}_2^2 + m_3 \left(\mathrm{a}_2^2 + \frac{1}{4} \mathrm{a}_3^2 + \mathrm{a}_2 \mathrm{a}_3 C_3 \right) \\ m_{23} = m_3 \left(\frac{1}{2} \mathrm{a}_2 \mathrm{a}_3 C_3 + \frac{1}{4} \mathrm{a}_3^2 \right) \\ m_{32} = -m_3 \left(\frac{1}{2} \mathrm{a}_2 \mathrm{a}_3 C_3 + \frac{1}{4} \mathrm{a}_3^2 \right) \\ m_{33} = \frac{1}{3} m_3 \mathrm{a}_3^2 \end{cases}$$

The inertia matrix $M(\theta)$ is nxn symmetric and positive definite, $M(\theta)^{\mathrm{T}} = M(\theta) > 0,$ where,

$$B(\theta) = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \quad \text{is Coriolis force.}$$

$$\text{Where,} \begin{cases} b_{11} = \frac{-1}{4} m_2 \mathrm{a}_2^2 S_{22} - m_3 \mathrm{a}_2^2 S_{22} - m_3 \mathrm{a}_2 \mathrm{a}_3 S_{22+3} - \frac{1}{4} m_3 \mathrm{a}_2^2 S_{2(2+3)} \\ b_{12} = -m_3 \mathrm{a}_2 \mathrm{a}_3 C_2 S_{23} - \frac{1}{4} m_3 \mathrm{a}_3^2 S_{2(2+3)} \\ b_{13} = b_{21} = b_{22} = b_{31} = b_{32} = 0 \\ b_{23} = -m_2 \mathrm{a}_2 \mathrm{a}_3 S_3 \\ b_{33} = \frac{1}{2} \mathrm{m}_3 \mathrm{a}_2 \mathrm{a}_3 S_3 \dot{\theta}_2 \dot{\theta}_3 \end{cases}$$

$B(\theta)$ and $\mathrm{C}(\theta)$ is the Coriolis torque matrix and centrifugal torque matrix respectively.

$$C(\theta)=\begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \quad \text{is centripetal force.}$$

$$\text{Where,}\quad \begin{cases} c_{11}=c_{12}=c_{13}=c_{22}=c_{31}=c_{33}=0 \\ c_{21}=\dfrac{1}{8}m_2\mathrm{a}_2^2S_{22}+\dfrac{1}{2}m_3\mathrm{a}_2^2S_{22}+\dfrac{1}{2}\mathrm{a}_2\mathrm{a}_3m_3S_{22+3}+\dfrac{1}{8}\mathrm{a}_3^2S_{2(2+3)} \\ c_{23}=-\dfrac{1}{2\dot{\theta}_3}m_3\mathrm{a}_2\mathrm{a}_3S_3 \\ c_{32}=\dfrac{1}{2}m_3\mathrm{a}_3S_{23}\left(\mathrm{a}_2C_2+\dfrac{1}{2}\mathrm{a}_3C_{23}\right) \end{cases}$$

Where, $G(\theta)$ is gravity vector for three link robotic arm. Where,

$$G(\theta)=\begin{bmatrix} 0 \\ \dfrac{1}{2}\mathrm{a}_2m_2gC_2+\mathrm{a}_2m_3gC_2+\dfrac{1}{2}\mathrm{a}_3m_3gC_{23} \\ \dfrac{1}{2}m_3g\mathrm{a}_3C_{23} \end{bmatrix}$$

Where $G(\theta)$ is the force of gravity vector for the proposed three link robot manipulator system and it is a nx1 matrix. In this thesis it is 3x1 matrix. The output angle θ is also an 3x1 matrix vector and $\dot{\theta}$ is joint velocity.

3.6.3 Considered Friction

In robotic manipulator, friction phenomena may affect the accuracy of position and joint velocity control. In order to obtain the dynamic equation of 3-DOF articulated robotic manipulator for real physical system at least there should be approximate model of friction of forces (Smith & Christensen, 2009). To simplify the model, the friction phenomena consist only the viscous and coulomb term. A precise and simple friction model can considerably improve the overall performance of accuracy and control stability of robotic manipulator system. Friction phenomena model can be described in (3.74) (Moldoveanu, 2014).

$$F(\dot{\theta})=F_v\dot{\theta}+F_s\,\mathrm{sgn}(\dot{\theta}) \tag{3.74}$$

Where, $F(\dot{\theta})$ is friction force, $\dot{\theta}$ is the joint velocity, F_v is the viscous damping coefficient and F_s is the direction dependent coulomb friction coefficient.

By considering the values of F_v and F_s from friction model 2.66 and 0.04 respectively (Moldoveanu, 2014). by inserting these values in to mathematical model of system, the overall system dynamics model of the system is evaluated.

3.6.4 Considered disturbance

As industrial articulated robot transfers an object from one place to another by lifting different masses, the dynamics model of the manipulator will also be changed. Then the robot manipulator is subjected to load uncertainty and vibration of torque. The above (3.73) is modified to (3.75) which include both friction model and external disturbance dynamics uncertainty. Therefore, in this thesis the general overall dynamics model of the system including disturbance uncertainty is expressed in (3.75).

In figure 3.10, 3.11 and 3.12, the disturbance torque due to payload variation, vibration and the combined of two disturbance uncertainty are illustrated below

$$\tau = M(\theta)\ddot{\theta} + B(\theta)[\dot{\theta},\dot{\theta}] + C(\theta)[\dot{\theta}]^2 + G(\theta) + F(\dot{\theta}) + \tau_d \tag{3.75}$$

Where the torque disturbance due to payload variation and vibration is denoted in terms of τ_d.

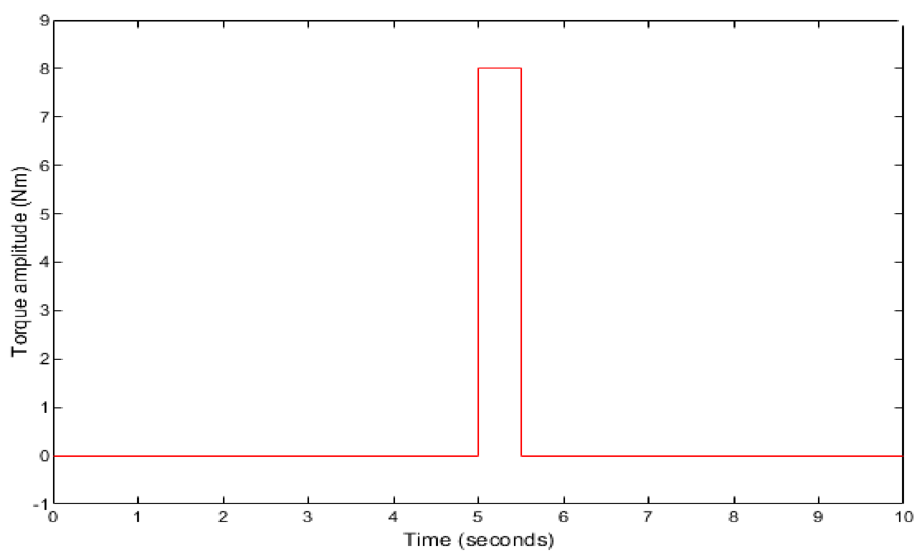


Figure 3.10 Disturbance torque due to payload variation

Industrial articulated robotic manipulators are worked with hazardous environment and when it worked, vibration torque noise and external payload torque disturbance happened on the system(Hsia & Gao, 1990). Therefore, in this thesis payload and vibration torque noise are considered. For vibration torque of uncertain disturbance, the consideration of white noise power for band limit, sampling time and seed are 0.000002, 0.001 and 100 respectively.

The input payload torque disturbance to the system is considered here for time between 5 secs to 5.5 secs. Considered amplitude of 8 Nm between the selected seconds for pay load variation is considered. Evaluation and analyzed of the performance of the proposed controller considering sum up of these two-torque input disturbance to the system.

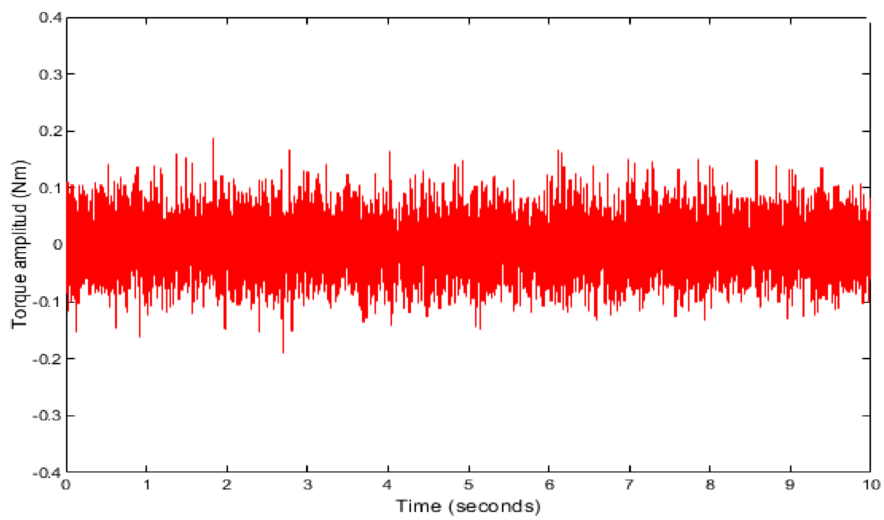


Figure 3.11 Disturbance torque due to vibration torque

Vibration torque disturbance is happened in the system when there is movement of link. in this thesis white noise band limit for vibration is used. In figure 3.12, the combination of torque due to payload variation and vibration torque is presented below to see the performance of the controller to reject the disturbance factors.

In Simulink model the proposed controller is tested in terms of rejecting external torque disturbance due to pay load variation and vibration torque.

Therefore, the payload variation disturbance is designed below signal:

$8u(t-5)-8u(t-5.5)$ where, u(t) is step input time but the vibration disturbance is unmeasured or unknown external disturbance.

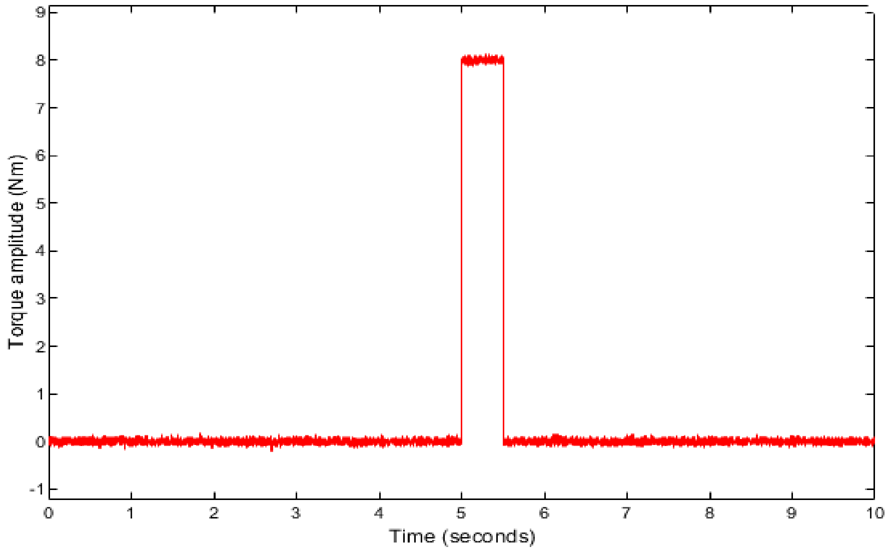


Figure 3.12 Disturbance torque combination due to payload variation and vibration.

Finally, the nonlinear mathematical or dynamic model representation for 3-DOF articulated robotic manipulator without torque disturbance is as follow as:

$$
\begin{aligned}
\begin{bmatrix}\ddot{\theta}_1\\ \ddot{\theta}_2\\ \ddot{\theta}_3\end{bmatrix} =& \begin{bmatrix} 0.0028+0.1875C_2^2+0.0625C_{23}^2+0.0625C_2C_{23} & 0 & 0\\ 0 & 0.1874+0.125C_3 & 0.0312+0.0625C_3\\ 0 & -0.0312-0.0625C_3 & 0.0417\end{bmatrix}^{-1}\\
&\left\{\begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\end{bmatrix} - \begin{bmatrix} -0.1562S_{22}-0.125S_{22+3}-0.0312S_{2(2+3)} & -0.125C_2S_{23}-0.0312S_{2(2+3)} & 0\\ 0 & 0 & -0.125S_3\\ 0 & 0 & 0.0625S_3\end{bmatrix}\right.\\
&\begin{bmatrix}\dot{\theta}_1\dot{\theta}_2\\ \dot{\theta}_1\dot{\theta}_3\\ \dot{\theta}_2\dot{\theta}_3\end{bmatrix} - \begin{bmatrix} 0 & 0 & 0\\ 0.0781S_{22}+0.0625S_{22+3}+0.0156S_{2(2+3)} & 0 & \dfrac{-0.0625}{\dot{\theta}_3}S_3\\ 0 & 0.0625C_2S_{23}+0.0312S_{23}C_{23} & 0\end{bmatrix}\\
&\left.\begin{bmatrix}\dot{\theta}_1^2\\ \dot{\theta}_2^2\\ \dot{\theta}_3^2\end{bmatrix} - \begin{bmatrix}0\\ 3.6788C_2+1.2263C_{23}\\ 1.2263C_{23}\end{bmatrix} - \begin{bmatrix}2.66\dot{\theta}_1+0.04\,\mathrm{sgn}(\dot{\theta}_1)\\ 2.66\dot{\theta}_2+0.04\,\mathrm{sgn}(\dot{\theta}_2)\\ 2.66\dot{\theta}_3+0.04\,\mathrm{sgn}(\dot{\theta}_3)\end{bmatrix}\right\} \qquad (3.76)
\end{aligned}
$$

CHAPTER FOUR

CONTROLLER DESIGN

4.1 Introduction

Model predictive control has a role and function in optimal control for the system's plant dynamics. The main principle of MPC is to anticipate future system behavior using a dynamic model based on current state and historical record measurement data in order to make the optimum decision by optimizing forecasted behavior at each sampling time recursively (Massaoudi et al., 2019). Model predictive control employs explicit models to forecast future plant behavior and can handle input and state limitations in dynamic systems. Every sampling time, the control sequence in MPC is determined by solving an optimization problem (Iordanou & Surgenor, 1996).

In this section, construction of model predictive control for a 3-DOF industrial articulated robotic manipulator and comparison of it to the performance of the PIDGA optimum control algorithms. First linearize the overall nonlinear dynamic model of the plant by using feedback linearization (inverse dynamics control). After linear model is obtained, model predictive control is developed for each joint or decoupled linear control approach.

4.2 Feedback Linearization

Inverse dynamics control (feedback linearization) in the joint space for the plant nonlinear dynamics of the system is used here. The goal of this linearization is to track any arbitrary input θ_d by providing a control input τ in terms of time, such that it cancels the nonlinearity terms of the plant dynamics of the system due to approximate friction model, Coriolis, centripetal, and gravity terms and these terms can be estimated with high degree of accuracy and making the resulted system decoupled. Here, derivate the output θ with respect to time until the control input torque appeared. In this thesis, the input appears in the second derivative.

$$\ddot{\theta} = M(\theta_i)^{-1}[\tau - B(\theta_i)[\dot{\theta}_i, \dot{\theta}_i] \text{-} C(\theta_i)[\dot{\theta}_i]^2 - G(\theta_i) - F(\dot{\theta}_i)] \tag{4.1}$$

The feedback linearization control joint torque τ_i

$$\tau_i = M(\theta_i)\chi_i + B(\theta_i)[\dot{\theta}_i, \dot{\theta}_i] + C(\theta_i)[\dot{\theta}_i]^2 + G(\theta_i) + F(\dot{\theta}_i) \tag{4.2}$$

Where: $\chi = \begin{bmatrix} \chi_1 & \chi_2 & \chi_3 \end{bmatrix}^{\mathrm{T}}$ is the synthetic control vector feed by controller for each joint and equal with the second derivative of the output joint variable vector (acceleration of the link). By comparing (4.2) with (3.75), the linear model is derived in (4.3).

$$\ddot{\theta}_i(t) = \chi_i(\mathrm{t}) \quad i = 1, 2, 3 \tag{4.3}$$

By using (4.3), the double integrator open loop transfer function is as follow:

$$\frac{\theta_i(s)}{\chi_i(s)} = \frac{1}{s^2} \tag{4.4}$$

Then, optimization of energy and prediction horizon time for developed controller design of three decoupled linear systems is solved.

4.3 Model Predictive Control Design

From above general decoupled obtained linearized model of a plant, the model predictive control for each link of the industrial articulated robotic arm is developed. But, the developed MPC control law for linearized plant model is the same for all of the link of robotic arm. Therefore, developing MPC control for one works also for others. After inverse dynamic control, the three decoupled linear systems is obtained (Piltan et al., 2012).

$$\begin{cases} \ddot{\theta}_1 = \chi_1 \\ \ddot{\theta}_3 = \chi_2 \\ \ddot{\theta}_3 = \chi_3 \end{cases} \tag{4.5}$$

For constructing an optimization model for model predictive control by taking assumption that, the synthetic control vector $\chi_1(t) = \chi_1$ is fixed for prediction horizon time for $[t,\ t+p]$ and by using above formula (4.5), the prediction model is obtained. Here, considered a discrete time approximation to the double integrator open loop transfer function system. The synthetic control input vector is fixed between starting time and prediction horizon time (Bemporad, 2006). By using bellow notation, the MPC control law for the 1st 2nd and 3rd joint angle of the industrial articulated robotic arm is developed.

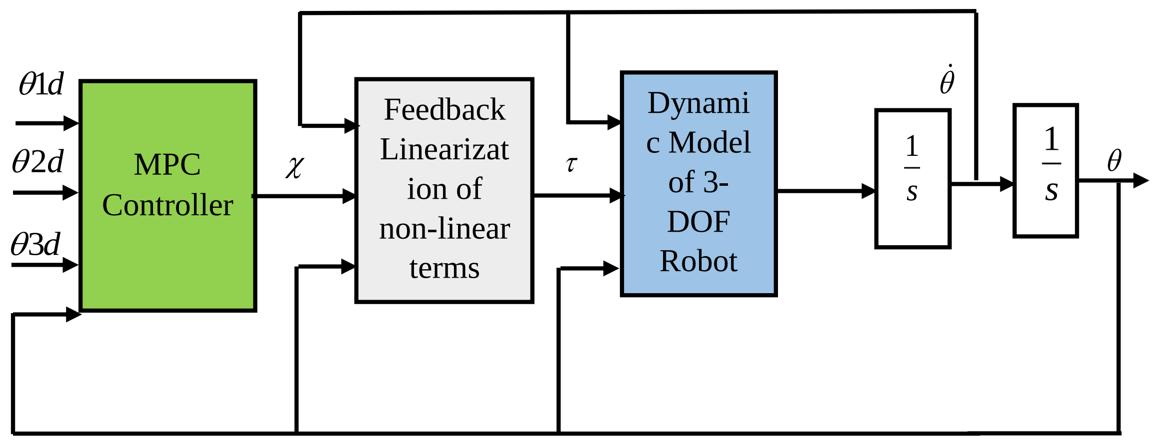


Figure 4.1 closed loop system of model predictive control

Where,

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} \quad and \quad \theta_d = \begin{bmatrix} \theta_{1d} \\ \theta_{2d} \\ \theta_{3d} \end{bmatrix}$$

The middle feedback linearization block represents the inverse dynamics control equation (4.6). The main point of this linearization is to cancel the non-linearity terms of nonlinear dynamic model representation.

Let, $x_1 = \theta_1 = y$, θ_1 is link angle and $x_2 = \dot{\theta}_1 = \dot{y}$, $\dot{\theta}_1$ is link angular velocity

$$\begin{aligned} \frac{d}{dt}x_1 &= \dot{x}_1(t) = x_2 = \dot{\theta}_1 \\ \frac{d}{dt}x_2 &= \dot{x}_2(t) = \ddot{x}_1 = \ddot{\theta}_1 = \chi_1(t) \end{aligned} \tag{4.6}$$

$\chi_1(t)$ is synthetic control vector for link 1

After integrating the second equation in (4.6) from interval t to $t+p$ gives as:

$$x_2(t+p) = x_2(t) + p\chi_1(t) \tag{4.7}$$

Where, p is prediction horizon time

Substitute (4.7) in to (4.6) and then integrating the first equation of (4.6) from time t to $t+p$. After that the prediction model for the plant is obtained:

$$\begin{aligned} x_1(t+p) &= x_1(t) + p x_2(t) + \frac{1}{2} p^2 \chi_1(t) \\ x_2(t+p) &= x_2(t) + p \chi_1(t) \end{aligned} \tag{4.8}$$

The discretization state space model for the linearized plant model is:

$$\begin{bmatrix} x_1(t+p) \\ x_2(t+p) \end{bmatrix} = \begin{bmatrix} 1 & p \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} \dfrac{p^2}{2} \\ p \end{bmatrix} \chi_1(t) \tag{4.9}$$

The general form for the above discretize state space model is:

$$\begin{aligned} x_{k+1} &= A x_k + B u_k \\ y_k &= C x_k \end{aligned}$$

By using considered notation above substitute (4.8) in to the obtained prediction model. The correct prediction model in terms of the system variable term θ_1 with time is obtained:

$$\begin{aligned} \theta_1(t+p) &= \theta_1(t) + \dot{\theta}_1(t) p + \frac{1}{2} \chi_1 p^2 \\ \dot{\theta}_1(t+p) &= \dot{\theta}_1(t) + \chi_1 p \end{aligned} \tag{4.10}$$

In order to minimize the energy and prediction horizon time, the variable weighting and horizon time are included in the cost function. For stabilizing and controlling the system with minimum effort, the horizon time quadratic cost function is chosen:

$$J = w \dot{e}_i^2(t+p) + \int_t^{t+p} e_i^2(t+p)\,dt \quad i = 1, 2, 3 \tag{4.11}$$

Where p and w are both positive parameter, they are prediction horizon time and the weight factor respectively.

MPC controller can handle constraints of input unit. In this research, the bounded control input torque for 3-DOF industrial articulated robotic manipulator of each link is defined as:

$$|\tau_i| \le 40 \text{ Nm} \quad \text{joint torque where i} = 1,2,3 \tag{4.12}$$

Where p and w are both positive parameter, they are prediction horizon time and the weight factor respectively.

The predicted angle error for the first link is:

$$e_1(t+p) = \theta_{1d} - \theta_1(t+p) \tag{4.13}$$

The predicted velocity error for:

$$\dot{e}_1(t+p) = -\dot{\theta}_1(t+p) \tag{4.14}$$

By inserting the above prediction model obtained in to the cost function and then minimize with respect to synthetic control input vector χ .

$$J = w(-\dot{\theta}_1(t+p))^2 + \int_t^{t+p} (\theta_{1d} - \theta_1(t) - \dot{\theta}_1(t)p - \frac{1}{2}\chi_1 p^2)^2 \ dt \tag{4.15}$$

$$\begin{aligned} J = w(-\chi_1 p - \dot{\theta}_1(t))^2 + \int_t^{t+p} (\theta_{1d}^2 - \theta_1(t)\theta_{1d} - \dot{\theta}_1(t)\theta_{1d}p - \frac{1}{2}\chi_1 p^2\theta_{1d} - \frac{1}{2}\chi_1 p^2\theta_{1d} + \frac{1}{4}\chi_1^2 p^4 + \ldots \\ \frac{1}{2}\chi_1 p^3\dot{\theta}_1(t) + \frac{1}{2}\chi_1 p^2\theta_1(t) - \dot{\theta}_1(t)\theta_{1d}p + \frac{1}{2}\chi_1 p^3\dot{\theta}_1(t) + \dot{\theta}_1^2(t)p^2 \ldots \\ + \dot{\theta}_1(t)\theta_1(t)p - \theta_1(t)\theta_{1d} + \frac{1}{2}\chi_1 p^2\theta_1(t) + \dot{\theta}_1(t)\theta_1(t)p + \theta_1^2(t))dt \end{aligned} \tag{4.16}$$

After integrating and rearranging, minimize with respect to input synthetic control vector χ_1, and the model new predictive control law is obtained:

$$\chi_1 = \frac{2p}{p^3 + 4w}(\theta_{1d} - \theta_1(t)) - \frac{2p^3 + 4w}{p^4 + 4pw}(\dot{\theta}_1(t)) \tag{4.17}$$

$$\chi_1 = \alpha(\theta_{1d} - \theta_1(t)) - \beta\dot{\theta}_1(t) \tag{4.18}$$

The model predictive control law for second and third link of robot manipulator is defined below respectively and its gains are the same with first link of robot.

$$\chi_2 = \alpha(\theta_{2d} - \theta_2(t)) - \beta\dot{\theta}_2(t) \tag{4.19}$$

$$\chi_3 = \alpha(\theta_{3d} - \theta_3(t)) - \beta\dot{\theta}_3(t) \tag{4.20}$$

Where α and β are predictive control gain of the linearized plant and the same is also true for second and third link of the robot.

$$\alpha = \frac{2p}{p^3 + 4w} \quad \text{and} \quad \beta = \frac{2p^3 + 4w}{p^4 + 4pw} \tag{4.21}$$

The general model predictive control law for the 1st, 2nd and 3rd link of robot is:

$$\chi_i = \alpha(\theta_{id} - \theta_i(t)) - \beta\dot{\theta}_i(t) \quad i = 1, 2, 3 \tag{4.22}$$

From (4.22), the new predictive control law block diagram representation for 3-DOF articulated robotic manipulator.

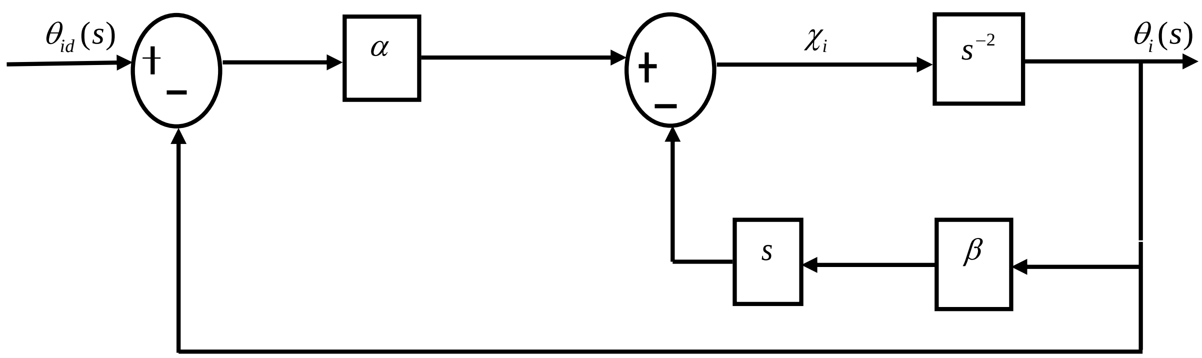


Figure 4.2 Block diagram of Model predictive control law

Transfer function is the Laplace transform of output per Laplace transform of the input by making all initial condition zero.

The system transfer function of closed loop for figure 4.2 is:

$$\frac{\theta_i(s)}{\theta_{id}(s)} = \frac{\alpha}{s^2 + \beta s + \alpha} \tag{4.23}$$

The general behavior form of second order system is (Bouzoualegh, 2019):

$$\frac{\omega_0^2}{s^2 + 2\xi\omega_0 s + \omega_0^2} \tag{4.24}$$

where ω_0 is natural frequency and ξ is a damping factor and equate the above second order system and closed loop response predictive control of system.

$$\frac{2p}{p^3+4w}=\omega_0^2 \tag{4.25}$$

$$\frac{2p^3+4w}{p^4+4pw}=2\xi\omega_0 \tag{4.26}$$

from first (4.25)

$$\frac{2p-p^3\omega_0^2}{4\omega_0^2}=w \tag{4.27}$$

By Substituting (4.27) in to (4.26), the behavior similar to second order equation is obtained.

$$p^2\omega_0^2-4\xi\omega_0 p+2=0 \tag{4.28}$$

$$\omega_0 p=c \tag{4.29}$$

$$c^2-4\xi c+2=0$$

$$c_1=2\xi+\sqrt{4\xi^2-2} \quad \text{and} \quad c_2=2\xi-\sqrt{4\xi^2-2} \tag{4.30}$$

From the solution the root should be $4\xi^2-2\geq 0$. Then the value of ξ is greater than or equal to 0.7071. From Figure 4.31 above to make the weight factor positive and real the above two solutions $c_1^2$and c_2^2 should be less than 2. Therefore, only the second solution satisfy the equation. Chose damping factor $\xi=0.74$. The closed loop performance of the system was solved for different value of natural frequency from 1 rad/s to 10 sec. in order to measure and know efficiency performance of the proposed MPC controller, the design control part of PIDGA control for 3-DOFarticulated robotic manipulator system for comparison purpose is included. From table below, to obtain acceptable control signal $|\tau_i|\leq 40$ Nm joint torque where $\mathrm{i}=1,2,3$, Chose the natural frequency, which values have 7.5 rad/sec because since it is optimal, it is bounded by PIDGA output. For this natural frequency prediction horizon is 0.13925 and weighting factor is 5.6329e-04.

Table 4.1 System performance for different value of natural frequency

MPC controller								
Natural Frequency (rad/sec)	a	β	τ_1 min (Nm)	τ_1 Max (Nm)	τ_2 Min (Nm)	τ_2 Max (Nm)	τ_3 Min (Nm)	τ_3 Max (Nm)
ω_0 =1	1	1.48	0.5907	1.8781	3.4374	4.6677	0.0081	1.2426
ω_0 =4	16	5.92	-0.1925	8.5645	0.9385	3.5709	-2.7859	1.4880
ω_0 =7	49	10.36	-0.2244	20.2442	-6.7606	3.8747	-5.8180	2.0280
ω_0 =7.5	56.25	11.1	-0.2259	22.909	-8.4806	4.0099	-6.3164	2.1466
ω_0 =10	100	14.8	-0.1892	39.8488	-18.860	4.9641	-8.7970	2.8625

At ω_0 =7.5 rad /sec and $\xi = 0.74$, the prediction horizon and weighting factor using (4.29) and (4.27) is 0.13925 and 5.6329e-04 respectively. Then, using (4.21) the model predictive control gain is obtained. Thus, α is 56.2500 and β is 11.1000.

4.4 PID Control Design

A proportional, integral, and derivative controller are combined into a PID controller, which incorporates the key properties of all three elements. The goal of the job, according to Figure 1, is to adjust the PID gains with the least amount of error between the preset value and the true concentration value, where e is the error, which is the difference between the expected input and the actual measured output value. Integral absolute error approach is one of the error criterion for achieving optimized PID tuning values. The objection function for best minimization of the error is chosen and the performance index is obtained by trapezoidal integration rule. Fitness value 0.5607.

$$J = \int_0^\infty (|e(:,1)| + |e(:,2)| + |e(:,3)|)dt \tag{4.31}$$

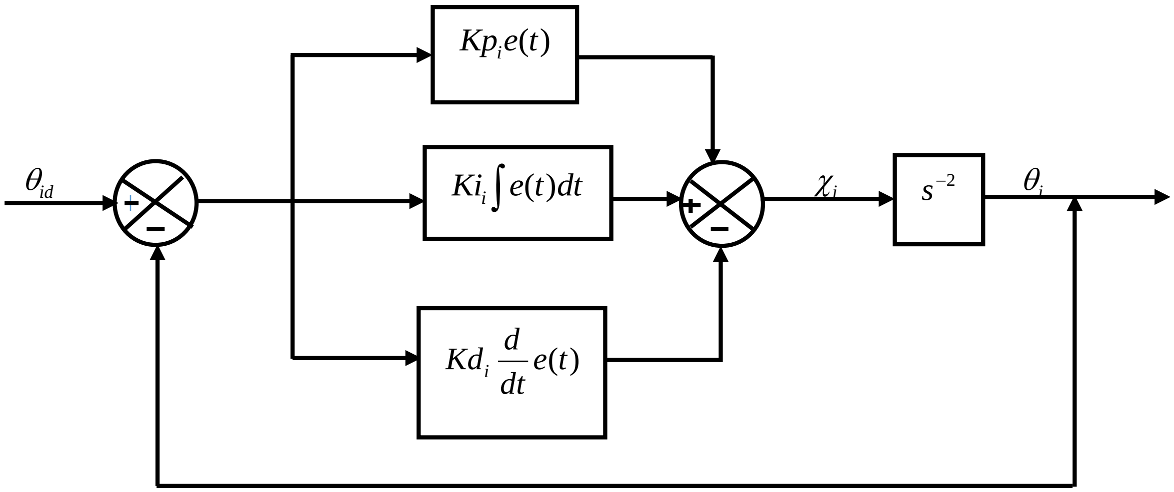


Figure 4.3 Block diagram of closed loop PID control law

From Figure 4.3, the PID control law for plant is:

$$\chi_i = Kp_i e(t) + Kd_i \dot{e}(t) + Ki_i \int e(t)dt, \quad i = 1,2,3 \tag{4.32}$$

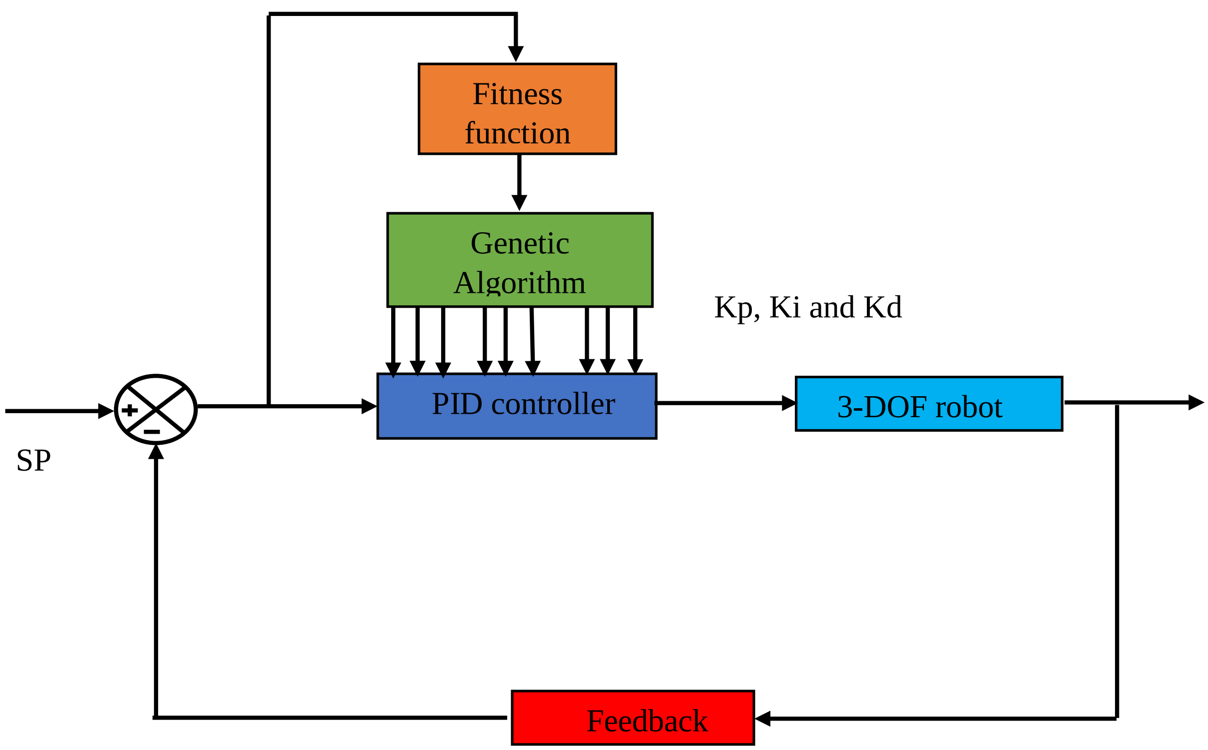


Figure 4.4 Block Diagram of Genetic Algorithms Based PID Controller

The population size and maximum generation for this thesis using PIDGA is chosen. The value 40 and 50 are chosen based on resulted outcomes that give best optimal gain and the consideration of the MATLAB that takes time to tune the PID controller for 3-DOF articulated robotic manipulator.

Table 4.3 GA parameter used for plant

Population size	**40**
Maximum generation	50

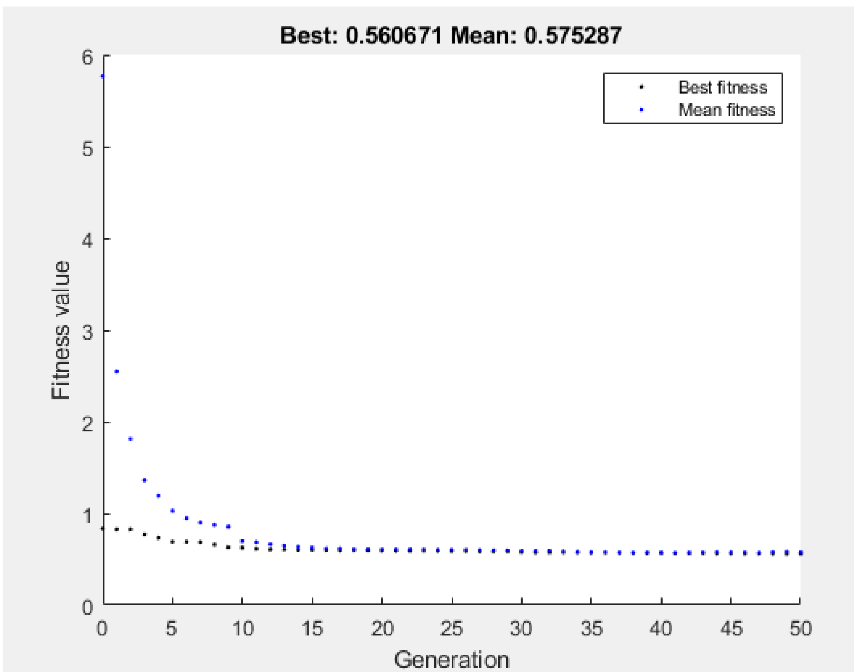


Figure 4.5 Fitness value using PIDGA

4.4.1 Genetic Algorithm

A genetic algorithm is a method for solving nonlinear systems of equations and optimizing difficult problems that uses a random search strategy. GA handles a population of alternative solutions known as individuals or chromosomes that evolve iteratively and uses probabilistic transition rules instead of deterministic ones. A generation is a unit of measurement for each iteration of the algorithm. A fitness function, as well as genetic operators including reproduction, crossover, and mutation, are used to mimic the development of solutions (Man et al., 1996). Fitness is a criterion for determining chromosomes fitness.

Steps of Genetic algorithm for code:

Step 1: Assign a population size of random solutions for the parameter of rated unit and maximum generations.

Step 2: Number of parameter variable that used and put the lower bound and upper bound

Step 3: Using trapezoidal integration rule, minimize the error

Step 4: go to Step 2, till get the best optimum result.

By looking the come out fitness value of PID controller, the bound is obtained. Firstly, take random number interval and then, by looking fitness value of it and looking the obtained optimal gain for the bounded interval, observe and analyze the value. After that, increase the bound number of the previous and looking the optimal gain, if gain of the current value is unchanged or close to the number of the previous, the bound that defined is taken for obtaining the best final optimal value and finally the bound is chosen. Based on these, the bound is chosen for proportional integral derivative with genetic algorithm for this thesis.

Table 4.3 Tuned parameter boundary of PID controllers using genetic algorithm

Boundary	**Kp1**	**Kp2**	**Kp3**	**Ki1**	**Ki2**	**Ki3**	**Kd1**	**Kd2**	**Kd3**
Lower	0	0	0	0	0	0	0	0	0
Upper	60	60	60	3	3	3	15	15	15

CHAPTER FIVE

RESULTS AND DISCUSSIONS

5.1 Introduction

In this chapter, the simulation is carried out using MATLAB Simulink and code to verify the performance of controller for three DOF industrial articulated robotic manipulator system and comparing the output result with Genetic algorithm PID controller. The trajectory tracking performance for three controllers was tested and obtained output result for GAPID and MPC controller. Based on the obtained result, MPC has better performance than GAPID.

The open loop dynamic model of the plant simulation was performed based on the parameters of robotic manipulator in Table 3.1. MATLAB/Simulink of open loop dynamic model of 3-DOF articulated robotic manipulator without disturbance is designed as shown in Figure 5.1.

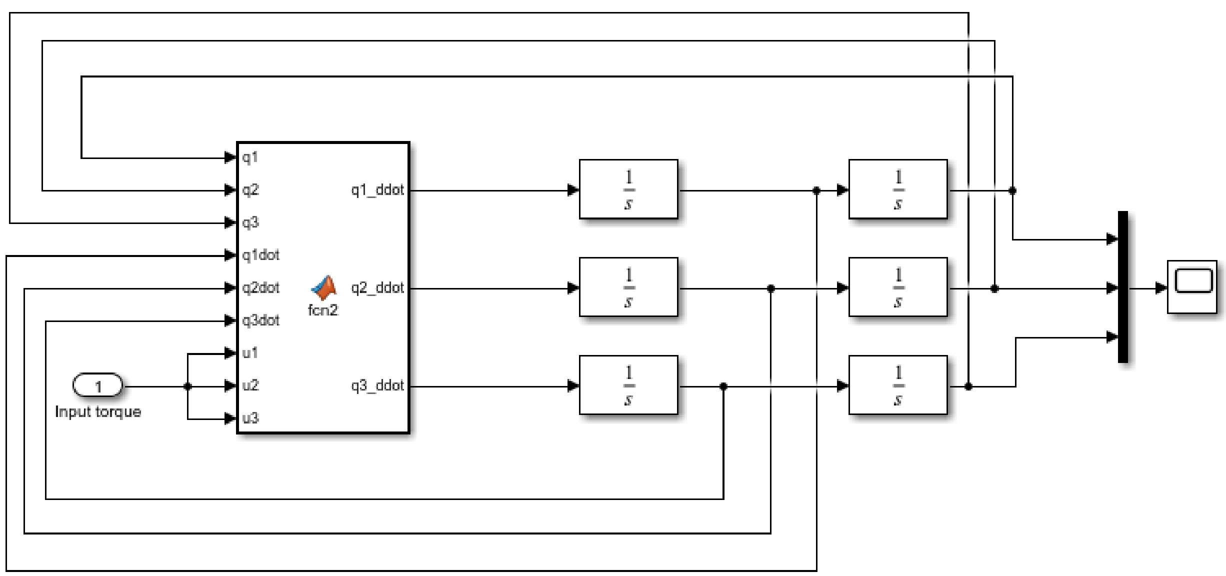


Figure 5.1 MATLAB/Simulink open loop dynamic model of 3-DOF articulated robotic manipulator

From figure 5.1 the notations are:

$q1=\theta_1$, $q2=\theta_2$, $q3=\theta_3$ are joint angle for 1st,2nd and 3rd joint

$q1dot=\dot{\theta}_1$, $q2dot=\dot{\theta}_2$, $q3dot=\dot{\theta}_3$ are joint angular velocity for 1st,2nd and 3rd joint

$u1 = \tau_1$, $u2 = \tau_2$, $u3 = \tau_3$ are the input torque for 1st , 2nd and 3rd joint

From the above Simulink fcn2 is a function of nonlinear dynamic system equation of motion for 3-DOF articulated robotic manipulator.

Based on the parameter in table 3.1, the result for open loop nonlinear dynamic model of the plant is obtained when torque is equal to some value, which is 5 Nm, in Figure 5.2.

As shown in figure 5.2, 3-DOF articulated robotic manipulator is highly nonlinear unstable system. Therefore, the plant dynamics needs a controller to stable the system.

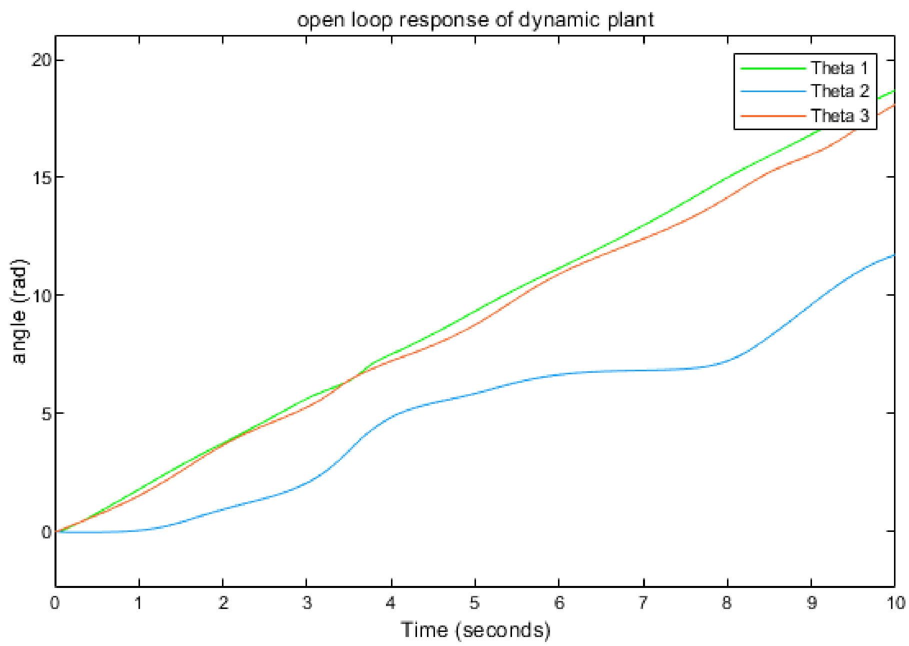


Figure 5.2 Open loop response dynamic model for 3-DOF articulated robotic manipulator

The above figure shows the plant dynamics system is unstable, thus it needs best optimal controller like MPC to stabilize it. Table 5.1 shows the optimal values PID parameters after tuning, corresponding to the minimum cost function value provided by the optimal GA. Moreover, Table 5.1, are also shown the minimum cost function value.

Table 5.1: Optimal value of PIDGA

PID using Genetic Algorithm								
Kp1	Kp2	Kp3	Ki1	Ki2	Ki3	Kd1	Kd2	Kd3
59.8715	57.1300	59.8410	0.0023	0.2977	0.1017	9.7203	8.8529	9.6193

Corresponding to the minimum cost function value provided by the optimal of MPC controller. The MPC control law is calculated by using the input constraint that satisfied torque as the end effector reach to the desired point optimally. At ω_0 =7.5 rad /sec and $\xi = 0.74$, the prediction horizon and weighting factor using (4.29) and (4.27) is 0.13925 and 5.6329e-04 respectively. Then, using (4.21) the model predictive control gain is obtained. Thus, α is 56.2500 and β is 11.1000.

Table 5.2: Optimal value of MPC control

Controllers	α	β
MPC controller	56.2500	11.1000

For Figure 3.7 Initially at angular position 000, the end effector coordinates are obtained using homogenous transformation matrix of petercorke MATLAB.

$$H = \begin{bmatrix} 1.0000 & 0 & 0 & 1.0000 \\ 0 & 0.0000 & -1.0000 & 0 \\ 0 & 1.0000 & 0.0000 & 0.15 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix}$$

From this the end effector coordinates at 000 degrees are:

$$\begin{cases} x = 0.0000 \\ y = 0.5624 \\ z = -0.5830 \end{cases}$$

The output of the final end effector coordinates of the axis for 3-DOF articulated robotic manipulator is obtained using MATLAB code. The homogenous transformation matrix at pi/2, -pi/6 and –pi/4.

$$H = \begin{bmatrix} 0.0000 & 0.0000 & 1.0000 & 0.0000 \\ 0.2588 & 0.9659 & -0.0000 & 0.5624 \\ -0.9659 & 0.2588 & 0.0000 & -0.5830 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix}$$

From this the end effector coordinates at pi/2, -pi/6 and –pi/4 are:

$$\begin{cases} x = 0.0000 \\ y = 0.5624 \\ z = -0.5830 \end{cases}$$

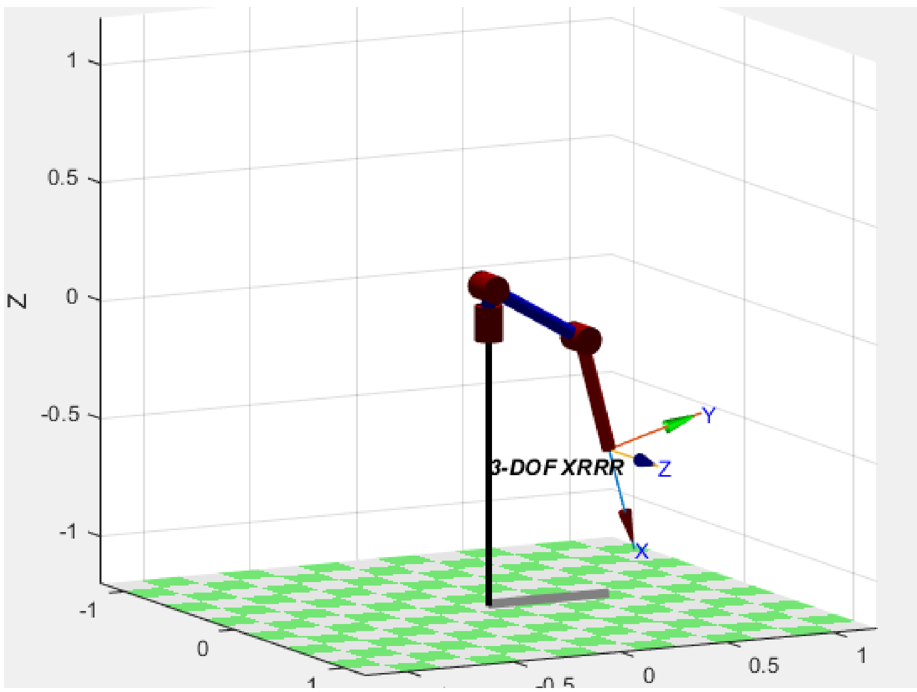


Figure 5.3 3-DOF robotic manipulator at pi/2, -pi/6 and –pi/4

5.2 Simulation Result using Controller

In addition to model predictive control (MPC) for comparison purpose and efficiency of my controllers, the plant is controlled and its performance is compared using PID with GA controller. From the result that obtained using two different controllers, the parameters of proportional derivative integral are optimized using genetic algorithm, and the parameters of MPC controllers are found by using the energy required (input torque constraint) of robotic manipulator to reach the robot end effector at the final point. For 3-DOF articulated robotic arm the comparison output articulation result of PIDGA and MPC controller are plotted and discussed in Figure 5.4, Figure 5.5 and Figure 5.6. The initial and desired articulations configurations are as followed:

$$\theta_{initial} = \begin{bmatrix} \theta_{10} \\ \theta_{20} \\ \theta_{30} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\theta_{desired} = \begin{bmatrix} \theta_{1d} \\ \theta_{2d} \\ \theta_{3d} \end{bmatrix} = \begin{bmatrix} \frac{\pi}{2} \\ \frac{-\pi}{6} \\ \frac{-\pi}{4} \end{bmatrix}$$

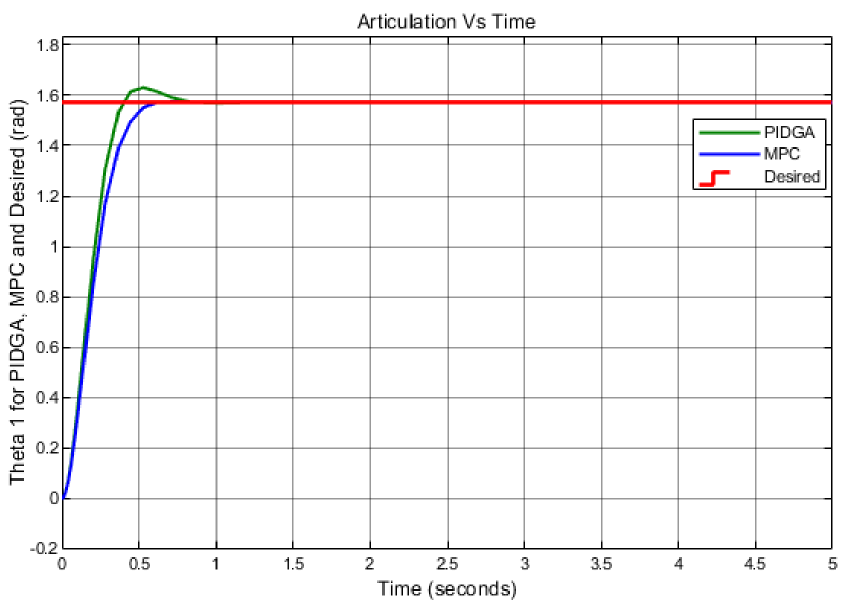


Figure 5.4 Desired and real orientations of the first link of the robot using PIDGA and MPC controller

From figure 5.4, in settling time, fast rate convergence, and overshoot, the proposed controller has better performance than PIDGA controller. However, PIDGA has less rise time than the proposed controller. In general, the result obtained in MPC controller has a better performance than PIDGA controller based on time domain specification performance.

Table 5.3: System performance of MPC controller with PIDGA for first articulation

Performance measurement	PIDGA	MPC
Rise time (secs)	0.2527	0.3168
Settling time (secs)	0.6608	0.5161
Overshoot (%)	3.7214%	0.275%

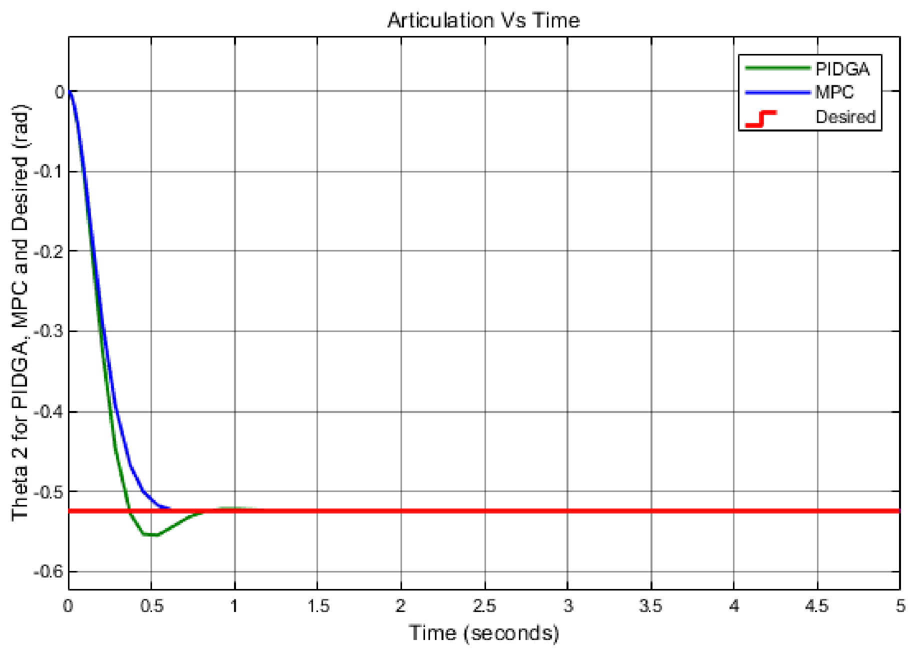


Figure 5.5 Desired and real orientations of the second link of the robot using PIDGA and MPC controller

From figure 5.5, the rise time, settling time and overshoot for the second articulation is the same as the first articulation for MPC controller. In table 5.4 the time domain specification for angular position two is presented.

Table 5.4: System performance of MPC controller with PIDGA for Second articulation

Performance measurement	PIDGA	MPC
Rise time (secs)	0.2410	0.3168
Settling time (secs)	0.7101	0.5161
Overshoot (%)	5.9657%	0.275%

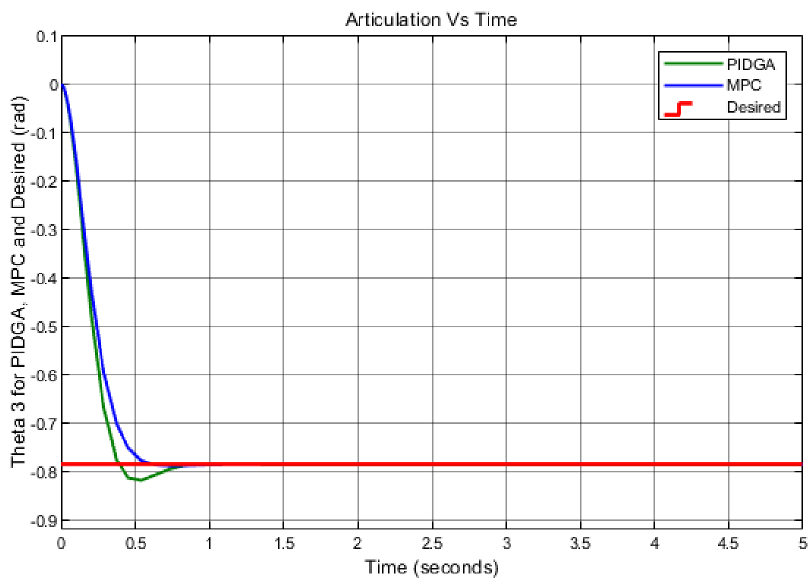


Figure 5.6 Desired and real orientations of the third link of the robot using PIDGA and MPC controller

From the above figure 5.4, figure 5.5 and figure 5.6, the blue color describes the angular positions for three joints are controlled by model predictive optimal control method approach. From the result that obtained the figure represent the tracking of articulation to their desired set point and describes the efficiency of the controller. It is observed that MPC controller have quickly converge to the desired point with small overshooting and handling constraint a behavior like critically damped oscillation. In table 5.5, the time domain specification performance value is presented.

Table 5.5: System performance of MPC controller with PIDGA for third articulation

Performance measurement	PIDGA	MPC
Rise time (secs)	0.2500	0.3168
Settling time (secs)	0.6677	0.5161
Overshoot (%)	4.0252%	0.275 %

In figure 5.7, 5.8 and 5.9, the input torques which is given by each actuator is illustrated in below using MATLAB/Simulink.

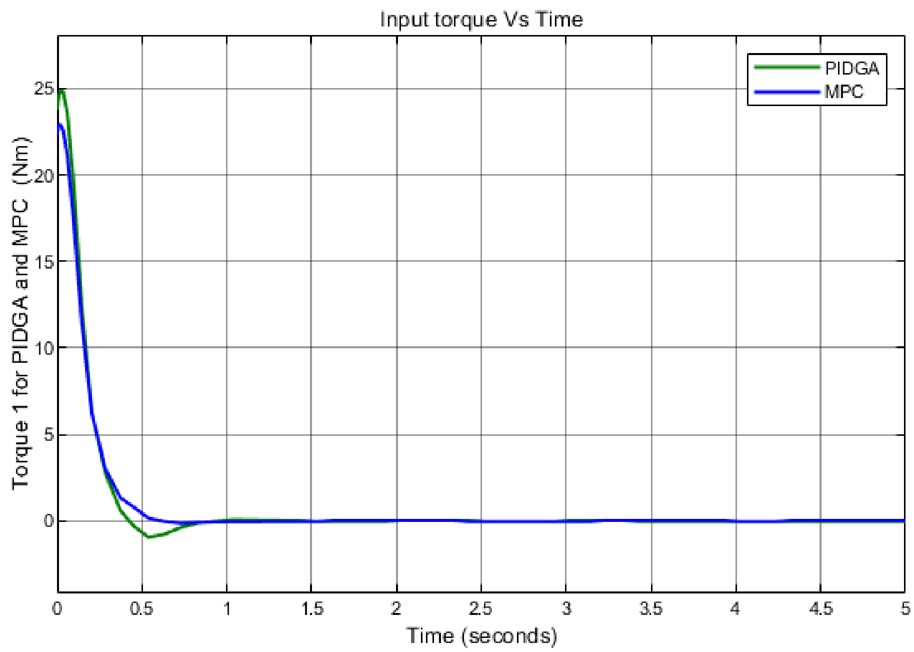


Figure 5.7 Torques of 1st link of the robot using PIDGA and MPC controller

By using PIDGA controller output, the ranged constraint value for MPC controller is taken and MPC performs well in a given constraint input which is 40Nm. In this thesis, in order to control the first link of the robot using PIDGA and MPC, an input torque is found, which is given by PMDC motor. The maximum torque that used to track the first joint variable link for PIDGA and MPC controllers are 24.8649 Nm and 22.9040 Nm respectively. therefore, from the result that obtained MPC controller use lower consumption of energy and converges quickly to the desired point. In Table 5.6 below describes, MPC track the 1st joint variable using efficient energy than PIDGA.

The amplitude of the torque means the distance between minimum and maximum value of the torque. maximum torque is the peak torque that used to track the desired set point.

Table 5.6 Starting, maximum and amplitude torque used at first joint for PIDGA and MPC

Torque	PIDGA	MPC
Starting torque (Nm)	23.776	22.3378
Amplitude of torque (Nm)	25.8143	23.0365
Maximum torque (Nm)	24.8649	22.9040

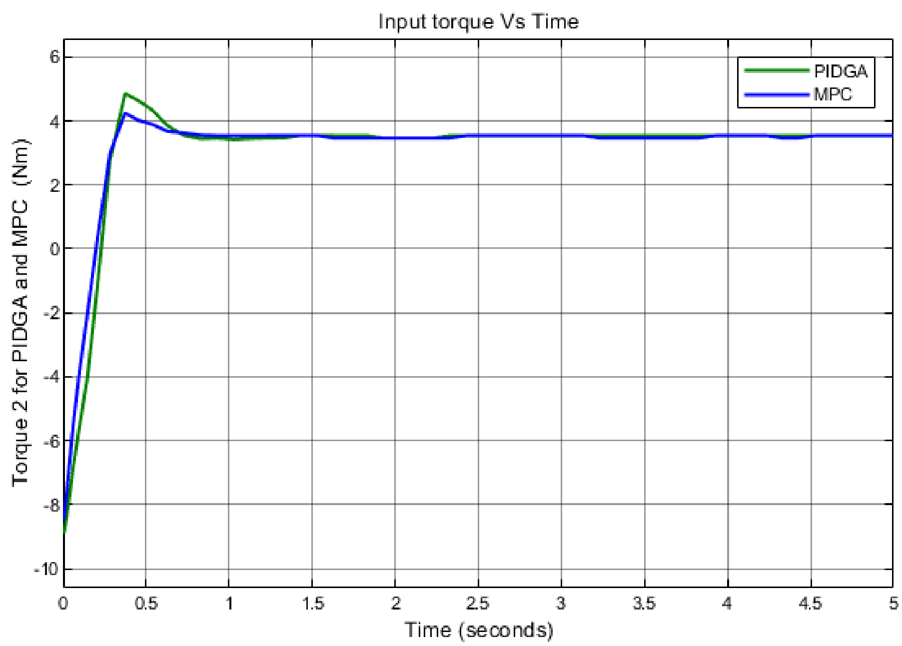


Figure 5.8 Torques of 2nd link of the robot using PIDGA and MPC controller

From the result obtained in order to control the second link of robotic manipulator, the proposed controller system used minimum torque and fast convergence than PIDGA controllers. The MPC controller input toque for second link robotic manipulator. The maximum torque that used to track the second joint variable link for PIDGA and MPC controllers are -8.8917 Nm and -8.4805 Nm respectively. In Table 5.7 describes, MPC controller uses lower consumption of energy than PIDGA.

Table 5.7 Starting, maximum and amplitude torque used at second joint for PIDGA and MPC

Torque	PIDGA	MPC
Starting torque (Nm)	-8.8490	-8.4406
Amplitude of torque (Nm)	13.7531	12.7277
Maximum torque (Nm)	-8.8917	-8.4805

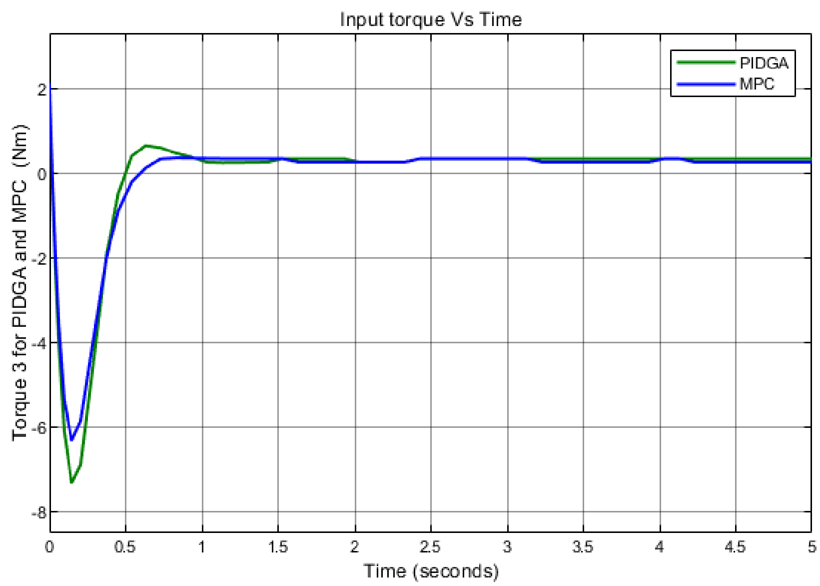


Figure 5.9 Torques of 3rd link of the robot using PIDGA and MPC controller

From figure 5.7, 5.8 and 5.9, from the obtained result, using an MPC controller for the plant dynamics have used lower energy minimization than PIDGA. The maximum torque that used to track the third joint variable link for PIDGA and MPC controllers are -7.3159 Nm and -6.3049 Nm respectively. Therefore, in order to control this nonlinear system, the proposed MPC controller is more preferable than PIDGA. Because the consumption of energy needed by the motor is lower. In Figure 5.10 the synthetic control vector is illustrated in below.

Table 5.8 Maximum starting, maximum and amplitude torque used at third joint for PIDGA and MPC

Torque	**PIDGA**	**MPC**
Starting torque (Nm)	2.0723	2.1466
Amplitude of torque (Nm)	9.3882	8.4515
Maximum torque (Nm)	-7.3159	-6.3049

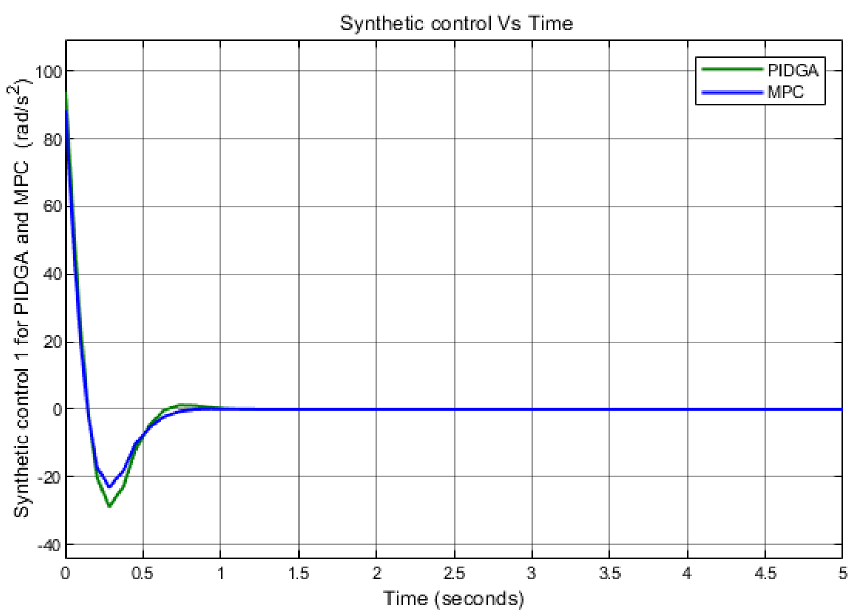


Figure 5.10 Synthetic control of 1st link of robotic manipulator using PIDGA and MPC controller

From figure 5.10, the synthetic control vectors χ_1 converge to zero after certain time that the end effector reaches the desired set point using PIDGA and MPC controllers. Initially, using PIDGA and MPC controllers the synthetic control is 94.0459 rad/s^2 and 88.3573 rad/s^2 respectively. then, goes to -28.8723 rad/s^2 and -23.1317 rad/s^2. Finally, after reach to the desired the synthetic control converges to 0. In figure 5.11 the synthetic control for second link of robot using three different controllers is presented.

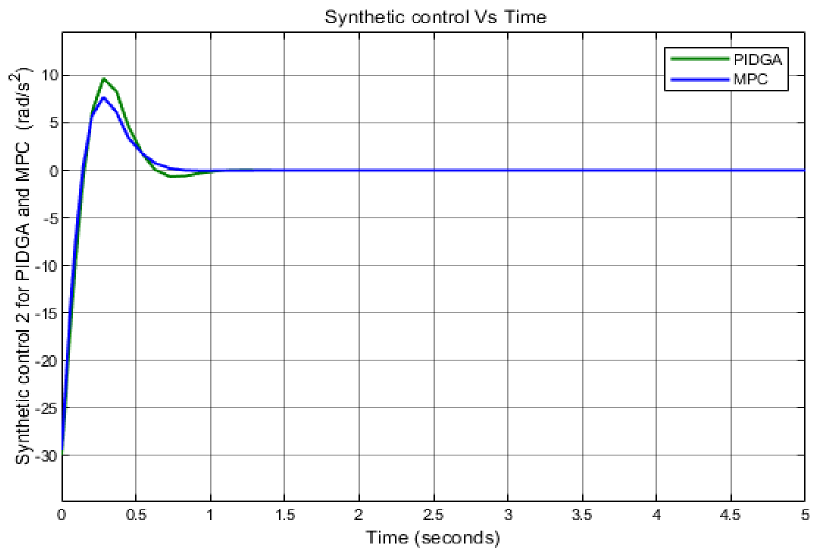


Figure 5.11 Synthetic control of 2nd link of robotic manipulator using PIDGA and MPC controller

The synthetic control for the second link of robotic manipulator in figure 5.11 using PIDGA and MPC controller is -29.9132 rad/s^2 and -29.4524 rad/s^2 respectively and then, goes to 9.6662 rad/s^2 and 7.7106 rad/s^2. Finally, after reach to the desired the synthetic control converges to 0. In figure 5.12 the synthetic control for three different controllers is presented.

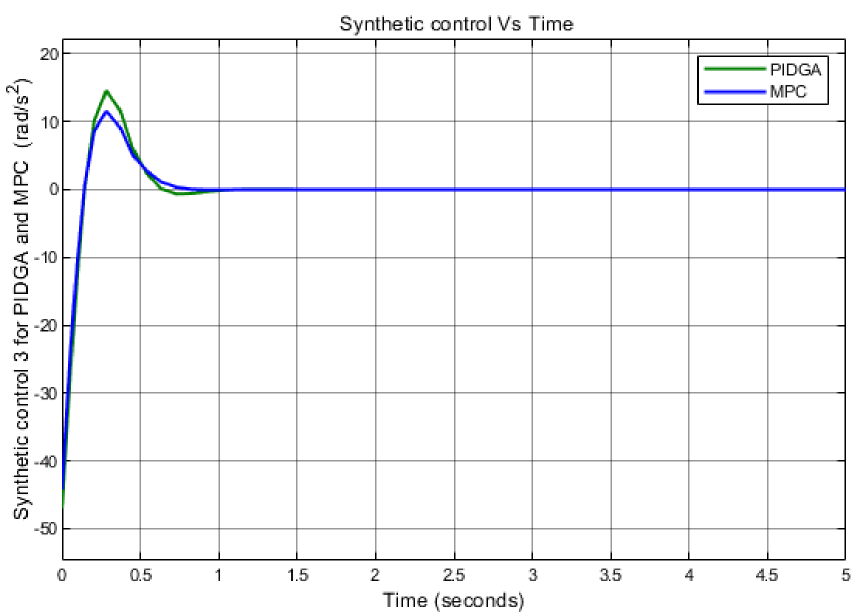


Figure 5.12 Synthetic control of 3rd link of robotic manipulator using PIDGA and MPC controller

In Figure 5.10, 5.11 and 5.12 graphical appearance and analyzed data, for two different controllers the synthetic control vectors $\chi_1,\ \chi_2\ and\ \chi_3$ converge to zero after certain time that the end effector reaches the desired set point. The synthetic control in figure 5.12 for the third link of robotic manipulator using PIDGA and MPC controller is -46.999 rad/s^2 and -44.1786 rad/s^2 respectively and then, goes to 14.5588 rad/s^2 and 11.5658 rad/s^2. Finally, after reach to the desired the synthetic control converges to 0. But the blue color, which is controlled by proposed model predictive control quickly return to zero with less overshoot.

where χ is a synthetic control which is the output of controller. After the robot reach to the desired set point, then the synthetic control went to the zero. Angular velocity for figure 5.13, 5.14 and 5.15 are illustrated below.

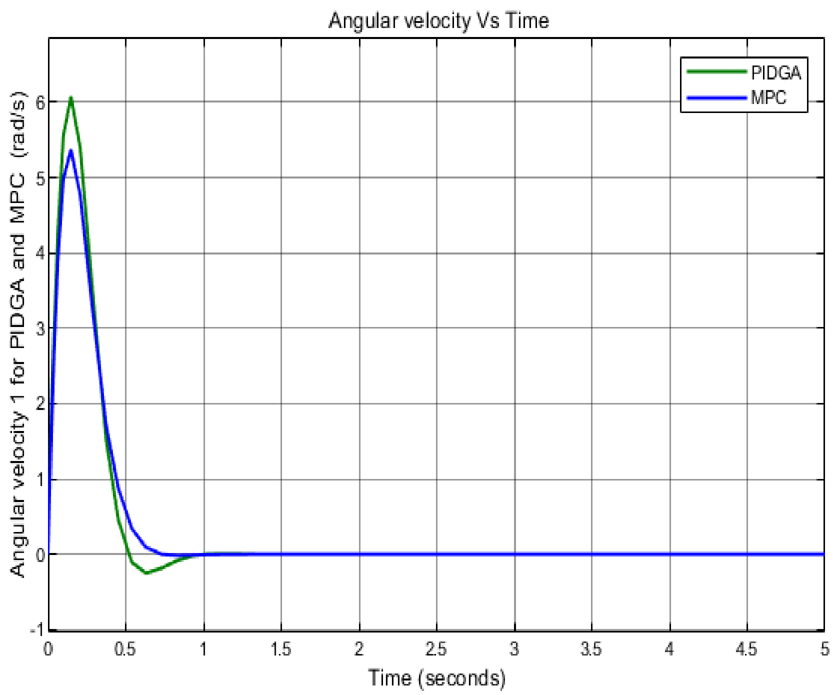


Figure 5.13 Angular velocity of 1st link of robotic manipulator using PIDGA and MPC controller

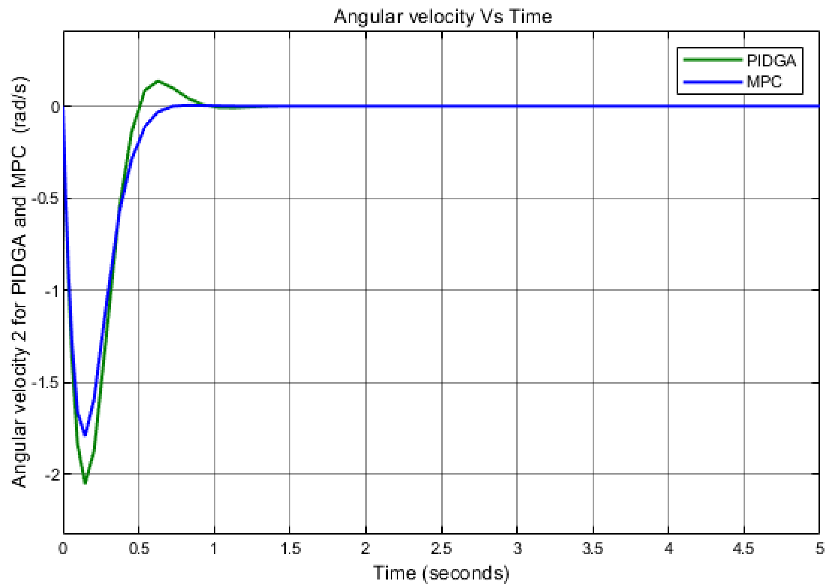


Figure 5.14 Angular velocity of 2nd link of robotic manipulator using PIDGA and MPC controller

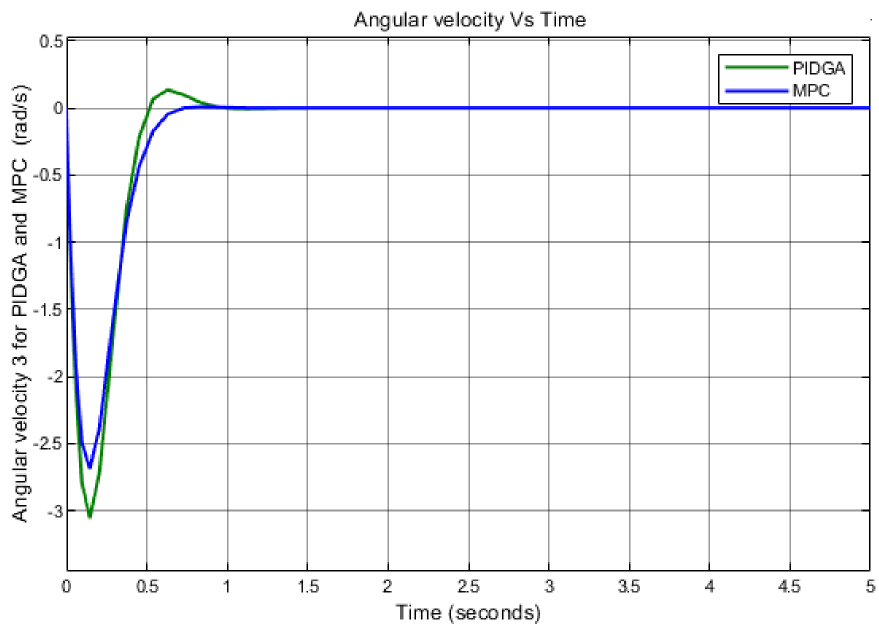


Figure 5.15 Angular velocities of 3rd link of robotic manipulator using PIDGA and MPC controller

In Figure 5.13, 5.14 and 5.15 the Angular velocities $\dot{\theta}_1,\ \dot{\theta}_2\ and\ \dot{\theta}_3$ converge to zero after the end effector reaches the desired set point. The angular velocity vector in Figure 5.13, 5.14 and 5.15, for the first, second and third link of robotic manipulator using MPC controller gave a good result and converge quickly to the desired set point. Finally, the desired the angular velocity converges to 0. here, the blue color, which is controlled by proposed model predictive control quickly return to zero. Thus, MPC controller is more efficient than PIDGA controller.

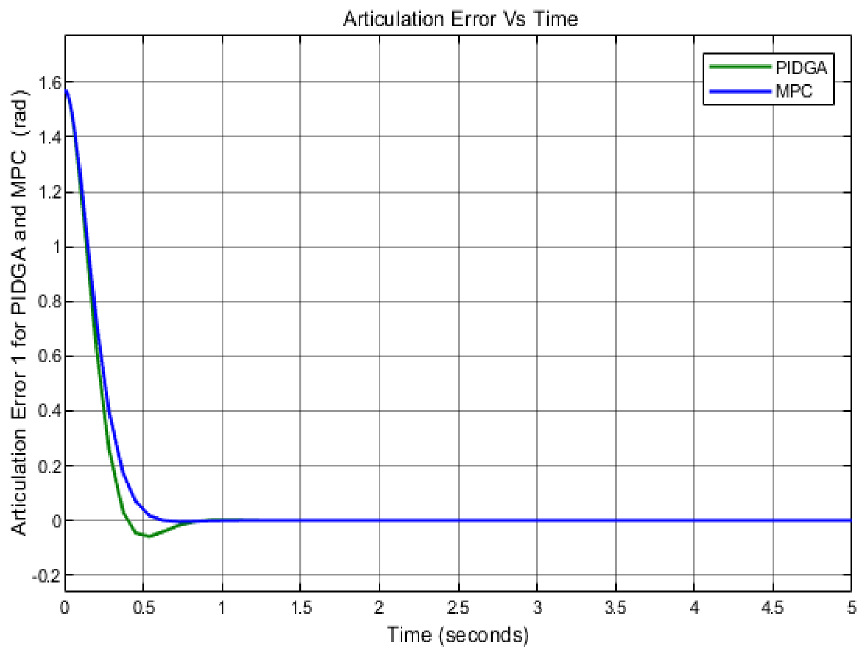


Figure 5.16 Articulation error of 1st link of robotic manipulator using PIDGA and MPC controller

From the Figure 5.16, the joint angular position error of the first link of robotic manipulator is converged quickly in to zero with small overshoot using the proposed approach. Initially, the error started from 1.5708 rad and finally converge to zero. In case of PIDGA, it has overshoot and the error started from 1.5708 rad and finally converge to -0.00001 rad.

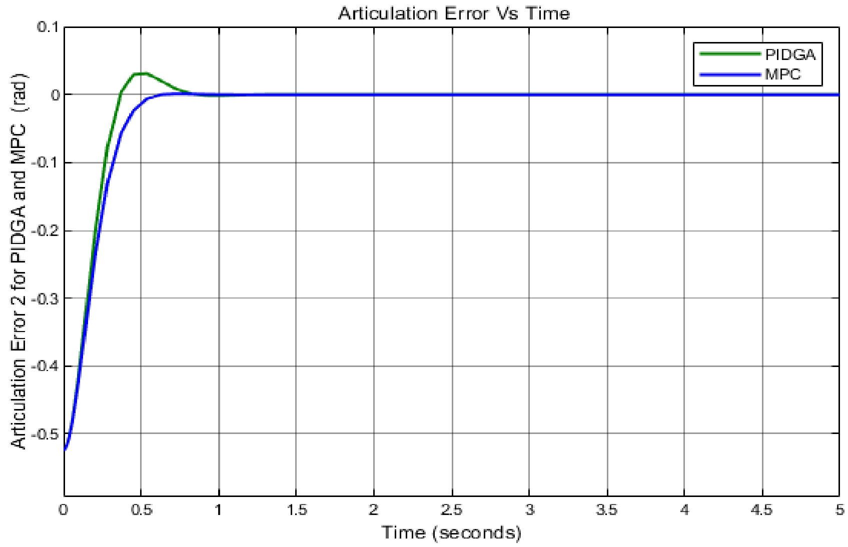


Figure 5.17 Articulation error of 2nd link of robotic manipulator using PIDGA and MPC controller

The articulation error for second link of robot in PIDGA and MPC controller converge to zero. At starting, the error begins from -0.5236 rad and finally converge to zero for all controllers. But specifically, the error converges to almost 0 rad and 0.0004 rad for MPC and PIDGA respectively. In Figure 5.18, the error for third link of robot manipulator is illustrated using PIDGA and MPC controllers.

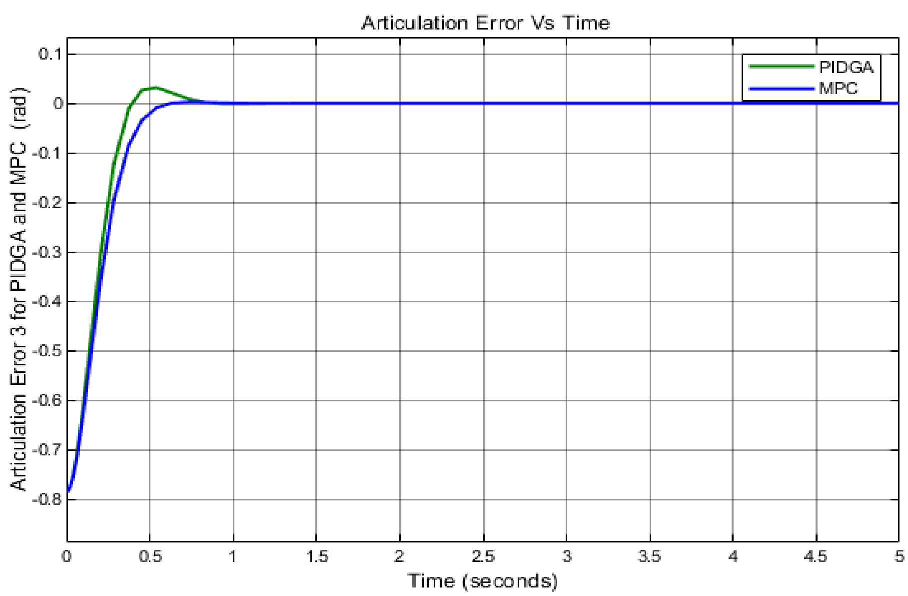


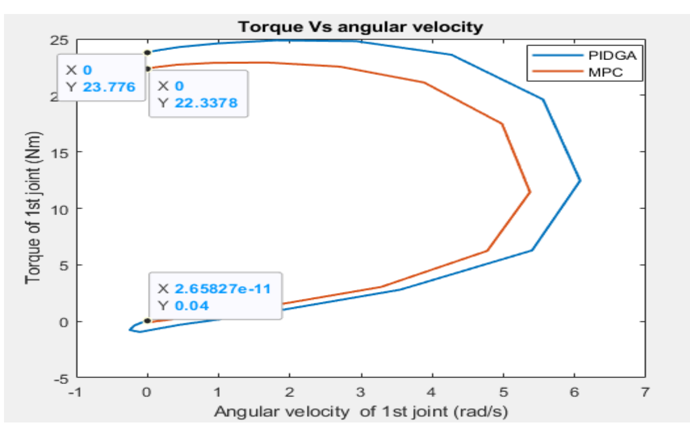
Figure 5.18 Articulation error of 3rd link of robotic manipulator using PIDGA and MPC controller

The articulation error for 3-DOF articulated robotic manipulator for each link of joint is converge to zero. In Figure 5.18, the error for MPC and PDGA is almost near to 0 rad and 0.0002 rad respectively. The angular error in figure 5.16, 5.17 and 5.18 using MPC controller gave a good result and converge quickly to zero. Thus, based on the result obtained the MPC controller is more error reduced than PIDGA controller.

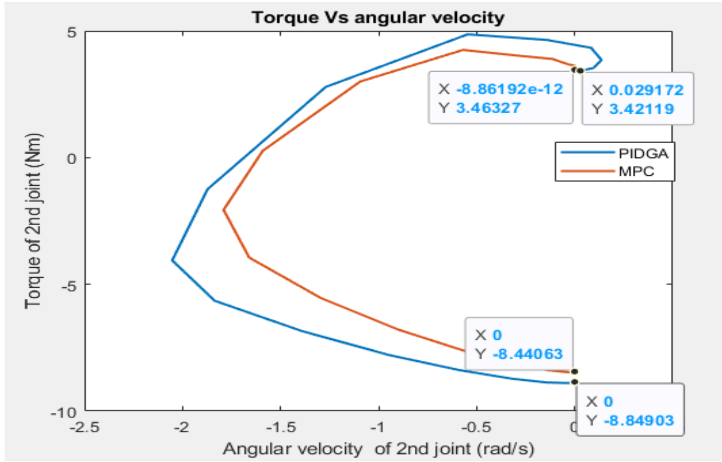
Table 5.9: Angular position error value using PIDGA and MPC Controllers at final value

Error for joint variables	PIDGA	MPC
Error at joint 1	-0.00001	Almost 0
Error at joint 2	0.0004	Almost 0
Error at joint 3	0.0002	Almost 0

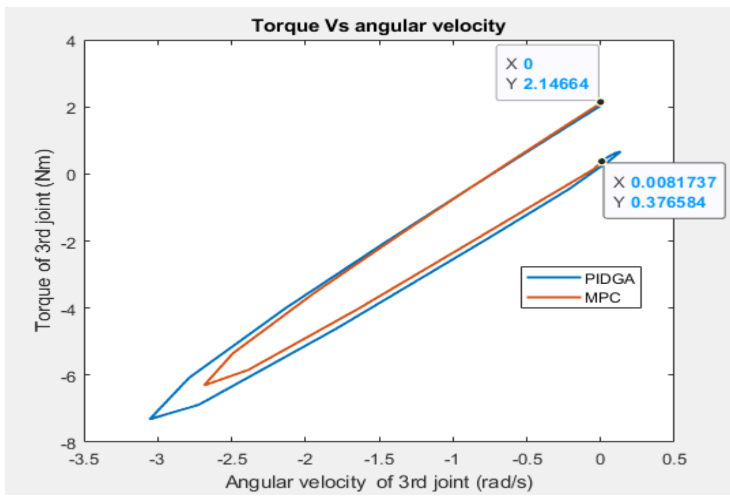
In Figure 5.19, the torque Vs angular velocity for both MPC and PIDGA is presented. At angular velocity 3 rad/s in (a) the torque that used in MPC is 22.1420 Nm and 2.6340Nm and PIDGA is 24.5561 Nm and 2.46742 Nm for two different time. Therefore, MPC is uses lower energy than PIDGA controllers. The reason why at fixed angular velocity became two different torque, the curve graph indicates the moment of inertia is dependent on the position of an object. The slope is changed and both torque and angular velocity graph is non minimum phase and one follow the others. At angular velocity 0 rad/s in (b) the torque that used in MPC is -8.44063 Nm and PIDGA is -8.84903 Nm. Therefore, MPC uses lower energy than PIDGA controllers. At angular velocity 0 rad/s in (c) the torque that used in MPC and PIDGA is 2.14664 Nm. But, the range of torque for MPC is lower than PIDGA.



(a) Torque Vs Angular velocity of 1st joint using MPC and PIDGA



(b) Torque Vs Angular velocity of 2nd joint using MPC and PIDGA



(c) Torque Vs Angular velocity of 1st joint using MPC and PIDGA

Figure 5.19 Torque vs Angular velocity for MPC and PIDGA of 3-DOF

The proposed MPC Controller can track also the sine wave input and ramp function input without considering. In figure 5.20 sine wave tracking using MPC controller. From the result obtained MPC can handle constraint and use lower consumption of energy to minimize the objective function and gave good result performance for 3-DOF articulated robotic manipulator. From the figure below, the blue color is MPC controller output that track the red desired sinusoidal wave.

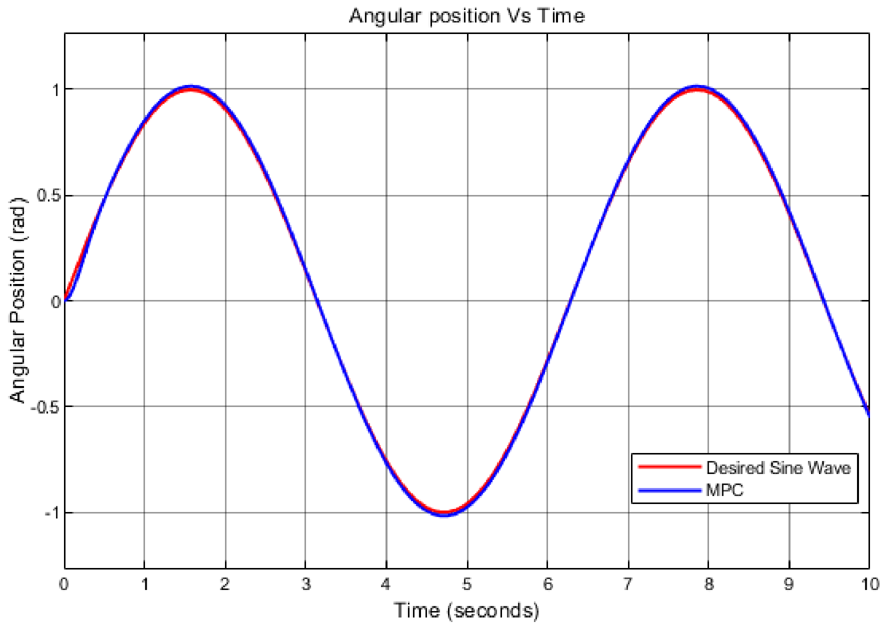


Figure 5.20 Sine wave input using MPC controller for 3-DOF articulated robotic arm.

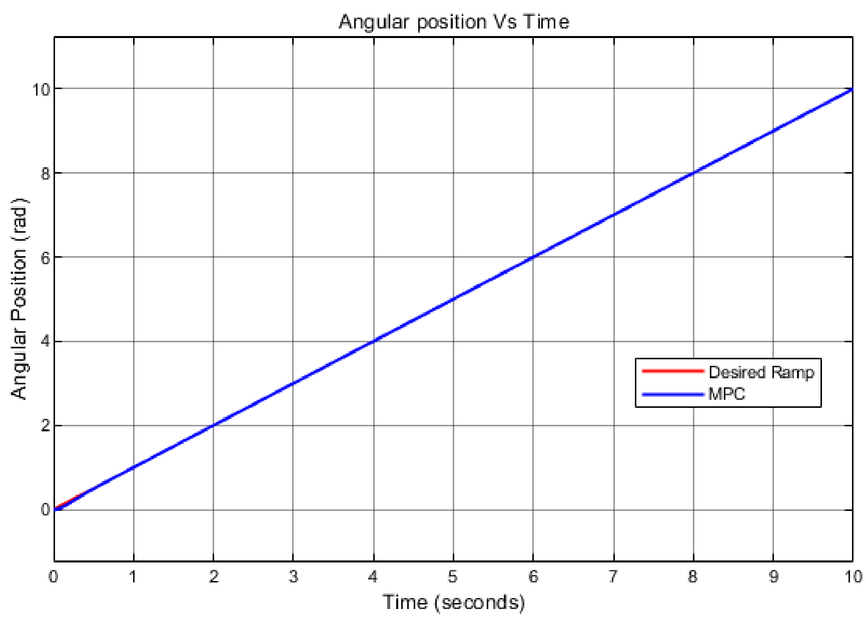


Figure 5.21 ramp function input using MPC controller for 3-DOF articulated robot

It can also track ramp input function. Here, for simulation time 10 sec, an output track ramp response using MPC controllers without disturbance is performed. In above figure, the red color is the reference (desired input) which is ramp input and the blue one is the proposed MPC optimal control that track the given input trajectory.

5.3 Disturbance rejection capability of controllers

Industrial articulated robotic manipulators are worked with hazardous environment and when it worked, vibration torque noise and external payload torque disturbance happened on the system. Therefore, in this thesis payload and vibration torque noise are considered. White noise power for band limit, sampling time seed are 0.000002, 0.001 and 100 respectively. In this chapter the proposed controller is tested in terms of rejecting sum up of payload variation and vibration torque.

The disturbance torque due to the combined of two disturbance uncertainty are illustrated in chapter 3 and used in the system model. Initially no payload variation up to 5 secs and then at 5 secs in the system model, payload varying torque up to 5.5 secs is considered. In this range, the controller's capability of rejecting disturbance is observed. Additionally, the amplitude that considered for payload varying effect is 8 Nm.

In Figure 5.22, 5.23 and 5.24, the angular position for first, second and third link of industrial articulated robotic manipulator is presented graphically. Initially the controllers for considering disturbance in the plant increased the overshoot of each link of robotic manipulator compared to the system without disturbance.

From the graph, the proposed controller converges with small overshooting and reaches quickly to the desired point. Thus, from the result that obtained MPC controller has rejection capability and good performance.

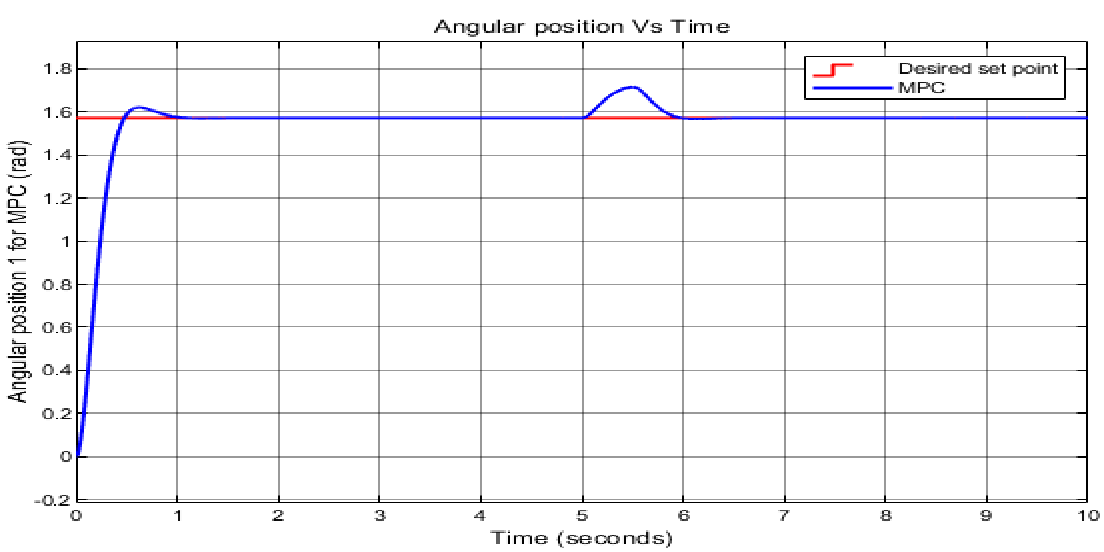


Figure 5.22 Angular position 1 under external disturbance using MPC

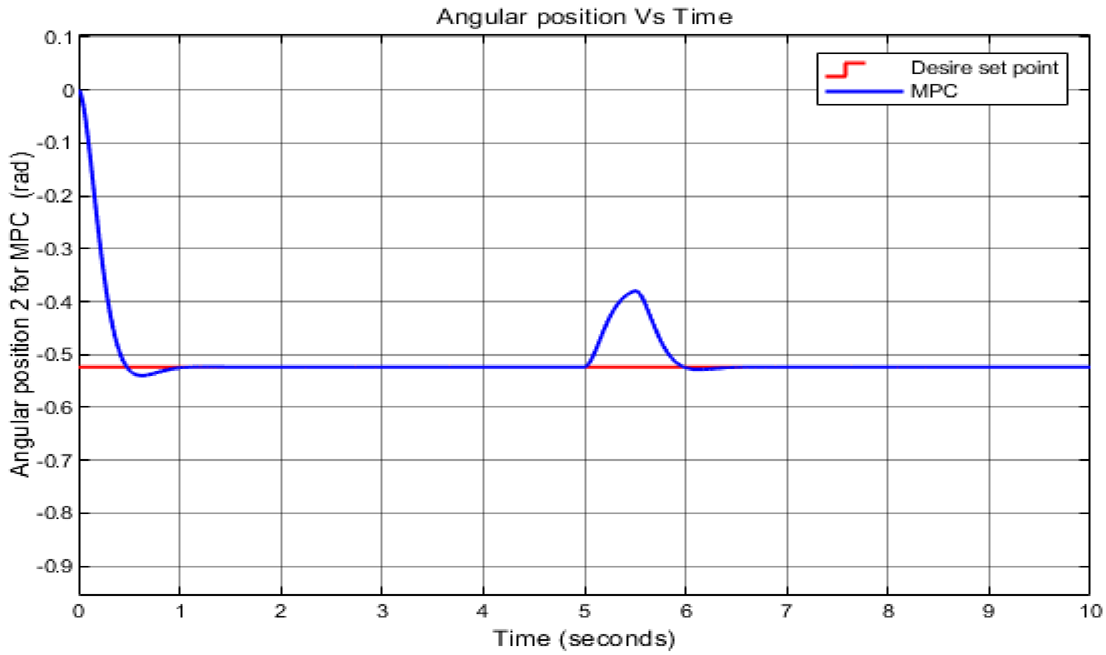


Figure 5.23 Angular position 2 under external disturbance using MPC controllers

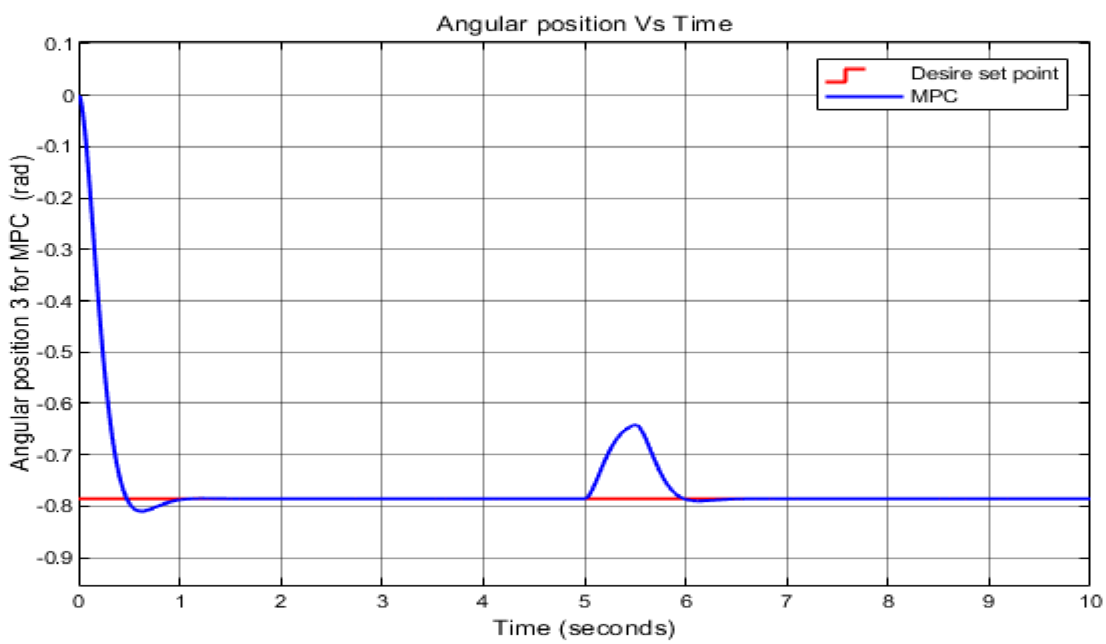


Figure 5.24 Angular position 3 under external disturbance using MPC controllers

From figure 5.25, 5.26 and 5.27, from the obtained result, using an MPC controller for the plant dynamics with considering external disturbance used low energy. Therefore, in order to control this nonlinear system, the proposed MPC controller is one of the best controller to improve it. Because the consumption of energy needed by the motor is low.

In figure 5.25 when disturbance happened at some time it uses low energy to control and back to normal stability of the system.

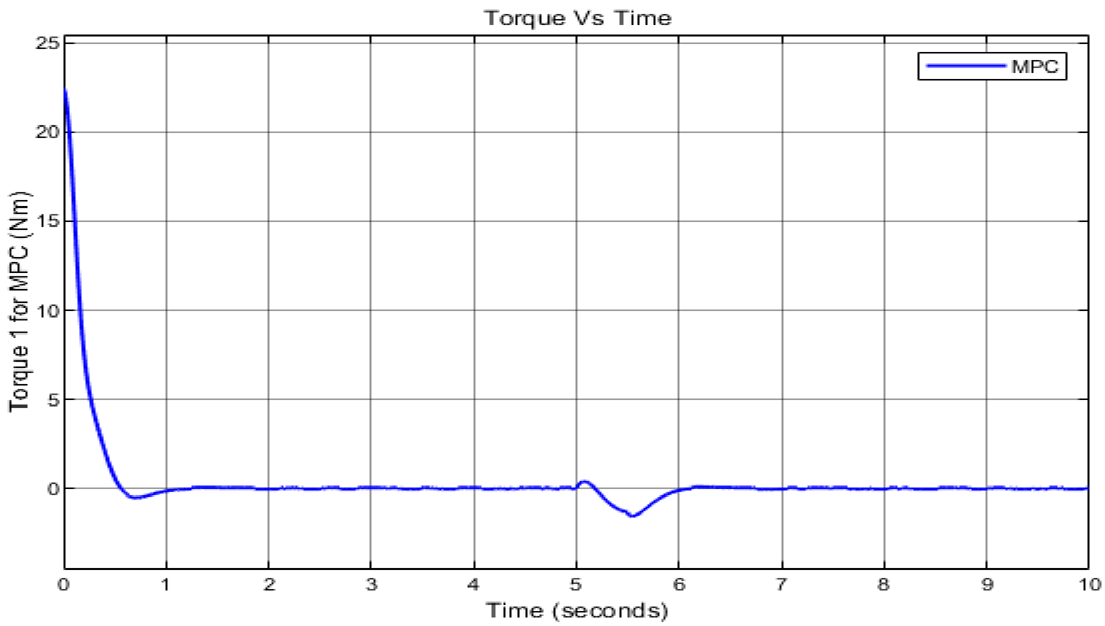


Figure 5.25 Control input torque 1 under external disturbance using MPC controllers

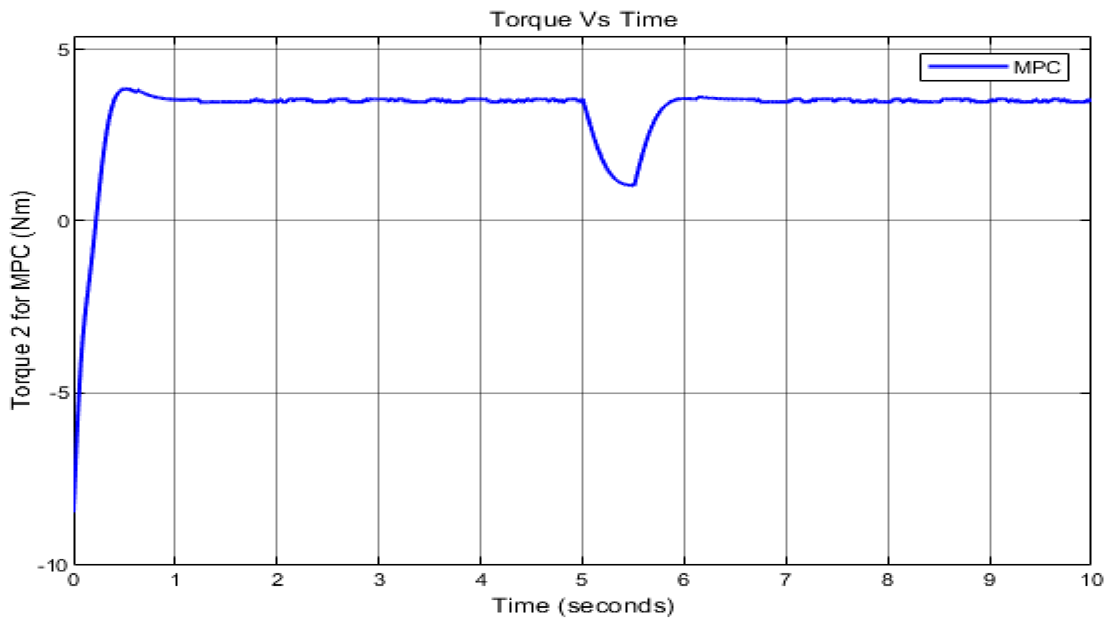


Figure 5.26 Control input torque 2 under external disturbance using MPC controllers

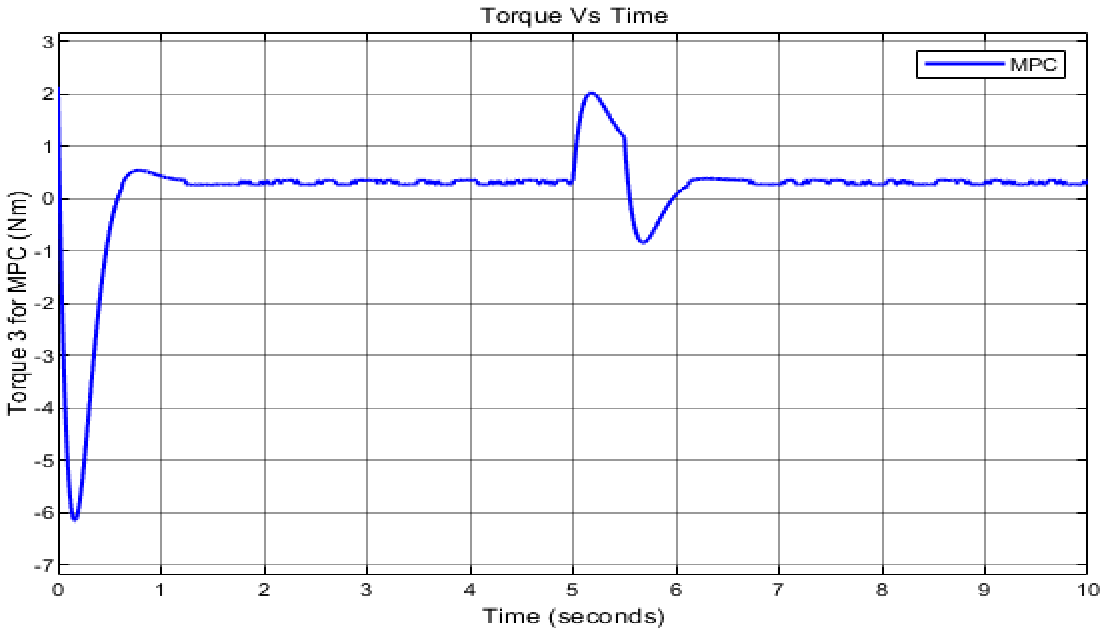


Figure 5.27 Control input torque 3 under external disturbance using MPC controllers

Figure 5.28, 5.29 and 5.30 show the plot of articulation tracking errors under external disturbance for MPC controllers. The results showed that the rejection capability of controlling the system and the performance angular position of each link with each joint using proposed controllers. MPC showed a high capability in controlling and handling the newly introduced external disturbance. MPC is also robust and performs well even in the presence of external disturbances with minimum tracking error.

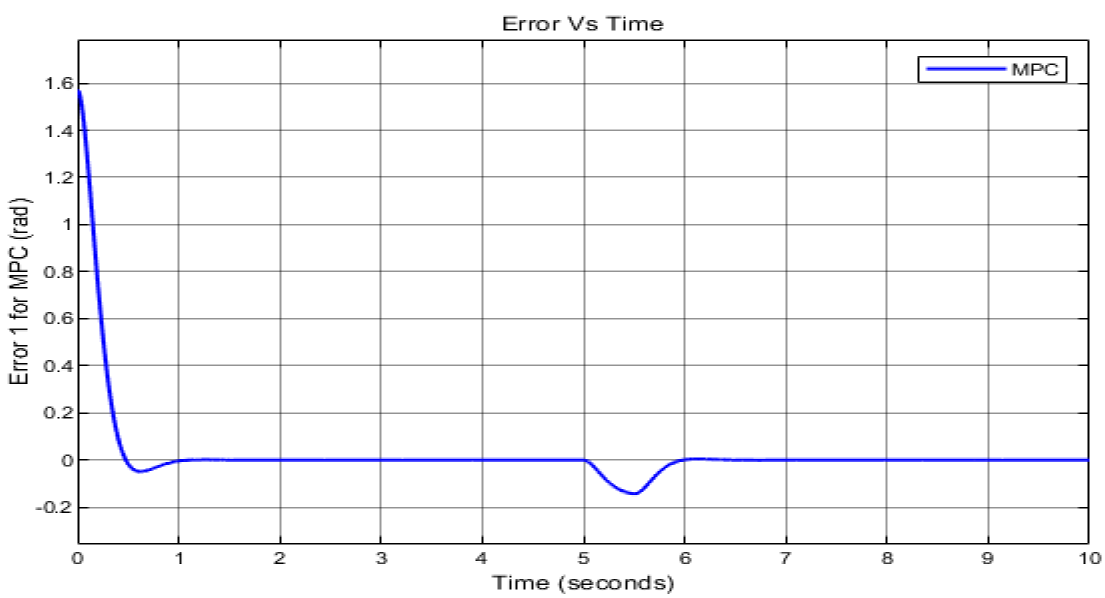


Figure 5.28 Articulation error 1 under external disturbance using MPC controllers

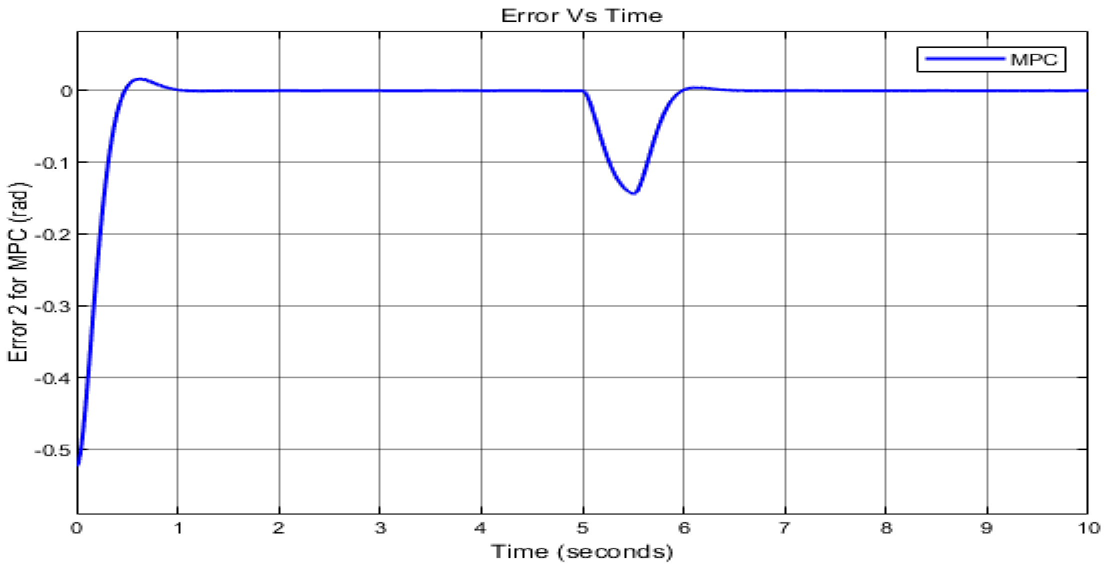


Figure 5.29 Articulation error 2 under external disturbance using MPC controllers

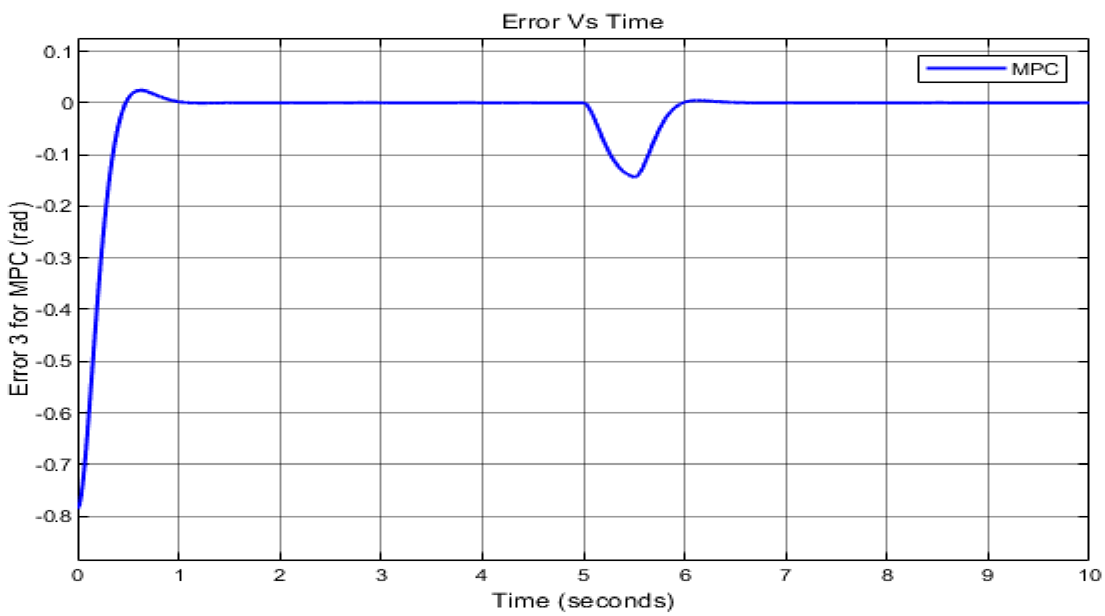


Figure 5.30 Articulation error 3 under external disturbance using MPC controller

6. CONCLUSION AND RECOMMENDATION

6.1 Conclusions

Robotic manipulators are very unstable and nonlinear systems. A controller is required for a set of tasks on an industrial 3-DOF articulated robotic manipulator. As a result, a model predictive controller for tracking robot articulation is proposed. A three-degree-of-freedom articulated robotic arm's kinematics and dynamics modeling has been accomplished. The model predictive control rule with plants in MATLAB/Simulink is interfaced and simulated, and the performance of the designed MPC control was evaluated using PIDGA. This study presents a model predictive control law for industrial articulated manipulator robot of three degrees of freedom. This solution begins using feedback linearization technique of converting the robot's nonlinear dynamic model into linear model. The weight factor and prediction horizon time are tuned based on torque energy constraint calculation method. MPC controllers were tested for disturbance capabilities and obtained good result. In this thesis natural frequency is chosen, which values have 7.5 rad/sec because since it is optimal, it is bounded by PIDGA output. For this natural frequency prediction horizon is 0.13925 and weighting factor is 5.6329e-04.

To attain the desired set point, the parameters of the proposed controller are tweaked using the energy input constraint analytic solution. Based on the results, the comparison is performed in terms of minimize error, rapid convergence, constraint management, and over shoot of plant dynamics to assess how effective the proposed controller is. For the 1st, 2nd and 3rd articulation, the rise time for PIDGA and MPC controller is 0.2527 sec, 0.2410 sec,0.2500 sec and 0.3168 sec, 0.3168 sec, 0.3168 sec respectively. Therefore, in rise time PIDGA has good performance than MPC controller. However, in Settling time for the 1st, 2nd and 3rd articulation for MPC controller is 0.5161 sec. The settling time for PIDGA for three respective articulations are 0.6608, 0.7101 and 0.6677. Therefore, from this MPC has better settling time performance than PIDGA. The maximum overshoot happened to control the 1st, 2nd and 3rd articulation for PIDGA and MPC controller is 3.7214%, 5.9657%, 4.0252% and 0.275%, 0.275%, 0.275% respectively. Thus, MPC has low over shoot than PIDGA controller. In order to track the desired set point, energy is required. For the 1st, 2nd and 3rd, the maximum energy required to track the desired signal for PIDGA and MPC is 24.8649 Nm, -8.8917 Nm, -7.3159 Nm and 22.9040 Nm, -8.4805 Nm, -6.3049 Nm. Therefore, for the same tracking process MPC controller used lower energy than PIDGA controller. The articulation error of PIDGA controller for 1st, 2nd and 3rd is 0.00001, 0.0004, 0.0002 and for MPC controller it is almost 0. After the robot end effector reach to the desired set point, the angular velocity and synthetic control is converged to 0. The proposed controller also tracked the sinewave and ramp input trajectory. The MPC control also handle the system in under external disturbance. The rise time of MPC controller in under disturbance case for 1st, 2nd and 3rd is 0.3006 sec, settling time is 5.8066 sec, 5.8958 sec, 5.8682 and overshoot is 9.1653%, 3.1043% and 3.1079%. when disturbance occurred in 5 - 5.5 sec, MPC converges to the normal steady operating condition.

Finally, compared to PIDGA controller, based on settling time result MPC has higher convergence rate, can manage input constraints, tracks the target signal with minimum error, used lower energy, and small overshoot.

6.2 Recommendations

In recent years, robot systems have taken over the world, and one of the issues in robotics systems is robotic manipulator. As a result of this concept, I advocate that this 3-DOF industrial articulated robotic manipulator be implemented in hardware employing model predictive control and real-time operation. Furthermore, by increasing the number of degrees of freedom of the robot, the trajectory tracking of the robotic manipulator can be controlled.

Future work: The next step will be to apply this optimal control law to a large number of degrees of freedom on a robot by adding additional controller in addition to PIDGA while taking actuator dynamics into account, and then to test the control law. Finally, real-time hardware implementation validation should be attempted.

Acknowledgments: This research thesis was funded by Adama Science and Technology University under the grant number of ASTU/SM-R/562/22, Adama, Ethiopia.

REFERENCES

Agbaraji, E., Inyiama, H., & Okezie, C. (2017). Dynamic Modeling of a 3-DOF Articulated Robotic Manipulator Based on Independent Joint Scheme. *Physical Science International Journal*, *15*(1), 1–10. https://doi.org/10.9734/psij/2017/33578

Ashagrie, A., Salau, A. O., & Weldcherkos, T. (2021). Modeling and control of a 3-DOF articulated robotic manipulator using self-tuning fuzzy sliding mode controller. *Cogent Engineering*, *8*(1). https://doi.org/10.1080/23311916.2021.1950105

Ashi, M. M. Al. (2014). Trajectory Tracking Control of A 2-DOF Robot Arm Using Neural Networks. *Thesis, February*.

Bemporad, A. (2006). Model Predictive Control design: New trends and tools. *Proceedings of the IEEE on Decision and Control 1*, 6678–6683. https://doi.org/10.1109/cdc.2006.377490

Billard, A., & Kragic, D. (2019). Trends and challenges in robot manipulation. *Science*, *364*(6446). https://doi.org/10.1126/science.aat8414

Bouzoualegh, S. (2019). Model Predictive Control of a Three Degrees of Freedom Manipulator Robot. *2019 International Conference on Advanced Systems and Emergent Technologies (IC_ASET)*, 84–89.

Chang, Y. C., Yen, H. M., & Wu, M. F. (2008). An intelligent robust tracking control for electrically-driven robot systems. *International Journal of Systems Science*, *39*(5), 497–511. https://doi.org/10.1080/00207720701832747

Cisneros, P. S. G., Sridharan, A., & Werner, H. (2018). Constrained Predictive Control of a Robotic Manipulator using quasi-LPV Representations. *IFAC-PapersOnLine*, *51*(26), 118–123. https://doi.org/10.1016/j.ifacol.2018.11.158

Darajat, A. U., & Istiqphara, S. (2020). Control of Two-Link Robot Manipulator with Uncertainty Parameter using Self-Tuning Sliding Mode Control. *MECnIT 2020 - International Conference on Mechanical, Electronics, Computer, and Industrial Technology*, 376–380. https://doi.org/10.1109/MECnIT48290.2020.9166643

Dereje, G. (2018). *Sliding Mode Control of a Two Dergee of Freedom Robot Arm Using Permanent Magnet Synchronous Motor*.

Ehsan, S. (2010). Sliding Mode Control of Robot Manipulators via Intelligent Approaches. *Advanced Strategies for Robot Manipulators*, 136–170. https://doi.org/10.5772/10193

Elkhateeb, N. A., & Badr, R. I. (2017). Novel PID Tracking Controller for 2DOF Robotic

Manipulator System Based on Artificial Bee Colony Algorithm. *Electrical, Control and Communication Engineering*, *13*(1), 55–62. https://doi.org/10.1515/ecce-2017-0008

Fateh, M. M. (2008). International Journal of Control, Automation, and Systems. *Springer*, *6*(5), 702–712.

Ghaleb, N. M., & Aly, A. A. (2018). Modeling and Control of 2-DOF Robot Arm. *International Journal of Emerging Engineering Research and Technology*, *6*(11), 24–31.

Guechi, E. H., Bouzoualegh, S., Zennir, Y., & Blažič, S. (2018). MPC control and LQ optimal control of a two-link robot arm: A comparative study. *Machines*, *6*(3), 1–14. https://doi.org/10.3390/machines6030037

Hsia, T. C., & Gao, L. S. (1990). *Robot manipulator control using decentralized linear time-invariant time-delayed controllers*. 2070–2075. https://doi.org/10.1109/robot.1990.126310

Incremona, G. P., Ferrara, A., & Magni, L. (2017). MPC for robot manipulators with integral sliding modes generation. *IEEE/ASME Transactions on Mechatronics*, *22*(3), 1299–1307. https://doi.org/10.1109/TMECH.2017.2674701

Incremona, G. P., Strada, S. C., Tanelli, M., & Ferrara, A. (2019). Switched adaptation strategies for integral sliding mode control: Theory and application. *International Journal of Robust and Nonlinear Control*, *29*(17), 6064–6080. https://doi.org/10.1002/rnc.4716

Iordanou, H. N., & Surgenor, B. W. (1996). Experimental Evaluation of the Robustness of Discrete Sliding Mode Control Versus Linear Quadratic Control. *Brain and Cognition*, *32*(2), 273–344. https://doi.org/10.1006/brcg.1996.0066

Iqbal, J. (2019). Modern Control Laws for an Articulated Robotic Arm: Modeling and Simulation. *Engineering, Technology & Applied Science Research*, *9*(2), 4057–4061. https://doi.org/10.48084/etasr.2598

Jin, L., Li, S., Yu, J., & He, J. (2018). Robot manipulator control using neural networks: A survey. *Neurocomputing*, *285*, 23–34. https://doi.org/10.1016/j.neucom.2018.01.002

Johansson, R. (1990). Adaptive Control of Robot Manipulator Motion. *IEEE Transactions on Robotics and Automation*, *6*(4), 483–490. https://doi.org/10.1109/70.59359

K. Kawamura, S. M. G. and P. R. (2007). Industrial Robotics Theory, Modelling and control. In *Journal of the Institution of Engineers (India), Part PR: Production Engineering Division* (Vol. 66). https://doi.org/10.1017/cbo9780511712173.002

Kim, J., Jin, M., Park, S. H., Chung, S. Y., & Hwang, M. J. (2020). Task space trajectory planning for robot manipulators to follow 3-d curved contours. *Electronics (Switzerland)*, *9*(9), 1–20. https://doi.org/10.3390/electronics9091424

Korayem, M. H., Alamdari, A., Haghighi, R., & Korayem, A. H. (2010). Determining maximum load-carrying capacity of robots using adaptive robust neural controller. *Robotica*, *28*(7), 1083–1093. https://doi.org/10.1017/S0263574710000081

L1, C. J. (1988). *A New Method of Dynamics for Robot Manipulators*. *18*(1), 105–114.

Mahil, M. S., & Al-durra, A. (2016). Modeling Analysis and Simulation of 2-DOF Robotic Manipulator. *IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, *October*, 16–19.

Man, K. F., Tang, K. S., & Kwong, S. (1996). Genetic algorithms: Concepts and applications. *IEEE Transactions on Electronics*, *43*(5), 519–534. https://doi.org/10.1109/41.538609

Massaoudi, F., Elleuch, D., & Damak, T. (2019). *Robust Control for a Two DOF Robot Manipulator*. *2019*.

Moe Myint, K., Min Min Htun, Z., & Myo Tun, H. (2016). Position Control Method For Pick And Place Robot Arm For Object Sorting System. *Position Control Method For Pick And Place Robot Arm For Object Sorting System*, *5*(6), 57–61. www.ijstr.org

Moldoveanu, F. (2014). Sliding Mode Controller Design for Robot Manipulators. *Bulletin of the Transilvania University of Braşov*, *7*(2).

Naing, W. W., Aung, K. Z., & Thike, A. (2018). Position Control of 3-DOF Articulated Robot Arm using PID Controller. *International Journal of Science and Engineering Applications*, *7*(09), 254–259.

Obasi, C. C., Braimoh, I. A., Odaba, A., Ogbewey, L. I., & Oluyomi, A. (2020). *Dynamics of 3 – Links Articulated Robotic Manipulator : A Computational Model*. *3*, 1911–1915. https://doi.org/10.35940/ijeat.C5370.029320

Önkol, M., & Kasnakoğlu, C. (2018). Adaptive model predictive control of a two-wheeled robot manipulator with varying mass. *Measurement and Control (United Kingdom)*, *51*(1–2), 38–56. https://doi.org/10.1177/0020294018758527

Pezeshki, S., Badalkhani, S., & Javadi, A. (2012). Performance analysis of a neuro-PID controller applied to a robot manipulator. *International Journal of Advanced Robotic Systems*, *9*, 1–10. https://doi.org/10.5772/51280

Piltan, F., Mehrara, S., Bayat, R., & Rahmdel, S. (2012). Design New Control Methodology of

Industrial Robot Manipulator : Sliding Mode Baseline Methodology. *International Journal of Hybrid Information Technology*, *5*(4), 41–54.

Raza, K., Khan, T. A., & Abbas, N. (2018). Kinematic analysis and geometrical improvement of an industrial robotic arm. *Journal of King Saud University - Engineering Sciences*, *30*(3), 218–223. https://doi.org/10.1016/j.jksues.2018.03.005

Smith, B. Y. C., & Christensen, H. I. (2009). *Robot Manipulators*. *December*, 75–83.

Tourassis, V. D., & Ang, M. H. (1992). Identification and analysis of robot manipulator singularities. *International Journal of Robotics Manipulator Research*, *11*(3), 248–259. https://doi.org/10.1177/027836499201100307

Voisembert, S., Riwan, A., & Mechbal, N. (2012). Numerical evaluation of a new robotic manipulator based on inflatable joints. *IEEE International Conference on Automation Science and Engineering*, 544–549. https://doi.org/10.1109/CoASE.2012.6386318

Yadmellat, P., Shafer, A. S., & Kermani, M. R. (2013). Design and development of a safe robot manipulator using a new actuation concept. *Proceedings - IEEE International Conference on Robotics and Automation*, 337–342. https://doi.org/10.1109/ICRA.2013.6630597

APPENDIXES

Appendixes A

Simulink block diagram

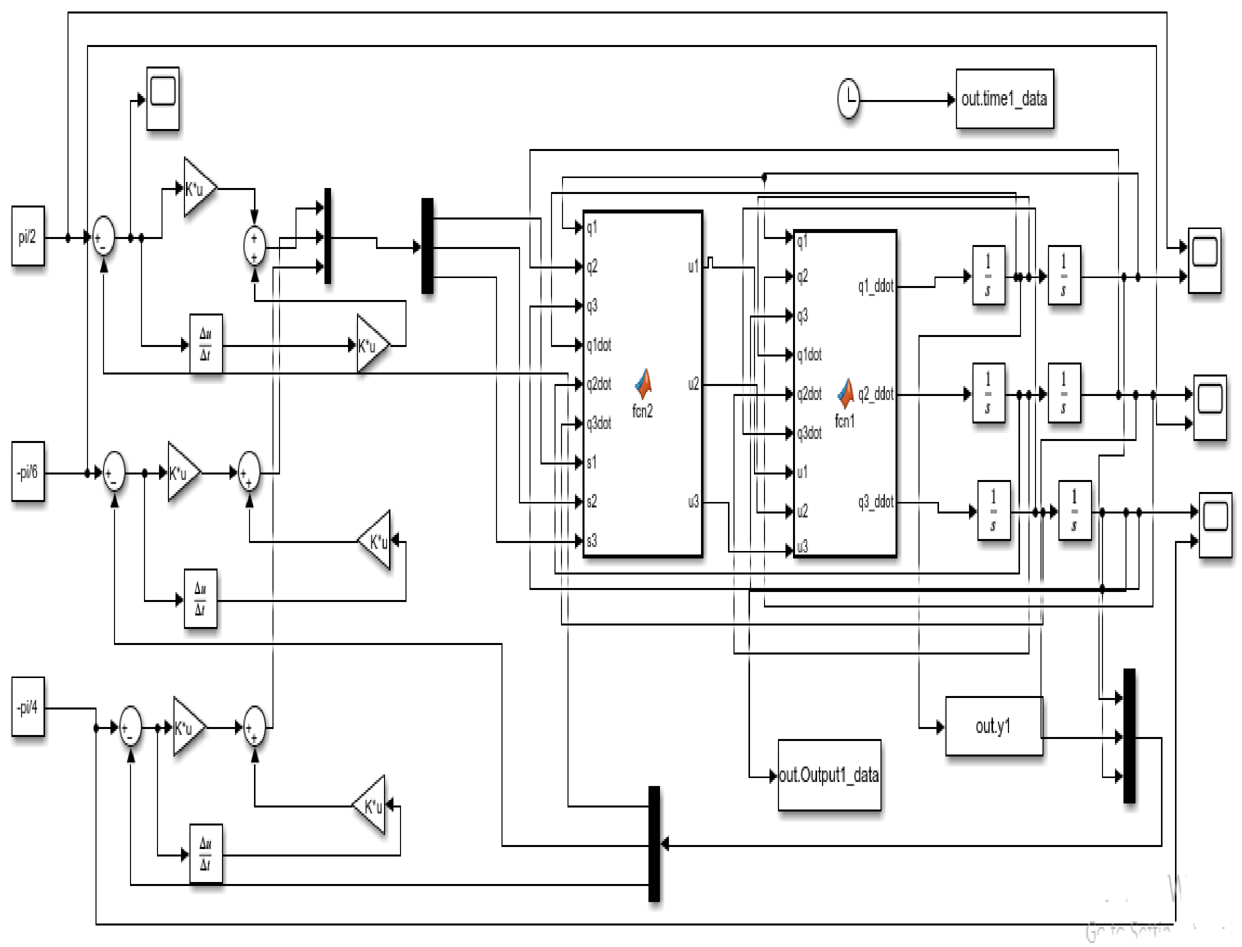


Figure A.1 Simulink Block for MPC

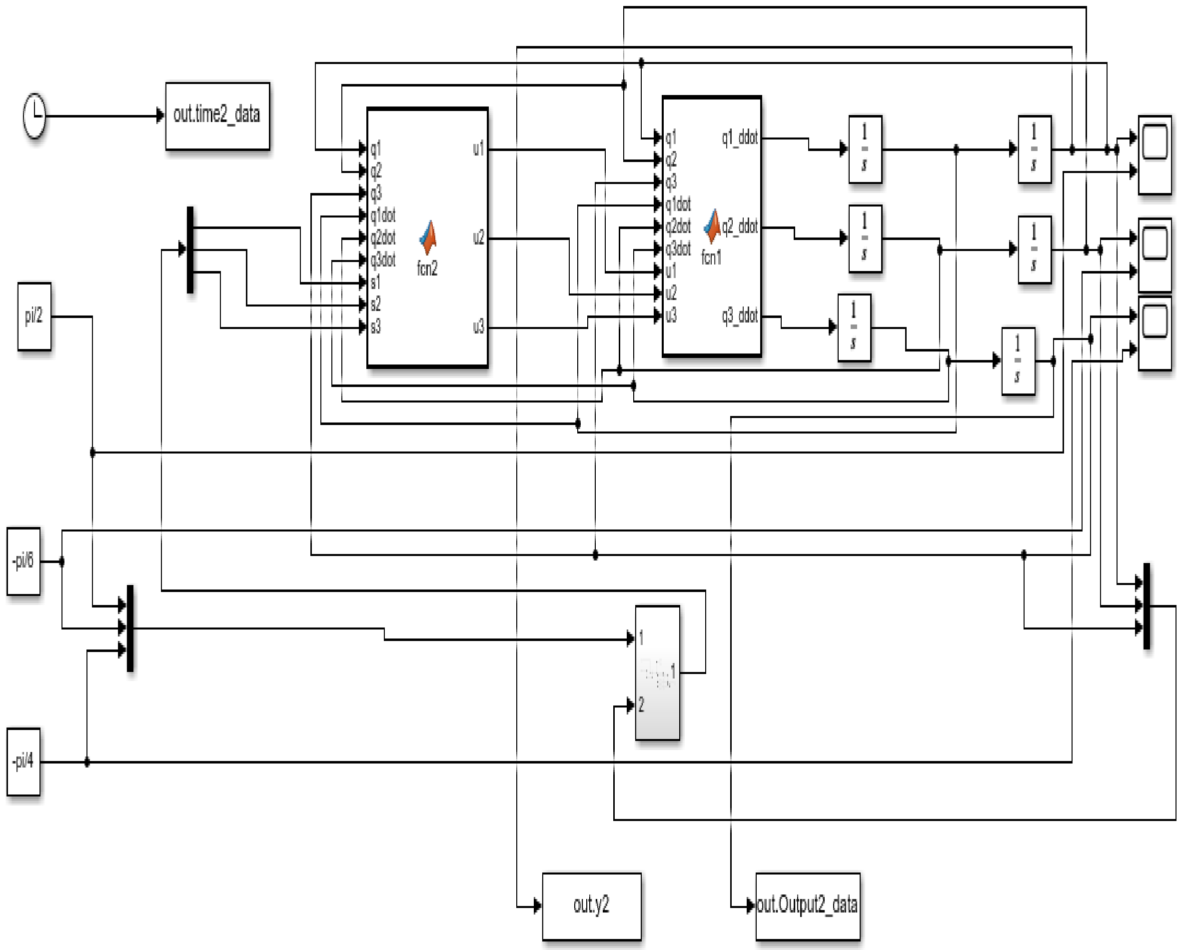


Figure A.2 Simulink block for PIDGA

Appendixes B

Optimization algorithm and code for nonlinear plant

```
% Tune the gains of the PID controller using a GA
function TuneRobotModel
%rng default % Initialize the random number generator
options= optimoptions('ga');
options.PlotFcn= 'gaplotbestf';
options.PopulationSize= 40;
options.MaxGenerations= 50;
nvar= 9;
LB= [0 0 0 0 0 0 0 0 0];
UB= [60 60 60 3 3 3 15 15 15];
[x,fval]= ga(@EvalObj,nvar,[],[],[],[],LB,UB,[],options)
%obj= EvalObj([55 55 55 3 3 3 15 15 15])
% Measures a performance index from the simulink model
function obj= EvalObj(x)
% Updates the values of Kp, Ki and Kd in Simulink model workspace
simulink_model= 'RobotModel';
load_system(simulink_model);
gains= get_param(simulink_model,'modelworkspace');
gains.assignin('Kp1', x(1));
gains.assignin('Kp2', x(2));
gains.assignin('Kp3', x(3));
gains.assignin('Ki1', x(4));
gains.assignin('Ki2', x(5));
gains.assignin('Ki3', x(6));
gains.assignin('Kd1', x(7));
gains.assignin('Kd2', x(8));
gains.assignin('Kd3', x(9));
```

```
% Simulates the simulink model to get the outputs
simOut= sim(simulink_model,'SaveOutput','on');
% Retrieves time t and error e
error= simOut.find('e');
t= error.Time;
e= error.Data;
% Calculate a performance index using the trapezoidal integration rule
n= length(e);
ae= abs(e(:,1))+abs(e(:,2))+abs(e(:,3));
obj= 0;
for i= 2:n
   stepsize= t(i)-t(i-1);
   obj= obj+0.5*stepsize*(ae(i-1)+ae(i)); % IAE
end
%lineariazation of input term to cancel nonlinearity
function [u1,u2,u3]=fcn2(q1,q2,q3,q1dot,q2dot,q3dot,aq1,aq2,aq3)
g=9.81;
r1=0.075;
d1=0.15;
a2=0.5;
a3=0.5;
m1=0.5;
m2=0.5;
m3=0.5;
m11=m1*r1^2+m2*a2^2*(cos(q2))^2/4+m3*a3^2*(cos(q2+q3))^2/4+m3*a2^2*(cos(q2))^2+m
2*a2*a3*cos(q2)*cos(q2+q3)/2;
m12=0;
m13=0;
m21=0;
m22=m2*a2^2/4+m3*a2^2+m3*a3^2/4+m3*a2*a3*cos(q3);
m23=m3*a2*a3*cos(q3)/2+m3*a3^2/4;
```

```
m31=0;
m32=-m3*a2*a3*cos(q3)/2-m3*a3^2/4;
m33=m3*a3^2/3;
M=[m11 m12 m13;m21 m22 m23;m31 m32 m33];
g1=0;
g2=m2*a2*g*cos(q2)/2+m3*a2*g*cos(q2)+m3*a3*g*cos(q2+q3)/2;
g3=m3*a3*g*cos(q2+q3)/2;
gvec=[g1;g2;g3];
b11=(-m2*a2^2*sin(2*q2)/4-m3*a2^2*sin(2*q2)-m3*a2*a3*sin((2*q2)+q3)-
m3*a2^2*sin(2*(q2+q3))/4)*q1dot*q2dot...
    +(-m3*a2*a3*cos(q2)*sin(q2+q3)-m3*a3^2*sin(2*(q2+q3))/4)*q1dot*q3dot;
b21=(-m2*a2*a3*sin(q3))*q2dot*q3dot;
b31=(m3*a2*a3*sin(q3)/2)*q2dot*q3dot;
b_qdot=[b11;b21;b31];
c11=2.66*q1dot+0.04*sign(q1dot);
c21=(m2*a2^2*sin(2*q2)/8+m3*a2^2*sin(2*q2)/2+a2*a3*m3*sin((2*q2)+q3)/2+m3*a3^2*sin(
2*(q2+q3))/8)*q1dot^2+...
    (-m3*a2*a3*sin(q3)/2)*q3dot+2.66*q2dot+0.04*sign(q2dot);
c31=(m3*a3*sin(q2+q3)*a2*cos(q2)/2+m3*a3*sin(q2+q3)*a3*cos(q2+q3)/4)*q2dot^2+2.66*q
3dot+0.04*sign(q3dot);
c_qdot=[c11;c21;c31];
u=M*[aq1;aq2;aq3]+b_qdot+c_qdot+gvec;
u1=u(1);
u2=u(2);
u3=u(3);
%nonlinear dynamic model of 3-DOF robotic manipulator
function [q1_ddot,q2_ddot,q3_ddot]=fcn1(q1,q2,q3,q1dot,q2dot,q3dot,u1,u2,u3)
g=9.81;
r1=0.075;
d1=0.15;
a2=0.5;
```

```
a3=0.5;
m1=0.5;
m2=0.5;
m3=0.5;
m11=m1*r1^2+m2*a2^2*(cos(q2))^2/4+m3*a3^2*(cos(q2+q3))^2/4+m3*a2^2*(cos(q2))^2+m2*a2*a3*cos(q2)*cos(q2+q3)/2;
m12=0;
m13=0;
m21=0;
m22=m2*a2^2/4+m3*a2^2+m3*a3^2/4+m3*a2*a3*cos(q3);
m23=m3*a2*a3*cos(q3)/2+m3*a3^2/4;
m31=0;
m32=-m3*a2*a3*cos(q3)/2-m3*a3^2/4;
m33=m3*a3^2/3;
M=[m11 m12 m13;m21 m22 m23;m31 m32 m33];
g1=0;
g2=m2*a2*g*cos(q2)/2+m3*a2*g*cos(q2)+m3*a3*g*cos(q2+q3)/2;
g3=m3*a3*g*cos(q2+q3)/2;
gvec=[g1;g2;g3];
b11=(-m2*a2^2*sin(2*q2)/4-m3*a2^2*sin(2*q2)-m3*a2*a3*sin((2*q2)+q3)-m3*a2^2*sin(2*(q2+q3))/4)*q1dot*q2dot...
    +(-m3*a2*a3*cos(q2)*sin(q2+q3)-m3*a3^2*sin(2*(q2+q3))/4)*q1dot*q3dot;
b21=(-m2*a2*a3*sin(q3))*q2dot*q3dot;
b31=(m3*a2*a3*sin(q3)/2)*q2dot*q3dot;
b_qdot=[b11;b21;b31];
c11=2.66*q1dot+0.04*sign(q1dot);
c21=(m2*a2^2*sin(2*q2)/8+m3*a2^2*sin(2*q2)/2+a2*a3*m3*sin((2*q2)+q3)/2+m3*a3^2*sin(2*(q2+q3))/8)*q1dot^2+...
    (-m3*a2*a3*sin(q3)/2)*q3dot+2.66*q2dot+0.04*sign(q2dot);
c31=(m3*a3*sin(q2+q3)*a2*cos(q2)/2+m3*a3*sin(q2+q3)*a3*cos(q2+q3)/4)*q2dot^2+2.66*q3dot+0.04*sign(q3dot);
```

```
c_qdot=[c11;c21;c31];
q2dot=inv(M)*([u1;u2;u3]-b_qdot-c_qdot-gvec);
q1_ddot=q2dot(1);
q2_ddot=q2dot(2);
q3_ddot=q2dot(3);
%forward kinematics
startup_rvc
%DH-Parameters using peter cork robotics
%Rhino XRRR
d1=0.15;
a2=0.5;
a3=0.5;
%create link using this code
%L=Link([Th d a alph])
L(1)=Link([0  d1   0     pi/2]);
L(2)=Link([0   0    a2     0]);
L(3)=Link([0   0   a3     0]);
Rob=SerialLink(L);
Rob.name='3-DOF XRRR';
theta1=0; theta2=0; theta3=0;
Rob.plot([theta1, theta2, theta3])
Rob.fkine([theta1, theta2, theta3])
%finding of forward kinematics at initial
Rob.fkine([0 0 0])
%finding of forward kinematics at final point
Rob.fkine([pi/2 -pi/6 -pi/4])
for theta1=0:0.01:pi/2
   Rob.plot([theta1 0 0]);
   pause(0.25)
end
for theta2=0:-0.01:-pi/6
```

```
    Rob.plot([pi/2 theta2 0]);
    pause(0.25)
end
for theta3=0:-0.01:-pi/4
    Rob.plot([pi/2 -pi/6 theta3]);
    pause(0.25)
end
%inverse kinematics
startup_rvc
%DH-Parameters using peter cork robotics
%Rhino XRRR
d1=0.15;
a2=0.5;
a3=0.5;
%create link using this code
%L=Link([Th d a alph])
L(1)=Link([0  d1   0     pi/2]);
L(2)=Link([0   0    a2     0]);
L(3)=Link([0   0   a3     0]);
Rob=SerialLink(L);
Rob.name='3-DOF XRRR';
T=[1 0 0 1;
     0 0 -1 0;
     0 1 0 0.15;
     0 0 0 1];
j=Rob.ikine(T, [0 0 0], [1 1 1 0 0 0])*180/pi
%time doain specification performance anaylsis
%x is 1 for  MPC and 2 for PIDGA
t=out.timex_data(:,1);
y=out.Outputx_data(:,1);
stepinfo(y,t,yfinal)
```