Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

Model Predictive Control Design for Agricultural Robot Operation in Row Culture

A Thesis Submitted to the Addis Ababa Institute of Technology in partial Fulfillment of the Requirement for the Degree of Master of Science in Control engineering

By

Asheberom Gebreslassie

Advisor: Dr. Dereje Shiferaw

Addis Ababa, Ethiopia

December, 2020

Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

(Control Engineering Stream)

Model Predictive Control Design for Agricultural Robot Operation in Row Culture

Submitted by: Asheberom G/slassie _____ _____

Student Signature Date

Approved by Board of Examiners

Chairman, Department of Graduate Committee Signature Date

_____ _____ _____

Dr. Dereje Shiferaw Signature Date

Advisor _____ _____

_____ Signature Date

Internal Examiner _____ _____

_____ Signature Date

External Examiner _____ _____

DECLARATION

I, the undersigned declare that this thesis is my original work, and has not been presented for a degree in this or other universities, and all sources of materials used in this thesis have been fully acknowledged.

Name: Asheberom Gebreslassie

Signature: _____

Place: Addis Ababa Institute of Technology, Addis Ababa University, Addis Ababa Ethiopia

This thesis work has been submitted for examination with my approval as a university advisor

Dr. Dereje Shiferaw	_____	_____
Advisor	Signature	Date

ACKNOWLEDGMENT

First of all, I would like to thank the Almighty God for his support to overcome trials and temptation to finish the whole work. May he continue to give me strength and vision that I may follow his path for saving of my soul from sin and its consequences.

Secondly, I want to express my greatest gratitude to my advisor, **Dr. Dereje Shiferaw** for giving me the topic and the opportunity to research under his guidance and supervision. I received motivation, comments, encouragement and continuous guidance from him during my graduate studies.

Last but not least, I would like to thank my family for their great encouragement from day to day during my study.

ABSTRACT

Agricultural robotic vehicles have the capacity to play a key role in the future of agriculture. For this to have controller designs that are cost effective and easy to use is very important.

Agricultural robots which operate in row culture agricultural need to strictly follow the wheel tracks. The robot kind selected in this thesis is a differential drive wheel agricultural robot with one rear mounted caster and two front wheels. Navigation errors where the robot sways of its path with one or more wheels may damage the crops.

Since model of the plant is paramount in designing of the MPC, mathematical model of the robot involves two identical series DC motors dynamics, the robot chassis dynamics and the robot kinematics. The model of the robot is simulated in MATLAB. The effect of change in mass of the robot is considered as disturbance for the better position and orientation straight crop row tracking. This paper focuses on the designing of MPC (Model Predictive Control) for agricultural robot operation in row cultures and then improves the performance of the robot to track its position and orientation from the desired crop line position. The two DC series motors found in each front wheels and the controller are simulated by MATLAB. The input component (pinhole camera) is specified.

The robot position (x, y), orientation (α) and control signals (u_L, u_R) are constrained to restrict the forward speed and maximum error of the angle or heading. The performance of the proposed controllers for position and orientation set point tracking is evaluated through simulation studies. The simulation results show that the cost of tracking the desired position (x_r, y_r) and orientation (α_r) of the crop in the row is $j = 1.0653 * e^{-10}$. The lowest cost function means the lowest error between the desired and actual position, orientation. The MPC approach is very advantageous and display better performance when facing the path constraints of operating in agricultural which follow row culture.

Key words: Model Predictive Control, Mathematical model, Differential drive wheel agricultural robot, Disturbance, Crop row tracking.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

MPC	Model Predictive Controllers
PID	Proportional-Integral-Derivative control
DDWAR	Differential drive wheel Agricultural Robot
DC	Direct current
LWPR	Locally Weighted Projection Regression
SHT	Standard Hough Transform
EMF	Electromotive force
MV	Manipulated variable
QP	Quadratic programming
NMPC	Nonlinear Model Predictive Controllers
MATLAB	Matrix laboratory
KVL	Kirchhoff's voltage law
WMR	Wheeld mobile robot
F	Actual force acting to a mass point
i_a	Armature current
v_f	Field voltage commands
i_f	Field current
v_b	The back EMF voltage
v_a	Voltage supplied by amplifier to move the motor
τ_m	Motor torque produced by the motor shaft

J_m , k_r	Moment of inertia and motor friction coefficient.
R_a	Resistance
L_a	Inductance
M	Load torque
V_a	Terminal voltage
k_b	Electro motoric constant
u_R , u_L	Control voltages of the right and left motors
i_L ,i_R	Left and right current flowing through winding
U_L, U_R	Left and right power supply voltage
ω_L	Left motor angular velocity
ω_R	Right motor angular velocity
M_L , M_R	Load moments on left and right wheels respectively
T	General center of gravity
K_v	*R*esistance coefficient against linear motion.
MG_L	Moment of the left drive
MG_R	Moment of the right drive
v_B	Linear motion speed
r	Semi diameter of the wheels
L_R	*D*istance of the right wheel from point B
L_L	Distance of the left wheel from point B
L_T	Distance of the center of gravity from point B
K_ω	Resistance coefficient against rotary motion
ω_B	Angular speed in point B
J_B	Moment of inertia with respect to rotation axis in Point B
F_L, F_R	Forces causing linear motion by drives and inertial force

F_S	Resistance force F_O proportional to speed v_B
v_L, v_R	Peripheral speeds
ωG_L, ωG_R	Angle speeds of gearbox output
P_G	The gear box transmission ratio
L, R	Driving wheels
x_B, y_B, α_B	Global coordinates of the robot and the orientation of the robot
v_B, ω_B	*L*inear and angular velocities
m_h	Mass of the herbicide in the tank
m	Total mass of the robot in operation
m_b	Mass of the robot body
ρ	Density of the herbicide
v	Velocity of the herbicide flowing
V	Volume of the herbicide flowing
A	Pipe cross sectional area
R, Q	Output weight, input weight.
W	Column vector of N future set points
Y	The outputs of the plant
J_m	Moment of inertia
k_r	Coefficient of rotation resistance and M_x is load moment
M_L , M_R	Load moments on left and right wheels respectively

CHAPTER ONE

1. INTRODUCTION

1.1. Background of the Study

The development of automated agriculture with respect to social desires needs proposing new method for food production especially in the fields of agriculture. Progresses achieved in robotics allowed to consider mobile robots as a promising solution to actually apply new methodologies. Nevertheless, in order to be fully autonomous, such device needs to be accurately and safely controlled [1].

As land for farms grow in size together with the size of the equipment used on them, there is a need for ways to automate processes previously performed by the farmer, such as controlling the fields for pests. The necessity of robots is a further new development as most of the existing solutions for automatic supervision is designed for standard farm equipment, such as tractors, combines and pesticide sprayers [2].

Applying herbicide to the crops is in most cases is important to remove unwanted plants and increases crop yields. The unwanted plants or weeds can be harmful to the crops. Usually though weeds only cover a very small fraction of the field. Adigo AS (Adigo Asterix) is currently developing an autonomous agricultural robot for detection and precision spraying of individual weed leafs, named Asterix. The robot follows the tractor wheel tracks in the field and treat the weeds in the crop rows with herbicide while navigating autonomously [3]

Model predictive controllers are used in industrial plants and in process control to optimize the operation values of the process. It is easier to use MPC in these environments due to longer time constant. If the time constants of the system are smaller as in mobile robot trajectory control, the controller must run with a higher control cycle. This needs high computing capacity for real time control. Research deals with implementation of the model predictive control for the path tracking purposes is scattered and those are mostly deal with the computational requirements. Model predictive control (MPC) is stronger than Proportional-Integral-Derivative (PID) control. Predictive control explains an approach to control design. Predictive controllers does not have a specific algorithm [4].

In this thesis model predictive control is used to track the trajectory of differential drive wheel agricultural robot in the crop row. Camera is used to identify rows position and orientation.

First, the dynamics of the differential drive wheel agricultural robot is modeled by considering motor dynamics and chassis dynamics. The nonlinear kinematic equations are linearized into error based model by forward difference method, where state variables are deviations from reference variables. Reference variables are variables of the ideal agricultural robot which follows reference trajectory or the desired robot path. Both the dynamic and kinematic models are augmented into a discrete time varying state space mathematical model. Control inputs are motor control variables and outputs are positions in x, y direction and orientation α_r .

1.1.1. Model predictive control

Model predictive control (MPC) is the merely control technology that can work with constraints on manipulated variables and output states. Operation near constraints can make the operation profitable and efficient [4]. The concepts behind predictive control are:

- Easy to understand
- Extensible to multivariable systems
- More powerful than Proportional-Integral-Derivative (PID) control

All formulations of the predictive controller have the following common components:

- An explicit internal model
- Idea of a receding horizon
- Computation of control signal by optimizing predicted plant conduct.

These ideas basically reflect human behavior

- Predictive control describes not a specific algorithm. It is an approach to control design and there are many types of predictive control laws.

Main parts of model predictive control are shared the following concepts.

- Actions depend on predictions
- Predictions are based on a model
- Achieving a best output is base for current input
- Limited time window or receding horizon
- Precise control requires an accurate model
- Handles constraints in a systematic way

1.1.2 Prediction of State and Output Variables

As the mathematical model is formulated the next step in the design of a predictive control system is to calculate the predicted process output with the future control signal or manipulated variables as the adjustable variables. This section examines in detail the optimization done within this window. It assumes that ki, Np are the current time and the length of the optimization window respectively. The derivation of the MPC algorism starts from the case of single-input and single-output systems and then the results are expanding to multi-input and multi-output system [4].

The predictive control solution extension is quite straightforward. Let us give attention to the dimensions of the state, control and output vectors in a mult-input multi-output environment. The vectors $Y\ and\ \Delta U$ are defined as

$$\Delta U = [\,\Delta u(ki)^T \Delta u(ki+1)^T \ldots \Delta u(ki+Nc-1)^T]^T \quad (1.1)$$

$$Y = [\,y(ki+1\mid ki)^T\, y(ki+2\mid ki)^T\, y(ki+3\mid ki)^T \ldots y(ki+Np\mid ki)^T]^T \quad (1.2)$$

The state-space model is basic to calculate the future state variables sequentially using the set of future control parameters.

$$x(ki+1\mid ki) = Ax(ki) + B\Delta u(ki) + Bde(ki)$$

$$x(ki+2\mid ki) = Ax(ki+1\mid ki) + B\Delta u(ki+1) + Bde(ki+1\mid ki)$$

$$= A^2x(ki) + AB\Delta u(ki) + B\Delta u(ki+1) + ABe(ki) + Bde(ki+1\mid ki)$$

...

$$x(ki+Np\mid ki) = A^{Np}x(ki) + A^{Np-1}B\Delta u(ki) + A^{Np-2}B\Delta u(ki+1) + A^{Np-Nc}B\Delta u(ki+Nc-1) + A^{Np-1}Bd\,e\,(ki) + A^{Np-2}Bde(ki+1\mid ki) + \ldots + Bde(ki+Np-1\mid ki) \quad (1.3)$$

Where, $e(k)$ is a zero-mean white noise sequence. The predicted value of$(ki+i\mid ki)$ at future sample i is assumed to be zero. The prediction of state variable and output variable is computed as the expected values of the respective variables. Hence, the noise impact to the predicted values is zero. For notational simplicity, the expectation operator is neglected without confusion.

Effectively, we have

$$Y = Fx(ki) + \Phi\Delta U \quad (1.4)$$

Where,

$$F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ . \\ . \\ . \\ CA^{Np} \end{bmatrix}, \quad \Phi = \begin{bmatrix} CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ CA^2\mathrm{B} & \mathrm{CAB} & \mathrm{CB} & \dots & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ CA^{Np-1}B & CA^{Np-2}B & CA^{Np-3}B & \dots & CA^{Np-Nc}B \end{bmatrix} \tag{1.5}$$

The incremental optimal control within one optimization window is obtained by

$$\Delta U = (\Phi^T\Phi + R)^{-1}(\Phi^T R_s\, r(ki) - \Phi^T F x(ki)) \tag{1.6}$$

Where matrix $\Phi^T\Phi$ has dimension $mNc \times mNc$ and $\Phi^T F$ has dimension $mNc \times n$ and $\Phi^T R_s$ equals the last q columns of $\Phi^T F$. The weight matrix R is a block matrix with m blocks and has its dimension equal to the dimension of Φ^T.The set-point signal is expressed as:

$$r(ki) = [r1(ki)\ r2(ki) \dots rq(ki)]^T \tag{1.7}$$

Applying the receding horizon control principle, the first m elements in ΔU are taken to form the incremental optimal control as follow:

$$\Delta u(ki) = [Im\ om \dots om](\Phi^T\Phi + R)^{-1}(\Phi^T R_s\, r(ki) - \Phi^T F x(ki)) \tag{1.8}$$
$$= Kyr(ki) - Kmpcx(ki)$$

Where, $[Im\ om \dots om] = Nc$,Im and om are respectively the identity and zero matrix with dimension $m \times m$.

1.1.3 Optimization

Model predictive control (MPC) is one of the optimal controllers. It is a balance between reactive controllers and infinite horizon optimal control methods. Model predictive control formulates the control problem as a finite horizon optimization problem with constraints [6].

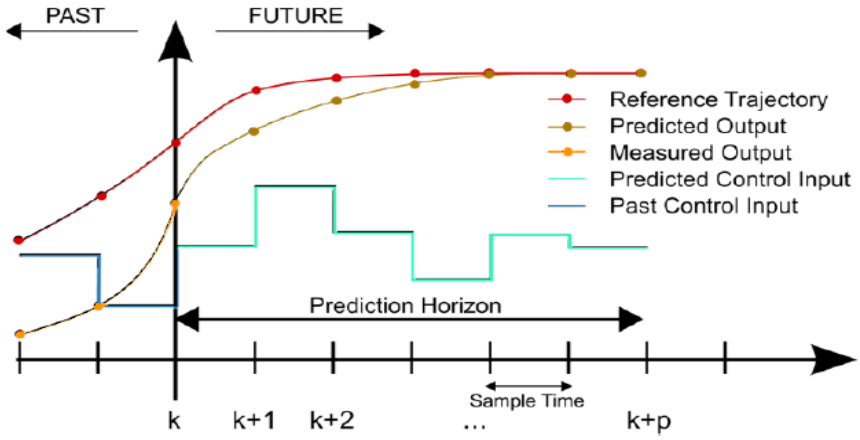


Figure 1.1: Model predictive control looks ahead over a finite horizon [6]

For a given set-point signal $r(ki)$ at sample time ki within a prediction horizon the objective of the predictive control system is to bring the predicted output as close as possible to the set-point signal. The reference signal kept constant in the optimization window. This idea is then transferred into a design to find the best control parameter vector ΔU in which an error function between the set point and the predicted output is minimized. The data vector that involves the set point information is formulated as

$$Rs^T = [1\ 1 \dots\ 1]\, r(ki) \tag{1.9}$$

Where, $[1\ 1 \dots 1] = Np$

The cost of the optimization J *that* reflects the control objective is explained as

$$J = (R_s - Y)^T (R_s - Y) + \Delta U^T R \Delta U \tag{1.10}$$

Where, the first part is connected to the objective of minimizing the deviations between the predicted output and the reference signal while the second part reflects the consideration given to the size of ΔU when the cost function *J* is considered to be as small as possible. *R* is a diagonal matrix in the form that $R = rwINc \times Nc\,, (rw \geq 0)$, where rw is used as a tuning parameter for the desired closed-loop performance. For the case that $rw = 0$, the cost function (1.10) is interpreted as the situation where we would not want to give any attention to how big the ΔU might be and our goal would be only to make the error $(R_s - Y)^T (R_s - Y)$ as small as possible. For the case of large rw, the cost function (1.10) is interpreted as the situation where we would carefully consider how large the ΔU might be and cautiously reduce the error $(R_s - Y)^T (R_s - Y)$.To find the optimal ΔU that will minimize *J*, by using (1.10), *J* is expressed

As,

$$J = (R_s - Fx(ki))^T (R_s - Fx(ki))^{-2}\, \Delta U^T \Phi^T (R_s - Fx(ki)) + \Delta U^T \Phi^T \Phi + R) \Delta U \tag{1.11}$$

Taking the first derivative of the cost function J, it is formulated as

$$\frac{\partial J}{\partial \Delta U} = -2\Phi^T \left(R_s - Fx(ki)\right) + 2(\Phi^T \Phi + R)\Delta U \tag{1.12}$$

The minimum cost function *J* is calculated as $\frac{\partial J}{\partial \Delta U} = 0$, the optimal value for the control signal is calculated as

$$\Delta U = (\Phi^T \Phi + R)^{-1} \Phi^T \left(R_s - Fx(ki)\right) \tag{1.13}$$

With the assumption that $(\Phi^T \Phi + R)^{-1}$ exists. The matrix $(\Phi^T \Phi + R)^{-1}$ is called the Hessian matrix in the optimization literature. Note that R_s is a data vector that contains the set-point information expressed as

$$R = [1\ 1\ 1 \ldots 1]^T\, r(ki) = R_s\, r(ki)$$

Where

$$R_s = [1\ 1\ 1 \ldots 1]^T$$

The optimal solution of the control signal is connected to the reference signal $r(ki)$ and the state variable $x(ki)$ via the following equation.

$$\Delta U = (\Phi^T \Phi + R)^{-1} \Phi^T \left(R_s r(ki) - Fx(ki)\right) \quad (1.14)$$

1.1.4. Multi-input and Multi-Output System

Because of the state-space formulation, the above SISO output system model design methodology can be extended to multi-input and multi-output systems without much further effort.

Consider that the process has *m* inputs, *q* outputs and *n*1 states. The number of outputs is considered to be less than or equal to the number of inputs (*i.e.*, $q \leq m$). If the number of outputs is greater than the number of inputs, it is not possible to hope to control each of the measured outputs independently with zero steady-state errors. In the general formulation of the predictive control problem, taking the plant noise and disturbance into consideration the state space looks as fellow.

$$xm(k+1) = Amxm(k) + Bmu(k) + Bd\omega(k) \quad (1.15)$$

$$y(k) = Cmxm(k) \quad (1.16)$$

where, ω (*k*) is refer to the input disturbance that considered to be a sequence of integrated white noise which means that the input disturbance $\omega(k)$ is related to a zero mean, white noise sequence e(*k*) by the difference equation written as fellow,

$$\omega(k) - \omega(k-1) = e(k) \quad (1.17)$$

Note that from (1.15), the following difference equation is also true:

$$xm(k) = Amxm(k-1) + Bmu(k-1) + Bd\omega(k-1) \quad (1.18)$$

By defining$\Delta xm(k) = xm(k) - xm(k-1)$ and $\Delta u(k) = u(k) - u(k-1)$, then subtracting (1.18) from (1.15) leads to $\Delta xm(k+1) = Am\Delta xm(k) + Bm\Delta u(k) + Bde(k)$ (1.19)

In order to relate the output *y*(*k*) to the state variable $\Delta xm(k)$, It is deduced to

$$\Delta y(k+1) = Cm\Delta xm(k+1) = CmAm\Delta xm(k) + CmBm\Delta u(k) + CmBde(k) \quad (1.20)$$

Where, $\Delta y(k+1) = y(k+1) - y(k)$.

Choosing a new state variable vector$x(k) = [\Delta xm(k)^T\ y(k)^T\ ]^T$, we have:

$$\begin{vmatrix} \Delta xm(k+1) \\ y(K+1) \end{vmatrix} = \begin{bmatrix} Am & 0m^T \\ CmAm & I_{q\times q} \end{bmatrix} \begin{bmatrix} \Delta xm(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} Bm \\ CmBm \end{bmatrix} \Delta u(k) + \begin{bmatrix} Bd \\ CmBd \end{bmatrix} e(k) \quad (1.21)$$

$$y(k) = [0m \quad I_{q\times q}]\begin{bmatrix}\Delta xm(k)\\ y(k)\end{bmatrix} \tag{1.22}$$

Where $I_{q\times q}$ is the identity matrix with dimensions' $q \times q$, which is the number of outputs; and *om* is a $q \times n1$ zero matrix. In (1.16, 1.15), *Am*, *Bm* and *Cm* have dimension $n1 \times n1, n1 \times m$ and $q \times n1$, respectively.

1.2. Problem statement

Most processes like agricultural robots are difficult to control with standard conventional controllers. There are useful process interrelations between the input to the plant and the output. More than one manipulated variable has a visible effect on a useful process variable.

Constraints (limits) on process variables and manipulated variables are necessary for normal control. Robot navigation in agricultural row crops must strictly follow the established wheel tracks. Deviations in stirring where the robot moves out of its path with one or more wheels may crashed the crop plants.

Model Predictive Control (MPC) is the unique advanced control technique that consist many influence on industrial process control. It is the only control technology that can deal with constraints on states, output and input variables of a process. Control system which does not involve the constraints cannot perform the goal of the controlling with outstanding performance like MPC. It is possible to get better capacity by involving the constraints directly into the model predictive control. Since model of the plant is important in designing a model predictive control, low plant models leads to incorrect control algorism. Hence the future manipulated variable and process outputs are predicted wrongly.

This research proposed a design of MPC for robot which sprays herbicide to the crops, the mass of the robot changes from time to time and has an effect on the control of the position(x, y) and orientation (α) of the differential drive wheel agricultural robot. The literatures covered on this paper did not deal with identifying the disturbance and design controller to minimize its effect. As model of the plant is paramount in designing of MPC, involving every component of the robot in the mathematical model has significant effect on tracking the output. Most researches consists the kinematic model alone for the design of MPC, hence as expect the robot faced starting problem.

In this research the MPC is designed for better performance with better plant model.

1.3. Objective

The proposed work has specific and general objects as shown below.

1.3.1. General objective

The main objective of this thesis is to design a model predictive control for differential drive wheel agricultural robot.

1.3.2. Specific objective

To achieve the general objectives of the thesis the following specific objectives is done:

- To model and implement the agricultural robot in MATLAB
- To design MPC for line following agricultural robot
- To restrict maximum angle difference to the Line/row and to follow a line with constant forward speed.
- To investigate better performance of MPC for the crop line following agricultural robot

1.3.3. Benefits

- Straight forward formulation based on well understood concepts
- Explicitly handles constraints
- Explicit use of a model
- Well understood tuning parameters. Best output achieved by tuning the tuning parameters.
- Optimization problem setup .The controller performance is determined by cost function.
- Easier to maintain and changing model
- Optimize the control signal
- Noisy state measurements estimation
- Reduce training
- Reduce cost

1.4. Scope and significance of the thesis

This thesis provides theoretical studies and simulation of model predictive control for agricultural robot operation in row culture. The scope of this thesis is to investigate how to improve the capability of tracking the desired position (x, y) and orientation (α) of the DDWAR (differential drive wheel agricultural robot) in order to protect the crop from damage, where the robot is controlled

by MPC then implement the system using the available mathematical computation tools of MATLAB /SIMULINK. In this thesis the nonlinear kinematic mathematical model of the DDWAR is linearized through Taylor series forward difference method and augmented with the defined robot dynamic model. This thesis up grades the detail knowledge of characteristics of the controller during navigation in straight line. This study also increases the knowledge that the MPC is best controller for the constrained MIMO system.

1.5. Methodology

The methodologies that followed in this thesis are listed by the following chart.

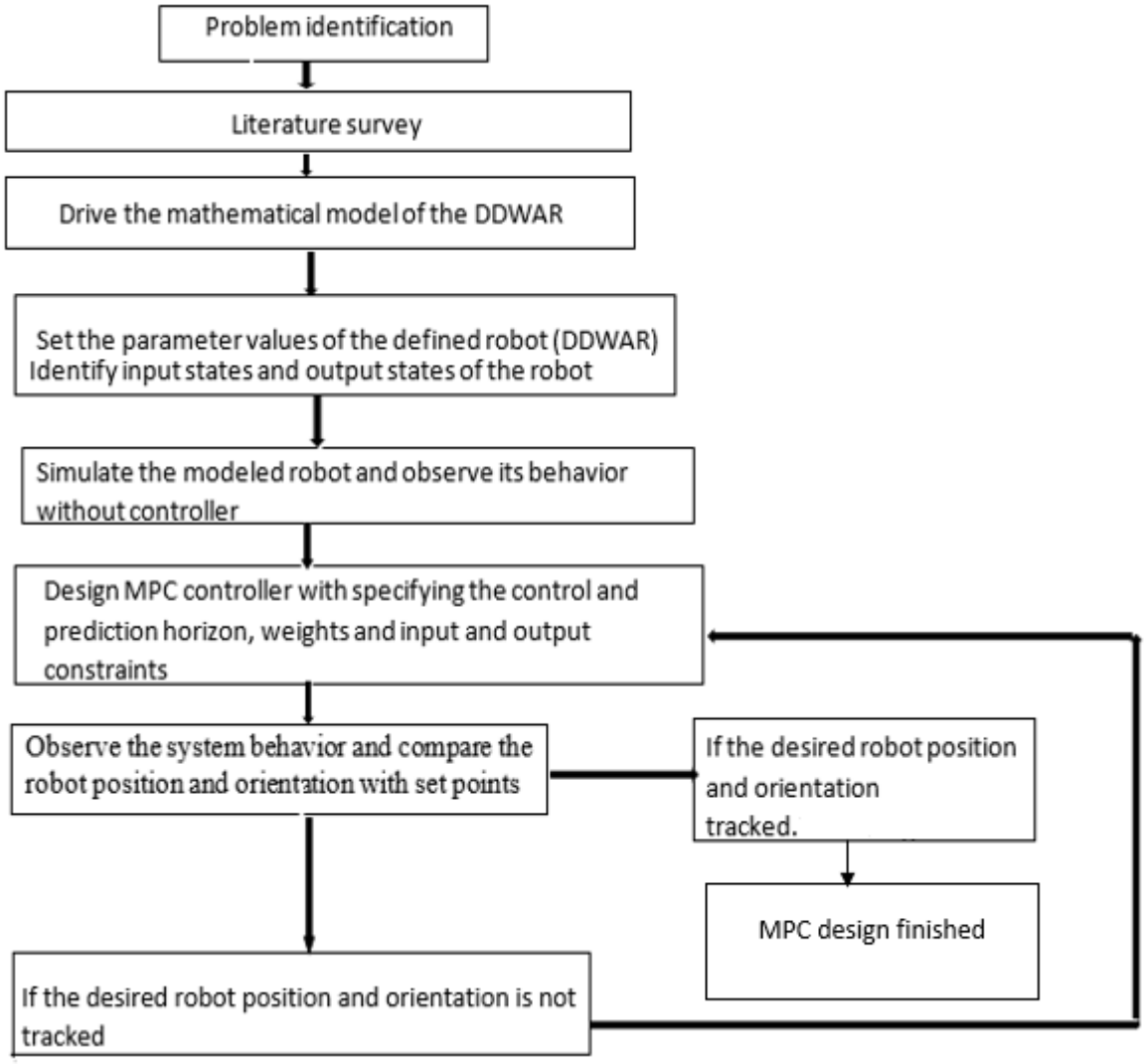


Figure 1.2: General methodology flow chart

1.6. Organization of this Thesis

This research contains totally five chapters and the content of each chapters are disused as fellow.

Chapter one is an introductory chapter and presents an overview of the thesis background, objectives, scope and significance and a brief summary of the thesis. It also presents a brief introduction to model predictive control.

Chapter two provides a review of the literature on some components of differential drive wheeled agricultural robot control and related works on MPC design for agricultural robots.

Chapter three contains the modeling of the differential drive wheeled agricultural robot and controller design. In this chapter the kinematic nonlinear model of differential drive wheeled agricultural robot is linearized and augmented with dynamic model of the robot. The camera used in the operation is explained. The DC series motor parameters, robot chassis parameters and MPC design parameters are specified. Disturbance that affects the control is modeled.

Chapter Four presents implementation results of the system using MATLAB. It describes each of the subsystems in detail. System response with disturbance and without disturbance is simulated and analyzed. This chapter also discusses the open loop control behavior of the differential drive wheel agricultural robot.

Chapter five involves conclusion, recommendation and future work potential research ideas of the thesis.

CHAPTER TWO

2. THEORETICAL BACKGROUND AND LITERATURE RE-VIEW

2.1. Agricultural Robot

An agricultural robot is a vehicle set for the purpose of automating agriculture. It is drawing new professionals, new companies and new investors. The agricultural robot is developing soon not only increasing the production capabilities of farmers but also improving robotics and automation technology. Agricultural robots are amending production yields for farmers in many ways. The technology is being implemented in creative and innovative application areas like drones, autonomous tractors and robotic arms. Agricultural robots automate slow, repetitive or tedious and dull tasks for farmers permitting them to focus more on developing overall production yields [5]. Some of the use of most common robots in agriculture are listed as fellow:

- Harvesting and picking
- Weed control
- Autonomous mowing, pruning, seeding, spraying and thinning
- Phenotyping
- Sorting and packing
- Utility platforms

This thesis deal with the development of an automatic row crop follower system for a differentially drive wheeled agricultural robot. The robot is named as Adigo or Asterix. It is developing an autonomous robot for weed control in row crops.

Asteri*x* is a differentially steered mobile platform with two actuated front wheels and a passive off-center rear caster wheel. The robot will drive autonomously in the tractor wheel tracks and treat the row crop in between the wheels. It *is* a differentially steered robot with an off-center rear caster wheel [3].

The overall task of this thesis is to develop a system that is able to follow the rows. In this thesis agriculture robot mathematical model is designed that it is assumed to perform spraying (water, liquid fertilizer or pesticide) in a crop row.

2.1.1. Three-wheel agricultural robot design

The design of differential drive wheel agricultural robot satisfies the following requirements. These are being able to move along a guidance path autonomously with steering control on a flat field, having open control system and platform for easy extension and reconstruction, which means that several farming operations such as spraying, weeding, cutting and so on could be performed on the same platform by adding or removing sensor(s), replacing actuator(s) and switching control software. Therefore, the platform can be designed based on an accepted concept of modular design [7].

The off center three-wheel configuration of the robot as shown in the figure *2.1*, allows for a lighter design with fixed wheel suspension, and only two motorized axes. A cost-effective robot design is using differential drive front wheels and rear castor wheels. Equipping to center the rear wheel is not a good design for row crops, as the wheel can damage the crop. Therefore, using one off-center rear castor wheel is better than centered rear wheel agricultural robot. This necessitates a design with special care to weight distribution and stability of the robot in the line or path. The operational speed of the vehicle is sufficiently low such that a wheel suspension system from a vibration perspective is not required.

Three-wheel design are not common and they have been disregarded in design of agricultural robot, as a symmetric design would not be suitable for operations in row crops. By designing the system with asymmetrical three-wheel configuration, a minimal wheel configuration while is maintaining the systems suitability for operations in row crops. By designing the robot ground up, a highly cost-effective robot with a minimum of movable parts and good handling capabilities and stability is obtained [7].

The three wheeled design is uncommon and the design maintains maneuverability and stability with the importance of decreasing weight, complexity and cost.

2.1.2. Differential drive robot wheel scheme

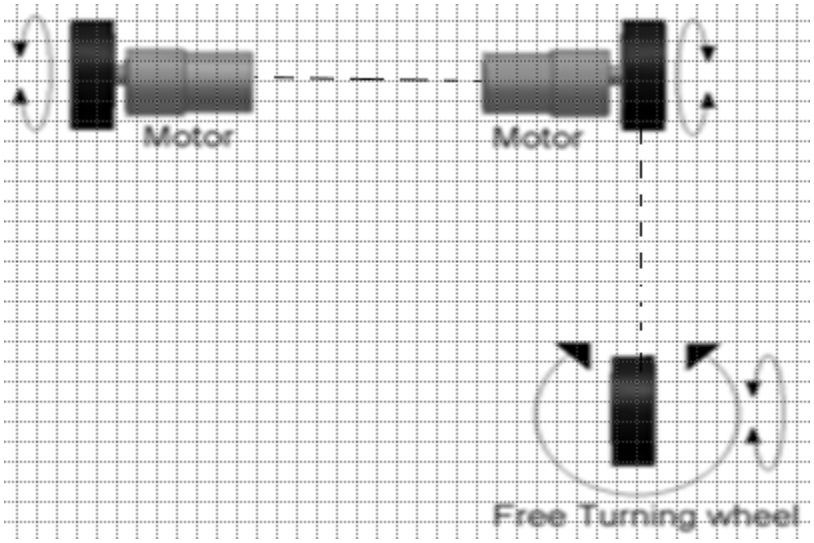


Figure 2.1: Differential drive wheel scheme

A differential drive wheel agricultural robot is a movable robot whose movement is deepened on two separately driven wheels connected on either side of its body. The direction of motion of the robot is changed by varying the relative rate of rotation of its wheels and hence does not need an extra steering motion. To make balance the robot, additional wheels or casters may be connected. The robot will go in a straight line if both the wheels are driven in the same direction and speed. As it is clear from the diagram shown, the robot will rotate about the central point of the axis if both wheels are turned with equal speed in opposite directions otherwise, depending on the speed of rotation and its direction, the center of rotation may fall anywhere on the line defined by the two contact points of the tires. When the robot is moving in a straight line, the center of rotation is infinite distance from the robot. Because of the direction of the robot is dependent on the rate and direction of rotation of the two driven wheels, these quantities should be sensed and controlled precisely.

2.1.3. Components of differential drive wheel agricultural robot

Components of differential drive wheel agricultural robot are divided into four parts. Those are sensing unit, a control unit, an actuating unit and a mobile platform. The sensing unit involves some sensors for target identification of navigation and operations. Sensors such as vision, en-

coder, gyro, laser radar, camera or their combination, etc. can be used to acquire necessary information for navigation and operation. The control unit processed the sensing information and output commands to the mobile platform for robot motion and the actuating unit for field operations [7].

2.2. Basic Control Structure

The control system of the robot consists all sensors, controllers with a host computer and a slave computer, all drivers for DC motors, stepping motors and actuating tools. A hierarchical structure is applied to the control system where the host computer usually a laptop computer with high speed and large memory capacity acted as a decision making unit and control core of the robot performing acquisition and processing from sensor(s) and path planning for navigation and operations according to sensing information. Also, the host computer coordinated commands and output them to the slave computer. The slave computer is usually a microcomputer only drove DC motors actuating tool by the important drivers to finish the assigned tasks from the host computer. An RS232 is useful for communication between the host computer and the slave computer [7].

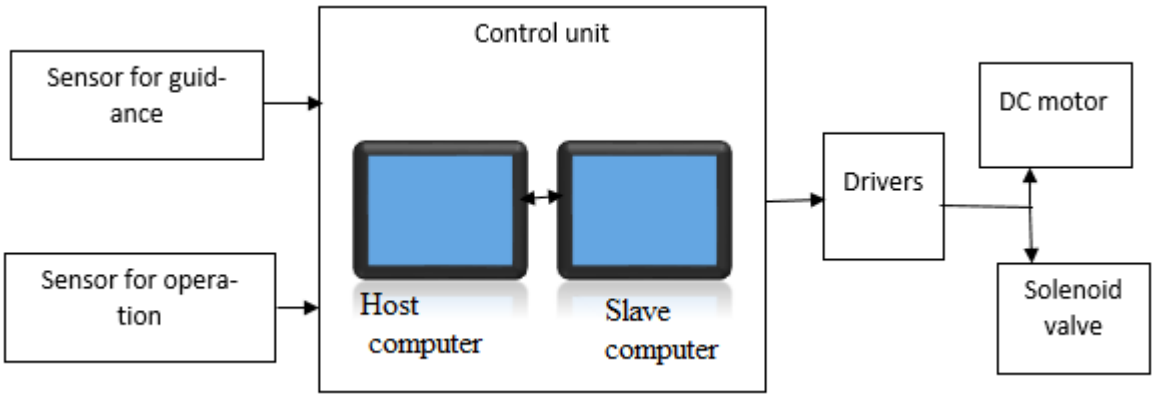


Figure 2.2: Control system of the agricultural robot [7]

2.3. Feedback Control Strategies

Feedback control is a closed loop control that determines the control input by using the output from the plant to estimate the system current state and measure deviation from the desired state. Based on the calculated error, the controller generates an input signal to reduce the error and improve tracking of the reference as shown in Fig. 2.3. Often in real world applications, there are unpredictable factors that affect the plant as well, such as uncertainty in the state estimate or external disturbances that are not part of the process model. Figure 2.4 categorizes different kinds of

feedback control based on their capability to handle disturbances, account for uncertainty and involve state or input constraints.

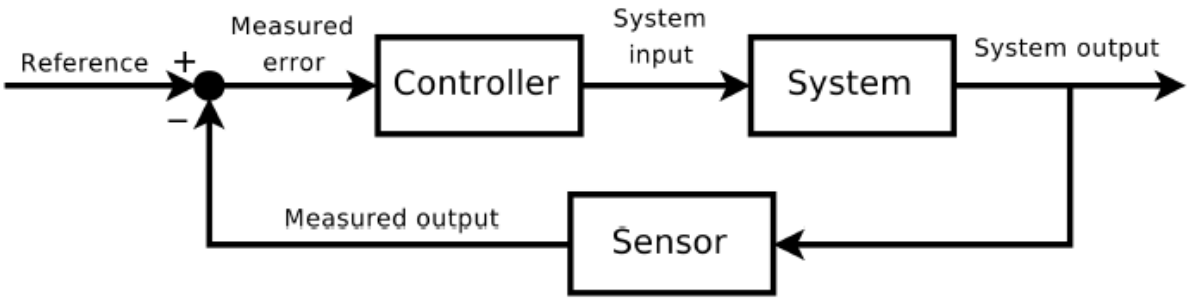


Figure 2.3: A block diagram of a closed loop control system using a feedback loop [8]

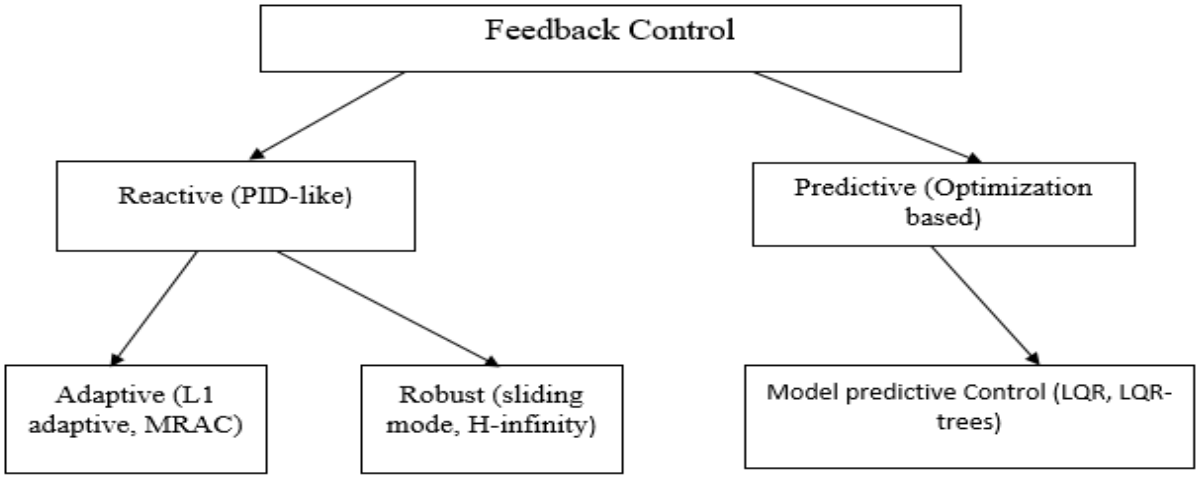


Figure 2.4: A taxonomy of feedback control [8].

Reactive control strategies aim to generate a control input based on past and current measurements of the system's state and error relative to the reference. The ubiquitous example of reactive control is proportional-integral-derivative (PID) control which can run at high rates due to its simplicity [9]. To account for uncertainty, this method can be augmented by adaptive control strategies such as L1 adaptive control or model reference adaptive control [10, 11]. Adaptive techniques are flexible by estimating and compensating for perturbations to the system online. However, guarantees of constraint satisfaction do not provide in such methods.

Alternatively, robust control strategies such as sliding mode or H-infinity control bound the effects of uncertainty to preserve stability and tracking performance. These techniques use a static control

policy and assume specific variables are known or bounded to perform disturbance rejection and provide guarantees. In this way, robust controllers are designed to handle a predefined worst case scenario. When these methods provide guarantees on constraint satisfaction and stability, they can be overly conservative and usually don't account for system limitations. In addition, both adaptive and robust control techniques seek to eliminate the effects of un-modeled dynamics, even if they could be beneficial [12, 13, 14].

In contrast to reactive control strategies, predictive methods look ahead to anticipate the evolution of a system's dynamics, estimate future states and select control actions to optimize the system's performance for current and future states [15]. This optimization problem seeks to minimize a given cost function that weighs the importance of multiple factors including tracking error, control effort and constraint satisfaction mathematically. All optimization based control strategies provide versatility, because they are shaped by a designated cost function and can be formulated to incorporate constraints. However, optimal control strategies are complex and computationally expensive to solve for nonlinear systems [16]. Hence, optimal control strategies are not easy to run in real time on computationally constrained platforms.

2.3.1. Model Predictive Control with Uncertainty

In real world environments, a fixed dynamics model may not accurately predict future states when there are modeling errors or unmolded, potentially time-varying disturbances affecting the process. This has led to MPC formulations that compensate for different sources of uncertainty by updating the dynamics model or tightening constraints to still provide constraint satisfaction [17]. There are two broad categories of methods called adaptive and robust formulations used to account for uncertainty.

Adaptive MPC approaches estimate uncertainty in the system's dynamics and update the predictive model accordingly. They often assume a structured system model with some uncertain parameters. Such methods often combine MPC with an online parameter estimator [18]. Rather than using parameter estimation to capture model uncertainty, learning-based function approximation techniques are semi-structured approaches that can be useful to describe more complex perturbations that occur. In this case, a structured system dynamics model is modified with a non-parametric, online learned component via a Gaussian process [19]. However, kernel-based approaches such as Locally Weighted Projection Regression (LWPR) [20] and Incremental Sparse Spectrum Gaussian Process Regression [21] scale better with training data, providing incremental updates

for fast model learning [22]. The disadvantage of adaptive techniques is the absence of constraint fulfilment. These strategies are sensitive to delay between the disturbance estimator and its incorporation into the system dynamics. In contrast, robust MPC involves constraints explicitly to include uncertainty, it can provide guarantees in the presence of bounded and unpredictable parameters [24, 23]. While robust methods tend to yield conservative controllers, these methods can handle high frequency noise in the state estimate and do not need a disturbance estimator to restrict growth in uncertainty. Strategies such as Tube MPC [25] can yield better performance by using local feedback control to limit the uncertainty as the system evolves. Another robust strategy modifies robustness bounds online based on the state estimate [26], thus improving performance by relying on a probabilistic representation and adjusting constraints accordingly.

2.4. Review of related Papers

This topic explains the detailed literature study on design a model predictive control for agricultural robot operation in row culture irrigation. The information was gathered from several sources such as books, journals and websites. Each reference used is explained in detail as follows.

Benet, R. Lenain [1] ,in this research, perception sensors embedded on a light mobile robot were used to identify natural objects in agricultural crops working in various fields with or without soil perturbation and with different speeds and several vegetation levels to achieve crop row tracking tasks from a desired lateral deviation between robot and crop line or travers ability operations which consisted to take a decision in vehicle navigation according to the size and nature of the detected objects in front of vehicle.

In Vougioukas [31], the steering angle and the speed of the vehicle were controlled by nonlinear MPC (NMPC). The criteria were the error between the desired robot path and the predicted path. Although experiments were done completely in a simulator, well results were achieved and the advantage of this approach was shown.

Klancˇar and Škrjanc [32] proposed a tracking-error model-based predictive control law for tackling the trajectory tracking problem for a nonholonomic wheeled mobile robot. The prerequisite was that the reference path should be a smooth twice the differentiable function of time. The control law was based on a linearized error dynamics model obtained around the reference path. The resulting control law was analytically derived and therefore it was computationally easy. The MPC was not guaranteed to be stable due to the finite prediction horizon.

Kuhne et al [30] presented nonlinear MPC and linear MPC methods used to solve path tracking problem on a nonholonomic wheeled mobile robot. In their research, the computational effort required to solve the optimization problems was studied and the performance of both controllers were compared. They found out that at the time of their research that the Nonlinear Model Predictive Controller was computationally too demanding to be solved in real-time with a prediction horizon larger than five and also that the linear MPC had a good performance with lower computational effort. But, they also noted that the linear model is only valid near the reference trajectory. Urdal, F., Utstumo, T., Ellingsen, S. A., Vatne, J. K. & Gravdahl, T. [29] focus on an agricultural robot operation in row cultures. A machine vision was used to detect weeds within the crop rows and treats them by high precision drop-on-demand application of herbicide.

It has been formulated as a nonlinear model predictive control (NMPC) problem with the objective of keeping the vision modules centered over the seed rows, and constraining the wheel motion to the defined wheel tracks. The robots and vehicles operating in the field are restricted to these wheel tracks to avoid damage of the crop. The disturbances are considered as zero for the simulations shown, while it provides a possible input for perturbing the system in simulation. The system has been simulated without disturbances from various perturbed initial conditions to investigate the system behavior. The target velocity has been set to u setpooint= 0.3m/s and the crop row is at the origin of the coordinate system, yrow = 0. The controller takes two iterations to converge to the optimal trajectory. In addition to the displayed figures, the NMPC controller has been initialized in several infeasible initial conditions. The trajectory then diverges from the desired trajectory for as long as the path constraint can be met. These scenarios break the assumption of small y deviations, which the symmetry assumption of the path constraint relies on. The PD controller is not able to converge as quickly without violating the constraints the computational performance of the algorithm has not been systematically evaluated, but the run-time is consistently below 50 ms per iteration.

Yulin Zhang, Daehie Hong, Jae H. Chung, and Steven A**.** Velinsky [27] the motion of such robots can be accurately described with a dynamic model that includes both the external forces and a complex tire model. Control design from the full dynamic model is not practical due to its complexity and non-analytic form. In this paper, they derive a simplified dynamic model which is adequate for control design and treat the remaining terms as model uncertainty. Then, the uncertainty is analyzed and a robust control algorithm is designed. Computer simulation was used to

prove performance and robustness. Full dynamic model as the basis of control design is not practical due to the complexity of modeling the tire-ground. In this way, the full dynamics of the applications, tracking ability is essential since most tasks system including the tire slip is effectively reflected on the involve tracking a predefined path and/or a detected path in a control design, while the developed control law is simple real-time manner; e.g., pavement crack detection by a laser enough to be practically implemented. Through computer range sensor and corresponding crack following. Simulations of the robust tracking control algorithm developed tracking control algorithms for WMRs have been in this paper shows high tracking performance and extensively studied in recent years.

In Jin-lin XUE, Bo-wen FAN, Xin-xin ZHANG and Yong FENG [7] a multipurpose robot was developed to do several tasks such as spraying and weeding in a greenhouse. Many tasks were conducted by adding or neglecting sensing component(s), replacing actuator(s) and switching control software, with little or no change of the platform. Sliding mode control was applied to control motion of the agricultural robot in light of its kinematic model. Vision based algorithms were developed for navigation of row planting crops, taking operations of spraying water and weeding by machine. The experiment were done to assess both guidance and operations in a greenhouse with green vegetable, respectively. The results showed that the robot moved with a maximum lateral error of 47 mm.

CHAPTER THREE

3. MATHEMATICAL MODEL OF DDWAR AND CONTROLLER DESIGN

3.1. General Block diagram of the system

The crop row is captured by forward facing camera. Once the robot has the desired position and orientation of the crop row it has to be navigated by motion controller.

To know the crop position, orientation and the line fits to them, detection and mapping of rows is an important part of autonomous navigation and has received a lot of research interest. But this research focuses on designing MPC for robot motion in the row. The detail detection and mapping of the crop row left to do in the future.

The aim is to create model that describes trajectory of a robot's in the crop row point. The trajectory of the agricultural robot depends on source voltage of both drive motors. The dynamic behavior of engines and chassis, form of coupling between engines and wheels and basic geometric dimensions are taken into account. The dynamic model and kinematic model combined together is used for design and showing of a robot's motion control in the crop row operation.

As shown in the diagram the desired position of the crop row is obtained from the camera and it is used for Precalculation of the reference control signals. The robot is controlled by model predictive control output control signal summed with reference control signal. This is happened because the plant model is error model and then the output control signals are error control signals, hence it is added with reference control signals. The reference generator is reference robot at steady state of its dynamic equation. The model of the plant is based on state space model design. The state variables based on the difference of the actual and the states of the reference robot.

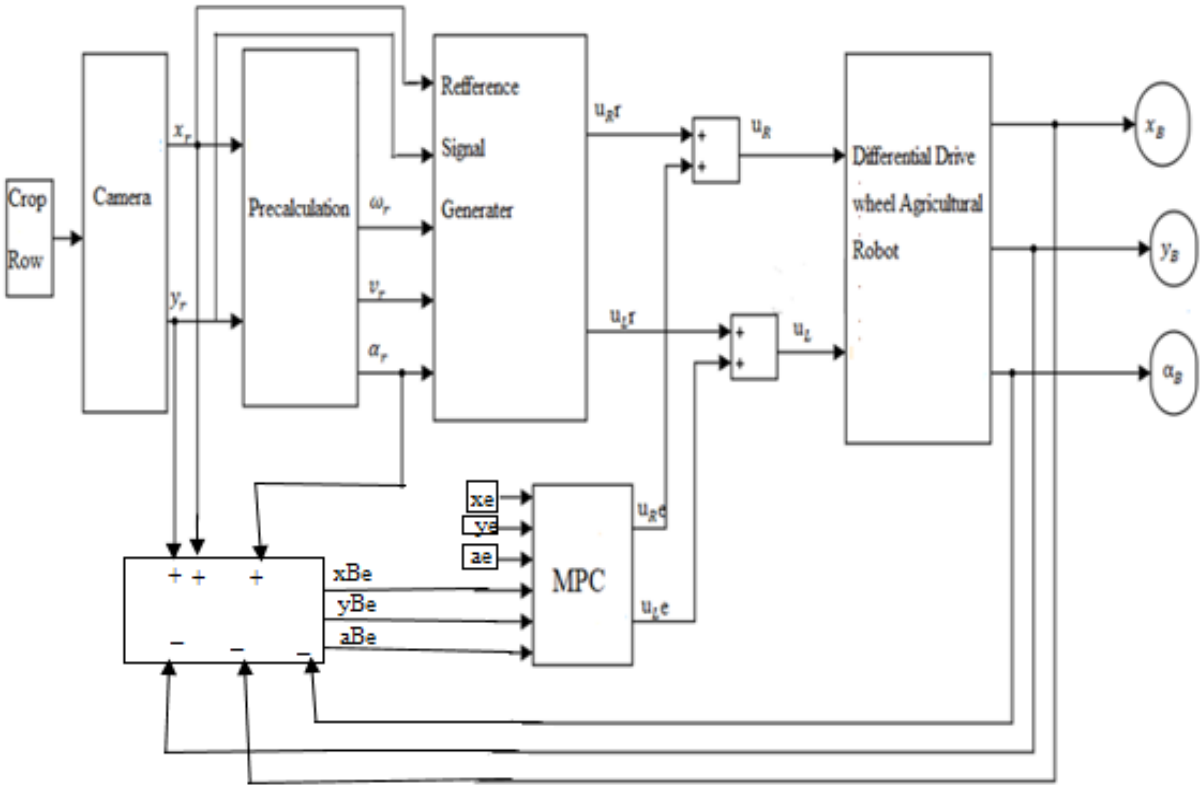


Figure 3.1: General Block diagram of the system

3.2. Straight line motion of the robot

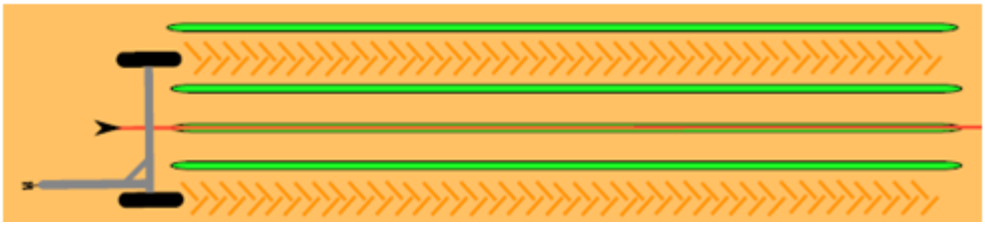


Figure 3.2: Illustration of the navigation problem

The robot starts at lower left position follows the middle row until the end is reached. The field has established tractor wheel tracks that the robot should follow and in between the wheel tracks there are three rows of crops as seen in Figure 3.2. The approach used here for navigation is to identify the middle row and attempt to follow it closely. The rear caster wheel is mounted off-center to ensure that no plants are damaged during operation. Its width is the same as a tractor to be able to follow the tractor wheel tracks in the field.

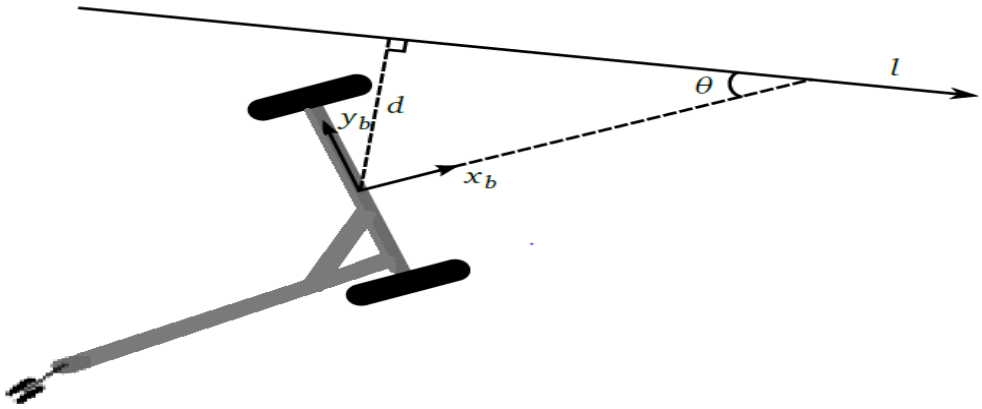


Figure 3.3: Illustration of a line l defined in global coordinates

That the robot is set to track an orthogonal projection from the point $(xb, yb) = (0, 0)$ on to l gives the distance d, while the angle Θ is the angle difference of the robot and the line. The controller attempts to satisfy$(d, \Theta) = (0, 0)$.

3.2.1. Pinhole camera model

A digital image is mostly expressed as a matrix. Each element of the matrix represents a pixel which is the smallest addressable element of an image. The position of a pixel can be expressed in a two dimensional coordinate system.$\mathbf{x} = [\mathrm{x}\ \ \mathrm{y}]^{\mathrm{T}} = (\mathrm{x}, \mathrm{y})$.

Agricultural robot has a camera equipped in the front angled about 45 degrees downwards. The purpose of this is to identify and estimate a crop row and its position and orientation relative to the robot. This means that once the row has been identified in an image, the coordinates of the row have to map into base link frame.

A pinhole camera is one of the simplest camera which consists of a closed chamber with a very small hole (pinhole) in the front with no lens [12]. The idea is that light rays from objects in the world pass through the pinhole and form an upside-down image in the back of the chamber. The distance from the pinhole to the image plane is known as the focal length. To describe this mathematically, it is easier to think of a virtual image at a distance equal to the focal length in front of the camera. This is illustrated in Figure 3.4.

The pinhole camera model is an ideal and simplified model of a camera, but in many cases it provides a good approximation. A mapping from world coordinates to image coordinates is shown below. The world coordinate system's origin is placed at the optical center (pinhole) and is represented as

$$\mathrm{W} = [\mathrm{u}\ \ \mathrm{v}\ \ \mathrm{w}]^{\mathrm{T}} \tag{3.1}$$

The image coordinate is represented as

$$X = [x \quad y]^T \tag{3.2}$$

The position and orientation of world and image coordinate systems are shown in Figure 3.3. In the next sections a model for perspective projection will be driven, i.e. the mapping of 3D points W to image points X.

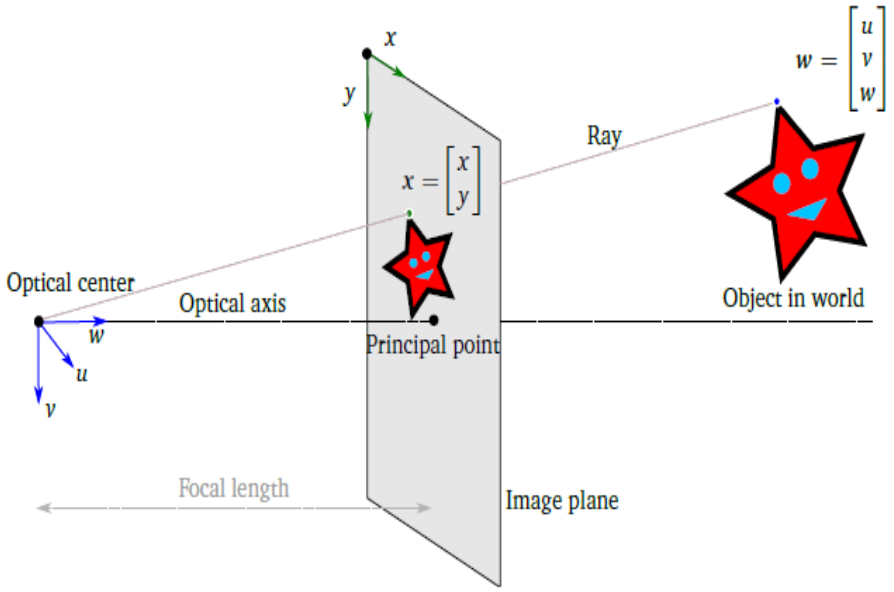


Figure 3.14: Illustration of coordinate systems for the pinhole camera model [5]

3.2.2. Intrinsic Parameters

Intrinsic parameters or camera parameters are parameters that describe the camera itself. These remain unchanged unless the camera configuration changes, e.g. by zooming or replacing the lens. First, let's consider the idealized normalized camera. The normalized camera is a pinhole camera with focal length equal to one and image coordinate system centered at the principal point [5]. This is illustrated in Figure 3.4. It can easily have been seen by use of similar triangles that the image coordinates can be calculated using

$$x = \frac{u}{w}, \quad y = \frac{v}{w} \tag{3.3}$$

Where, all lengths are measured in length unit. E.g. meter, and not in pixels. This model has some faults. First of all, image position is usually given in pixels instead of lengths, and there's no reason why focal length should be exactly one. This can be compensated by multiplying (3.3) with scaling factors ϕ_x and ϕ_y. Secondly, the image point $\boldsymbol{x} = [0 \quad 0]^T$ will rarely be at the principal point

compensate for this offsets δ_x and δ_y are added to (3.3). Finally, a skew term γ *is* added to the *x* position which is multiplied with the real world *v* value. This term does not

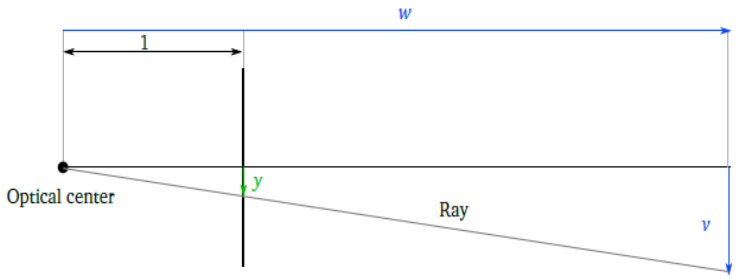


Figure 3.15: Normalized pinhole camera model [5]

have a simple physical explanation, but it is useful in practice. The resulting model with the added corrections for intrinsic parameters is [12] .

$$x = \frac{\phi_x u + \gamma v}{w} + \delta_x \quad , y = \frac{\phi_y v}{w} + \delta_y \tag{3.4}$$

3.2.3. Extrinsic Parameters

Extrinsic parameters describe the position and orientation of the camera. This is done by converting world coordinates W using a rotation matrix and a translational vector Ω the transformed point W' is expressed as [12]

$$W' = \begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \begin{bmatrix} \omega_{11} & \omega_{12} & \omega_{13} \\ \omega_{21} & \omega_{22} & \omega_{23} \\ \omega_{31} & \omega_{32} & \omega_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} + \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \Omega w + \tau \tag{3.5}$$

3.2.4. Full Pinhole Camera Model

Three-dimensional real world objects projection in to a two-dimensional image is an important basis for finding real world position and orientation of a row based on an image.

The full pinhole camera model can be described by combining (3.5) and (3.4). The mapping from a three-dimensional point in world coordinates $W = [u \;\; v \;\; w]^T$ to two-dimensional image point $\boldsymbol{x} = [x \;\; y]^T$

$$x = \frac{\phi_x \left(\omega_{11}\, u + \omega_{12}\, v + \omega_{13}\, w + \tau_x\right) + \gamma\left(\omega_{21}\, u + \omega_{22} v + \omega_{23} w + \tau_y\right)}{\omega_{31} u + \omega_{32}\, v + \omega_{33} w + \tau_z} + \delta_x \tag{3.6}$$

$$y = \frac{\phi_y \left(\omega_{21}\, u + \omega_{22}\, v + \omega_{23}\, w + \tau_y\right)}{\omega_{31} u + \omega_{32}\, v + \omega_{33} w + \tau_z} + \delta_y \tag{3.7}$$

3.3. Hough Transform

An important step of estimating the direction and position of row crops in an image is line detection and fitting. Hough transform techniques are presented in this section.

The Hough function implements the standard Hough transform (SHT) and it is designed to detect lines using the parametric representation of a line:

$$rho = x * cos(theta) + y * sin(theta) \quad (3.8)$$

The distance from the origin to the line along a vector perpendicular to the line refers to rho. The angle between the x-axis and this vector refers to theta. A parameter space matrix whose rows and columns correspond to these rho and theta values, respectively can be generated by Hough function. After calculating the Hough transform, it is possible to use the Hough peaks function to find peak values in the parameter space. These peaks refer to the potential lines in the input image. After identifying the peaks in the Hough transform, it is possible to use the Hough lines function to get the endpoints of the line segments corresponding to peaks in the Hough transform. This function automatically fills in small gaps in the line segments.

3.4. Mathematical model of differential drive wheel agricultural robot

Before designing a control system for a robot understanding its construction is very important. The more information is given to the controller, the better the performance is obtained.

In the differential drive wheel agricultural robot control system design, the main objective is to order the robot to follow the line on the crop. The control design principle is quite clear that is motor voltage controls the motor speed on each side between the two motors controls the position orientation, or direction, of the robot.

Mathematical model of the differential drive wheel agricultural robot involves three independent parts.

- Dynamics of the DC series motor
- Chassis dynamics
- Kinematics of the robot

This mathematical model expands basic model with dynamic part describing wheel speed relation with motor supply voltage by respecting dynamics, construction, geometry and other parameters of chassis and motors. Motor supply voltage actuating the wheel creates driving torque and then

wheel rotation. Inertial and resistance forces act opposite to driving torque and both driving torques influence each other because of these forces. Planar curvilinear motion of the agricultural robot is result of various time variant wheel rotation speeds. Planar curvilinear motion is a sum of linear motion (translation) and rotation motion.

Forces balance is the initial point for the derivation of motion equations. If '*F'* is actual force acting to a mass point with weight 'm' and with distance '*r'* from axis of rotation, then it holds true for general curvilinear motion that vector sum of all forces acting to a selected point is zero [33].

$$F = m\frac{dv}{dt} + m\frac{d\omega}{dt} + 2m\omega \times \frac{dr}{dt} + m\omega \times (\omega \times \mathrm{r}) = 0 \qquad (3.9)$$

Where

$m\frac{dv}{dt}$,$m\frac{d\omega}{dt}$, $2m\omega \times \frac{dr}{dt}$, $m\omega \times (\omega \times \mathrm{r})$ is inertial force, Euler's force, Coriolis force, centrifugal force respectively.

Application of this general equation requires specification of individual forces according to actual conditions and/or eventually implementing other acting forces. The forces created by motion of real body induced with resistances (losses) in addition to curvilinear motion forces is considered. These forces are approximated in simplest manner to be proportional to a speed. Actual wheel motor voltages are result of the dynamic part. Equivalent circuit of motor selection of the point where actual translation and rotation speed is evaluated influences significantly initial equations and hence complexity of the resulting model. Centre of the join between wheels is used as the selected chassis point in common literature. This choice leads to easiest recalculation of actual wheel speeds to motion equations of that point. Point as center of gravity projection to join between wheels is selected.

The differential drive wheel agricultural robot described is driven by two DC motors with common voltage source and independent control of each motor. Motors are connected with driving wheels through gear-box with constant gear ratio. Ideal gear-box means that it reduces linearly angular speed and increases the moment. Loses in motor and also in gear-box are proportional to rotation speed. The chassis is connected with caster wheel with no impact to chassis movement. Its influence is included in resistance coefficients acting against motion.

In this DC motors connected in series Motion equations are presented and equations which show dependency between translation and rotation speed of the selected point on moments acting to

driving wheels are evaluated. The model is transformed to simpler form which is more suitable for next using and for trajectory of arbitrary point calculation.

3.4.1. DC motor in series connection dynamics

DC motors are commonly used in robotic applications and are the main types of actuators used in mobile robots. Consequently, it is important to analyze and integrate their dynamics into the robot model. Current Controlled. In a Field-Current Controlled motor, the armature current $i_a$is kept constant while the field-current is controlled using field voltage commands v_f .

On the other hand, in a Armature-Current Controlled motor, the armature voltage $v_a$is the command to control the armature current $i_a$while keeping the field-current i_f constant. Armature-current controlled DC motors are more common choice in mobile robots.

It is circuit with resistance and inductance connected in series way. The input voltage v_a(t) is the voltage supplied by amplifier to move the motor. The back EMF v_b(t) is induced by the rotation of the armature windings in the fixed magnetic field. To derive the model, the DC motor, the system is divided into three major parts of equation. Those are electrical equation, mechanical equation and electro-mechanical equation [34].

3.4.2. Electrical equation of DC series motor

The differential equation (electrical equation) for the equivalent circuit can be derived by applying the Kirchhoff voltage laws around the electrical loop. Kirchhoff's voltage law states that, the sum of all voltages around the loop must be equal to zero. Field current is generally kept constant and steady state.

Armature circuit voltage equation from KVL is shown below.

$$v_a - v_{Ra-} v_{La} - v_b = 0 \tag{3.10}$$

Where,

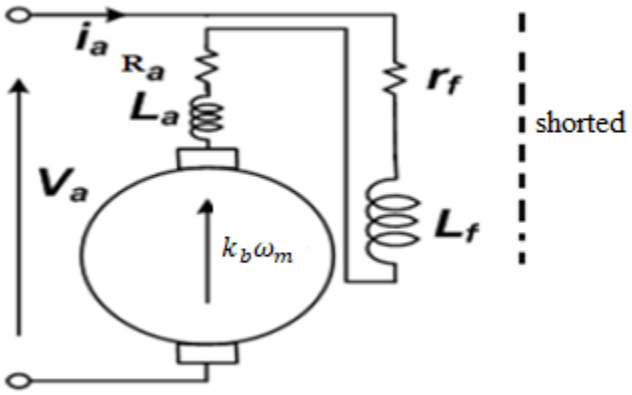


Figure 3.6: Electrical series DC motor [35]

Based on ohm's law, the voltage across the resistor can be represented as:

$$v_{Ra} = i_a(t)R_a \tag{3.10a}$$

Where,i_a is the armature current. The voltage across the inductor is proportional to the current change through the coil with respect to time and can be written as:

$$v_{La} = L_a \frac{di_a}{dt} \tag{3.10b}$$

Where, La is the inductance of the armature coil. At the end, the back emf can be formulated as:

$$v_b = k_b\omega_m(t) \tag{3.10c}$$

Where k_b is the velocity constant, determined by the flux density of the permanent magnets, the reluctance of the iron core of the armature, and the number of turns of the armature winding ω_m is the rotational velocity of the armature.

Substituting eqns. (3.10a), (3.10b) and ((3.10c) into eqn. (3.10) gives the following differential equation:

$$v_a - i_aR_a - L_a \frac{di_a}{dt} - k_b\omega_m = 0 \tag{3.11}$$

3.4.3. Mechanical equation of DC series motor

The mechanical equation (Newton low of motion) is obtained as follows:

$$\tau_m(t) = \frac{J_m\, d^2\theta(t)}{dt^2} + \frac{\mathrm{k_r} d\theta(t)}{dt} \tag{3.12}$$

Where, τ_m is the motor torque produced by the motor shaft, J_m and $\mathrm{k_r}$ are denoted as the moment of inertia and motor friction coefficient.

Based on the previous two equations (3.11) and (3.12), when the input voltage $v_a(t)$,is applied, the armature current $i_a(t)$goes through resistance R_a and inductance L_a producing magnetic flux and causing the motion of the rotor according to the motor torque as illustrated in equation (3.13).

$$\tau_m\,(t) = k_t\, i_a(t) \tag{3.13}$$

Where k_t is the torque constant and like the velocity constant, it depends on the flux density of the fixed magnets, the reluctance of the iron core, and the number of turns in the armature winding.

The equations (3.10), (3.11), (3.12) and (3.13) are combined as follow:

$$L_a \frac{di_a(t)}{dt} + R_a i_a(t) = v_a(t) - k_b \frac{d\theta(t)}{dt} \tag{3.14}$$

$$\frac{J_m\, d^2\Theta(t)}{dt^2} + \frac{\mathrm{k_r} d\theta(t)}{dt} = k_t\, i_a(t) \tag{3.15}$$

The mechanical equation does not involve load torque but, the following state space representation consists load torque.

3.4.4. State Space Representation of the DC series motor

The above differential questions can also represented by state space model which consisting load torque that makes the state space to have two inputs (a load torque (M) and terminal voltage V_a). Commutator is not considered. Rotor produces electrical voltage with reverse polarity than source voltage electromotive voltage($k_b\omega(\mathrm{t})$) which is proportional to rotor angular velocity. Twisting moment of the rotor M_m is proportional to current $i_a(t)$. Ideal behavior means that whole electric energy used to magnetic field creating is transformed without any loses to mechanical energy moment of the motor. Loses in magnetic field are not considered, but only electric loses in winding and mechanical loses proportional to rotor speed are considered.

$$R_a i_a(t) + \frac{L_a \mathrm{d} i_a(t)}{\mathrm{dt}} = v_a(\mathrm{t}) - k_b \omega_m(\mathrm{t})$$

$$\frac{J_m\, \mathrm{d}\omega_m(\mathrm{t})}{\mathrm{dt}} + \mathrm{k_r}\omega_m(\mathrm{t}) + \mathrm{M} = k_t i_a(t)$$

$$\frac{\mathrm{d}i_a(t)}{\mathrm{dt}} = \frac{v_a(\mathrm{t})}{L_a} - \frac{k_b\omega_m(\mathrm{t})}{L_a} - \frac{R_a i_a(t)}{L_a} \tag{3.16}$$

$$\frac{\mathrm{d}\omega_m(\mathrm{t})}{\mathrm{dt}} = \frac{\mathrm{K}i_a(t)}{J_m} - \frac{\mathrm{k_r}\omega_m(\mathrm{t})}{J_m} - \frac{\mathrm{M}}{J_m} \tag{3.17}$$

The rotor position results 3rd order differential equation of the motor model.

$$\frac{d\theta(t)}{dt} = \omega_m(\mathrm{t}) \tag{3.18}$$

The state space representation of the differential equations (3.16), (3.17), (3.18) is shown as fellow.

$$\frac{d}{dt}\begin{bmatrix} i_a \\ \omega_m \\ \Theta \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{k_b}{L_a} & 0 \\ \frac{k_b}{J_m} & -\frac{\mathrm{k_r}}{J_m} & 0 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} i_a \\ \omega_m \\ \Theta \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & -\frac{1}{J_m} \\ 0 & 0 \end{bmatrix}\begin{bmatrix} v_a \\ M \end{bmatrix} \tag{3.19}$$

$$\begin{bmatrix} \omega_m \\ \Theta \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} i_a \\ \omega \\ \Theta \end{bmatrix}$$

Where, $\begin{bmatrix} \omega_m \\ \Theta \end{bmatrix}$ are outputs and $\begin{bmatrix} v_a \\ M \end{bmatrix}$ are inputs.

Table 1: DC motor parameters selected based on matlab simulation

Notation	Value	Unit(dimension)	Meaning
R_a	4.000	Ω	Motor winding resistivity
L_a	0.025	H	Motor inductance
R_Z	0.4	Ω	Source resistance
v_a	24	V	Source voltage
J_m	0.0025	kg.m^2	Total moment of inertia of rotor and gearbox
k_r	0.00005	N.ms	Coefficient of the resistance against rotating of rotor and gearbox
p_G	0.25	------	Gearbox transmission ratio

+

Inserting the parameters values in to equation (3.19)

$$\frac{d}{dt}\begin{bmatrix} i_a \\ \omega_m \\ \theta \end{bmatrix} = \begin{bmatrix} -160 & 2.548 & 0 \\ 254.8 & -20 & 0 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} i_a \\ \omega_m \\ \theta \end{bmatrix} + \begin{bmatrix} 40 & 0 \\ 0 & -4000 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} v_a \\ M \end{bmatrix}$$

$$\begin{bmatrix} \omega_m \\ \theta \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} i_a \\ \omega_m \\ \theta \end{bmatrix}$$

The DC series motor represented by the state space above can also represented with two inputs and one output as shown below.

$$\frac{d}{dt}\begin{bmatrix} i_a \\ \omega_m \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{k_b}{L_a} \\ \frac{k_b}{J_m} & \frac{-\mathrm{k_r}}{J_m} \end{bmatrix}\begin{bmatrix} i_a \\ \omega_m \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & -\frac{1}{J_m} \end{bmatrix}\begin{bmatrix} v_a \\ M \end{bmatrix} \quad (3.20)$$

$$\omega_m = [0 \quad 1]\begin{bmatrix} i_a \\ \omega_m \end{bmatrix}$$

Then, transforming the above two equations using Laplace transformation obtained the two equations as follow.

$$L_a s i_a(s) + R_a i_a(s) = v_a(s) - k_b s\theta(s) \quad (3.21)$$

$$J_m s^2\theta(s) + \mathrm{k_r} s\theta(s) = k_t i_a(s) \quad (3.22)$$

The transfer function of the motor speed is:

$$G_{speed}(s) = \frac{s\theta(s)}{V_a(s)} = \frac{k_t}{J_m L_a s^2 + (L_a \mathrm{k_r} + R_a J_m)s + \mathrm{k_r} R_a + k_t R_a} \text{ .at load torque equal to zero (M=0)} \quad (3.23)$$

In addition, the transfer function of the motor position is determined by multiplying the transfer function of the motor speed by the term $\frac{1}{s}$

$$G_{position}(s) = \frac{\theta(s)}{v_a(s)} = \frac{k_t}{J_m L_a s^2 + L_a \mathrm{k_r} + R_a J_m + + \mathrm{k_r} R_a + k_t R_a} \quad (3.24)$$

Inserting the motor parameter values

$$G_{speed}(s) = \frac{\omega_m(s)}{v_a(s)} = \frac{7e{-}05}{s\text{^}2 + 0.01\,s + 5e{-}05} \quad \text{at M} = 0$$

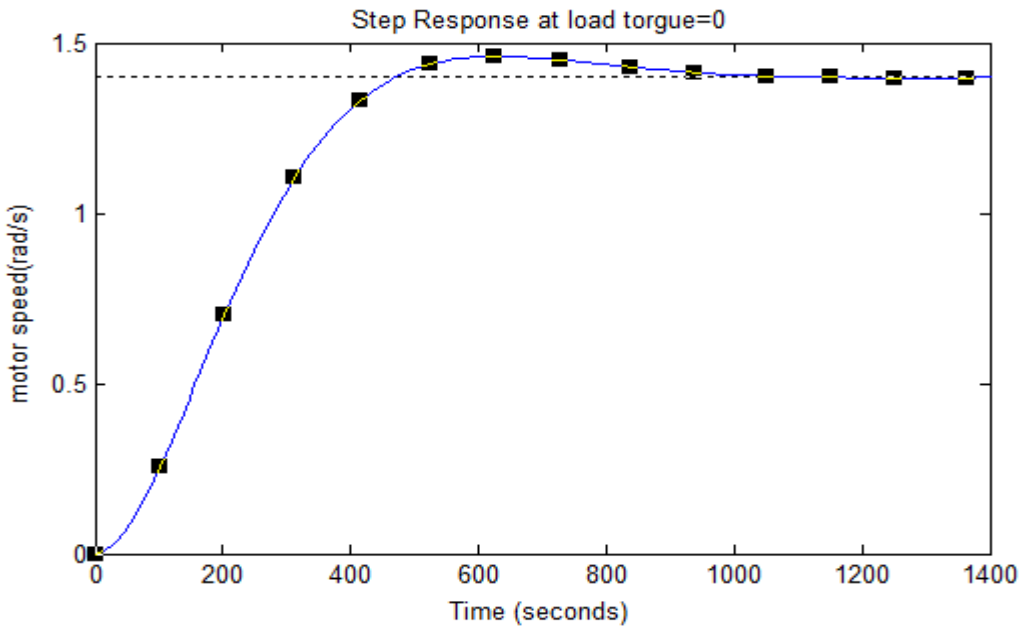


Figure 3.17: Speed step response at no load torque

Motor parameters are selected based on the above graph.Once I modeld the motor I have used two idenical DC series motors in the robot. The Dynamics of the two series DC motor used in the system can be derived from balancing of voltages (Kirchhoff's law) and balancing of moments. From Kirchhoff's voltage law, as

$$R_a i_L(\mathrm{t}) + \mathrm{Rz}\left(i_L(\mathrm{t}) + i_R(t)\right) + \frac{L_a \mathrm{d}i_L(\mathrm{t})}{\mathrm{dt}} = u_L V_a - k_b \omega_L(\mathrm{t}) \quad (3.25)$$

$$R_a i_R(\mathrm{t}) + \mathrm{Rz}\left(i_L(\mathrm{t}) + i_R(t)\right) + \frac{L_a \mathrm{d}i_R(t)}{\mathrm{dt}} = \mathrm{u}_R V_a(\mathrm{t}) - k_b \omega_R(\mathrm{t}) \quad (3.26)$$

Where,

Rz [Ω] is internal resistance

R_a [Ω] is motor winding resistance

L_a [H] is motor inductance

k_b [V /m s^{-1}] is the back EMF constant

V_a [V] is source voltage

ω_L [rad. s^{-1}] is left motor angular velocity

ω_R [rad. s^{-1}] is right motor angular velocity

u_R and u_L are the control voltages of the right and left motors respectively. i_L and i_R [A] are left and right current flowing through winding. Each motor has its own power supply voltage (U_L, U_R) disbranched from the common source of voltage V_a Control of the supply voltage of both motors using amplifier with control (u_R and u_L) is shown in Figure.3.8. Because both engines are powered from the common source it was taken into account also effect of the internal resistance R_Z .Both motors are considered with the same parameters and the second equation is balance of moments (electric energy) moment of inertiaM_S, rotation resistance proportional to rotation speed (mechanical loses) M_O, load moment of the motor M_x and moment M_m caused by magnetic field which is proportional to current.

$$M_S + M_O + M_O = M_m$$

$$\frac{J_m \, d\omega_L(t)}{dt} + k_r \omega_L(t) + M_L = k_b i_L(t) \quad (3.27)$$

$$\frac{J_m \, d\omega_R(t)}{dt} + k_r \, \omega_R(t) + M_R = k_b i_R(t) \quad (3.28)$$

Where

J_m [$kg.m^2$] is moment of inertia.

k_r [N.ms] is coefficient of rotation resistance and M_x [$kg.m^2 .s^{-2}$] is load moment.

M_L and M_R are the load moments on left and right wheels respectively.

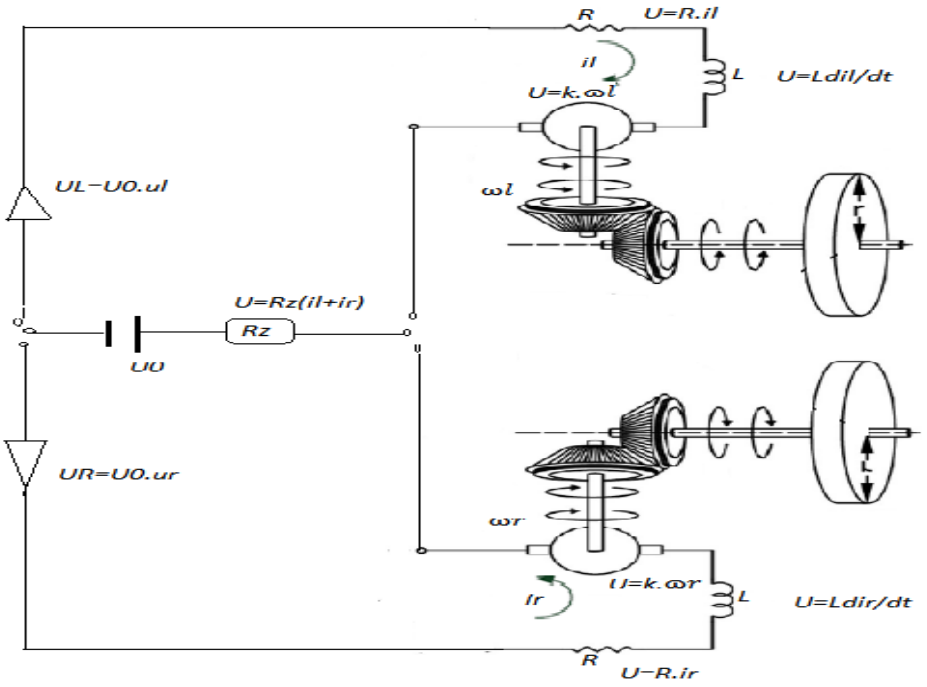


Figure 3.18: DC motor wiring of the robot

3.4.5. Chassis dynamics

It is defined with a vector of linear velocity v_B acting on a chassis reference point and with rotation of this vector of angular velocity ω_B (constant for all chassis points). The chassis reference point 'B' is the point of the intersection of the axis joining the wheels and center of gravity normal projection see figure below. The point 'T' is common enter of gravity which is placed at the center of the axis joining the wheels.

It is possible to calculate trajectory of arbitrary chassis point from these variables. Point 'B' for which the equations are derived is center of gravity normal projection to join between wheels. This leads to easiest set of equation for whole model. Forces balances are considered 'as starting equations. It is possible to replace two forces $'F_L'$ and $'F_R'$ acting to chassis in left (L) and right (R) wheel ground contact points with one force $'F_B'$ and torsion moment *Mb* acting in point 'B'. Chassis is characterized with semi-diameter of the driving wheels '*r*', total weight '*m*', moment of inertia 'J_T 'with respect to center of gravity located with parameters $L_{T,}\ L_L, L_R$.

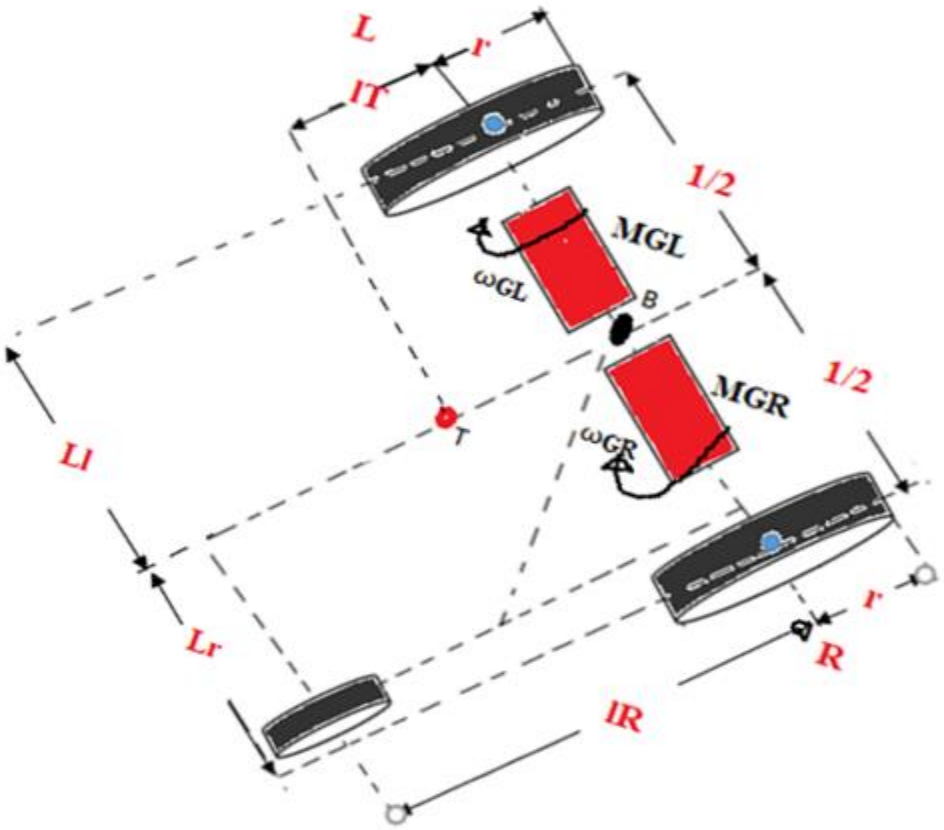


Figure 3.19: Chassis Scheme and Forces

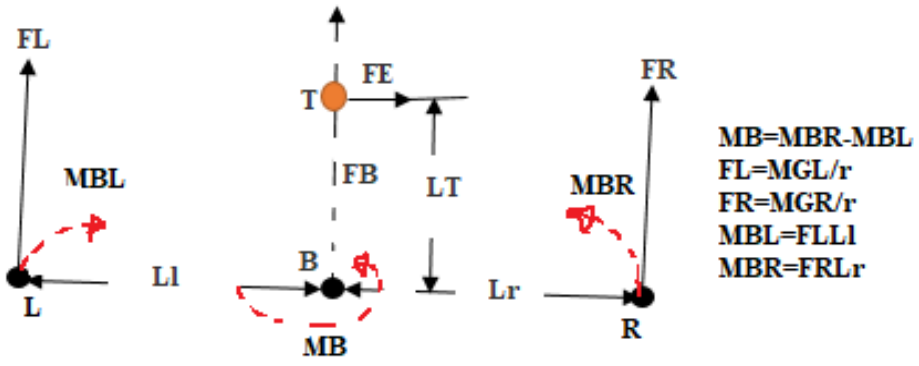


Figure 3.10: Chassis Scheme and Forces

Let specify equation (3.9) for this case. Position of the center of gravity is constant with respect to axis of rotations. Coriolis force is not needed. Coriolis force has to be considered for example if the chassis moves on rotating surface. Similarly, centrifugal force is not considered. Chassis is supposed to be solid body represented as mass point (center of gravity). Because force vector causing the movement acts in point b and goes through center of gravity. It is enough if we consider

inertial force by linear motion. By rotational motion it is necessary to consider moment caused with Euler's force because the axis of rotation does not go through the center of gravity.

By the balance of forces causing linear motion, consider except of forces F_L, F_R caused by drives and inertial force 'F_S' also resistance force F_O proportional to speed v_B. The balance of all forces influencing linear motion is expressed as fellow.

$$F_L + F_R + F_O + F_S = 0$$
$$\frac{MG_L}{r} + \frac{MG_R}{r} - k_v v_B - \frac{md v_B}{dt} = 0 \qquad (3.29)$$

Where,

m[kg] is robot mass.

K_v [kg. s^{-1}] is resistance coefficient against linear motion.

MG_L [kg.m^2. s^{-2}] is moment of the left drive.

MG_R [kg.m^2. s^{-2}] is moment of the right drive.

v_B [m/s] is linear motion speed.

r[m] is semi-diameter of the wheels.

Balance of moments is slightly more complicated because the rotation axis does not lie in center of gravity. That's why it is necessary to take into account except chassis momentum M_T also moment $M_E = l_t F_E$ caused by Euler's forceF_E. Similarly, as by linear motion is considered moment M_O caused with resistance against rotation to be proportional to angular velocityω_B.

$$MB_L + MB_R + M_O + M_T + M_E = 0$$
$$-\frac{MG_L L_L}{r} + \frac{MG_R L_R}{r} - K_\omega \omega_B - \frac{J_T d\omega_B}{dt} - \frac{L_T m d\omega_B}{dt} = 0 \qquad (3.30)$$

Where

L_R [m] is distance of the right wheel from point B,

L_L [m] is distance of the left wheel from point B,

L_T [m] is distance of the center of gravity from point B,

K_ω [kg.m^2. s^{-1}] is resistance coefficient against rotary motion

J_T [kg.m^2] is moment of inertia with respect to rotation axis in center of gravity and ω_B [rad.s^{-1}] is angular speed in point B .Resulting moment of inertia J_B with respect to rotation axis in Point B is given by equation below which is parallel axis theorem or Huygens-Steiner theorem [33].

$J_B = J_T + m_B L_T$

Where, J_T [kg.m^2] is moment of inertia with respect to center of gravity and L_T [m] is distance between center of gravity and point B.

3.4.6. Relationship between rotation speed of the motor and center of gravity chassis movements

The equation describing the behavior of the two motors (currents and angular velocity) and the behavior of the chassis (the speed of the linear movement and speed of the rotation) are connected only through moments of engines. Equations express the law of conservation of energy which is conversion of electrical energy to mechanical including one type of losses but represent only one relationship between the speed of the two motors (peripheral speed of the drive wheels) and rates of movement and rotation of the chassis. Additional relation is given by design of the drive and chassis. Both drive wheels are firmly linked to rotors of relevant engines over ideal gearbox with gear ratio P_G without nonlinearities and any flexible members. Gearbox decreases output angular velocity ωG_x with relation to the input angular speed ω_x according to the transmission ratio P_G and simultaneously in the same proportion increases output torque MG_x with relation to the input torqueM_x.

$$\omega G_L = \frac{\omega_L}{P_G}, \omega G_R = \frac{\omega_R}{P_G} \tag{3.31}$$

$$MG_L = P_G M_L, MG_R = P_G M_R \tag{3.32}$$

The chassis dynamics can be expressed by the combination of balance of the forces and balance of the moment. Equation (3.33) is the result of applying balance of forces and Equation (3.34) from balance of moments.

$$\left(\frac{P_G M_L}{\mathrm{r}} + \frac{P_G M_R}{\mathrm{r}}\right) - k_v v_B - \frac{\mathrm{mdv}}{\mathrm{dt}} = 0$$

$$M_L + M_R - r_G k_v v_B - r_G \frac{\mathrm{mdv}}{\mathrm{dt}} = 0 \tag{3.33}$$

$$-\frac{L_L P_G M_L}{\mathrm{r}} + \frac{L_R P_G M_R}{\mathrm{r}} - \mathrm{k}\omega\omega_B - \frac{(\mathrm{JT} + \mathrm{mLT\hat{}2})\mathrm{d}\omega_B}{\mathrm{dt}} = 0$$

$$-L_L M_L + L_R M_R - r_G k_\omega \omega_B - \frac{r_G J_B \mathrm{d}\omega_B}{\mathrm{dt}} = 0 \tag{3.34}$$

Where, P_G is the gear box transmission ratio, k_ω is the resistance coefficient against rotational motion. The rest parameters are shown in figure 2. The parameters r_G and J_B are described as,

$$r_G = \mathrm{r}/P_G \tag{3.35}$$

$$J_B = J_\mathrm{T} + \mathrm{m}L_T{}^2 \tag{3.36}$$

Both drive wheels have the same radius r and their peripheral speedsv_L, v_R depend on the angle speeds of gearbox output ωG_L, ωG_R according to relations:

$$v_L = r\omega G_L = r\omega_L/P_G \tag{3.37}$$

$$v_R = r\omega G_R = r\omega_R/P_G \tag{3.38}$$

To determine the value of the linear speed in point b and the angular velocity of rotation let us start from Figure 3.11, Both drive wheels have the same axis of rotation and therefore their peripheral speeds are always parallel. The illustration shows the positioning where the peripheral speeds $\mathrm{v_L}$and $\mathrm{v_R}$ actually operate (driving wheels L and R) and the point B. Specifying such a linear speed $\mathrm{v_B}$ and angular speed that have the same effect as the action of the peripheral speed of the driving wheels. From the theorems of similar triangles, depicted in figure 3.11 we can recalculate the peripheral velocities of the wheelsv_L, v_R to the linear velocity $\mathrm{v_B}$ and angular velocity ω_B at point B as

$$v_B = (v_L L_R + v_R L_L)/L_L + L_R = r/P_G(L_L + L_R)(L_R\omega_L + L_L\omega_R)$$

$$\omega_B = v_B\,/(x + L_L) = (v_R - v_L)/L_L + L_R = r/P_G(L_L + L_R)(-\omega_L + \omega_R)$$

$$v_B = \frac{r_G}{L_L+L_R}(L_R\omega_L + L_R\omega_R) \tag{3.39}$$

$$\omega_B = \frac{r_G}{L_L+L_R}(-\omega_L + \omega_R) \tag{3.40}$$

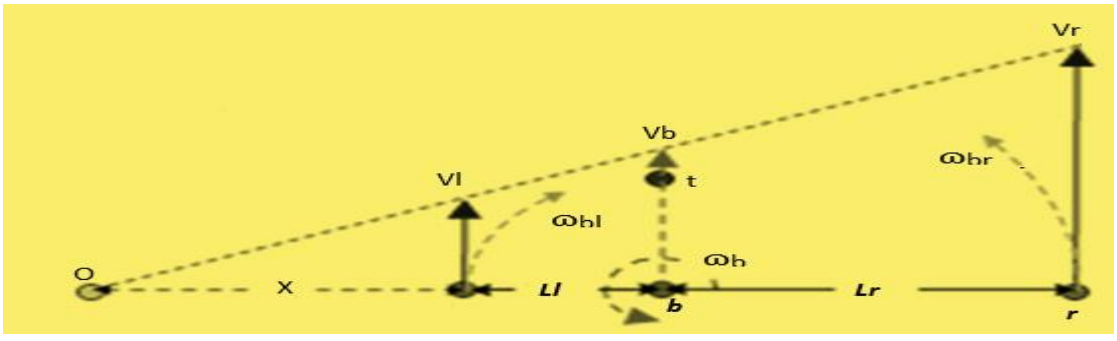


Figure 3.11: Linear and Angular Velocity Recalculation

3.4.7. Kinematics of the mobile robot

The following derivations closely follow despite some notation changes which have been used. Let the global coordinates of the robot be (x_B,y_B), the orientation of the robot be α_B, and v_B, ω_B are the linear and angular velocities.

$$\frac{dx_B}{dt} = v_B \cos\alpha_B \qquad (3.41a)$$

$$\frac{dy_B}{dt} = v_B \sin\alpha_B \qquad (3.41b)$$

$$\frac{d\alpha_B}{dt} = \omega_B \qquad (3.41c)$$

This can be represented as a simple model

$$\frac{dx_B}{dt} = f(x_B, u_B) \qquad (3.42)$$

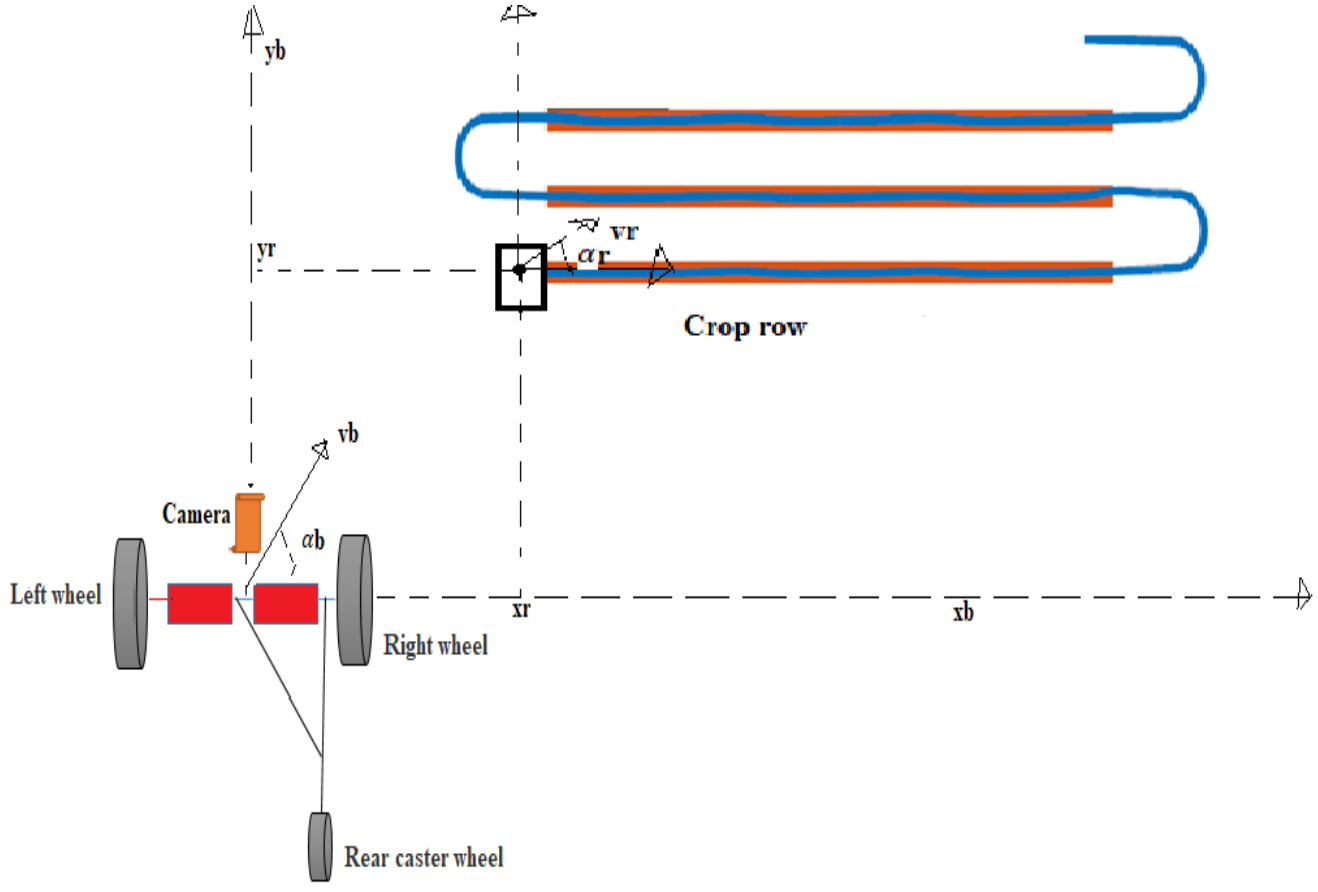


Figure 3.12: Coordinate System of Real Robot and Reference Robot

Where state variables $X_B = [x_B \ y_B \ \alpha_B]^T$ and control inputs, $U_B = [v_B \ \omega_B]^T$

Linear model can be derived from the non-linear model, Equations (3.42), from an error model with respect to the reference robot (see figure 3.12). A reference robot can be considered as a robot with reference (desired) parameters of the robot to follow a trajectory which can be represented as

$$\frac{dx_r}{dt} = f(x_r, u_r) \tag{3.43}$$

The reference parameters are $[x_r \ y_r \ \alpha_r \ v_r \ \omega_r]$. The linear velocity, orientation angle and angular velocity of the reference robot can be derived from Equation (3.41) as

$$v_r(t) = \sqrt{\left(\frac{dx_r(t)}{dt}\right)^2 + \left(\frac{dy_r(t)}{dt}\right)^2} \tag{3.44}$$

$$\alpha_r = arctan^2\left(\frac{dy_r(t)}{dt}, \frac{dx_r(t)}{dt}\right) \tag{3.45}$$

$$\omega_r(t) = \frac{d\alpha_r r(t)}{dt} = \frac{\frac{dx_r(t)}{dt} * d^2y_r(t)^2 - d^2x_r(t)^2 * \frac{dy_r(t)}{dt}}{\left(\frac{dx_r(t)}{dt}\right)^2 + \left(\frac{dy_r(t)}{dt}\right)^2} \tag{3.46}$$

Applying the Taylor series approximation to Equation (3.42), around the reference points (x_r, u_r)

$$\frac{dx(t)}{dt} = f(x_r, u_r) + \frac{\partial f(x,u)}{\partial x}(x_B - x_r) + \frac{\partial f(x,u)}{\partial x}(u_B - u_r) \tag{3.47}$$

Evaluated at $u = u_r, x = x_r$ Subtracting Equation (3.47) from Equation (3.43) gives

$$\frac{d\dot{x}}{dt} = fx_r.x + fu_r.u \tag{3.48}$$

$\dot{x}$ is the error vector of state variables and u is the error vector of control variables with respect to the reference robot. The approximation of $\frac{d\dot{x}}{dt}$ in Equation (3.48), by the forward differences gives the following discrete-time linear time-variant (LTV) state-space model.

$$xk(k+1) = Ak(k)xk(k) + Bk(k)uk(k) \tag{3.49}$$

$$yk(k) = ckxk(k) \tag{3.50}$$

Where,

$$Ak(k) = \begin{bmatrix} 1 & 0 & -v_r(k)\sin\alpha_r(k)T \\ 0 & 1 & v_r(k)\cos\alpha_r(k)T \\ 0 & 0 & 1 \end{bmatrix}$$

$$Bk(k) = \begin{bmatrix} cos\alpha_r(k)T & 0 \\ sin\alpha_r(k)T & 0 \\ 0 & T \end{bmatrix}$$

$$Ck(k) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$xk = \begin{bmatrix} y_B(k) - x_r(k) \\ y_B(k) - y_r(k) \\ \alpha_B(k) - \alpha_r(k) \end{bmatrix}$$

$$uk = \begin{bmatrix} v_B(k) - v_r(k) \\ \omega_B(k) - \omega_r(k) \end{bmatrix}$$

Where T is the sampling period and xk is deviation state vector which represents the error with respect to the reference robot, and uk is associated with the control input. The reference values, v_r, α_r, ω_r are the reference linear velocity, orientation angle and angular velocity respectively which can be calculated from Equations (3.44-3.46).

3.5. Computational form of the model

A mathematical model is used in particular for the design and simulation validation of control movement of the robot. From the control point of view control action variables are signals u_L and u_R that control the supply voltage of the motors. Momentary speed v_B and speed of rotation ω_B are output variables from linear part of the model. These variables are the inputs to the consequential non-linear part of the model, whose outputs are controlled variables the coordinates of selected point position x_B, y_B and the rotation angle of the chassis α. The last part is the calculation of coordinates of the position of arbitrary points of the chassis. Linear part of the model can be modified into simpler form for control design purposes to reduce number of differential equations from six to four. Substitute equations (3.39,3.40) into (3.33,3.34) and eliminate moments M_L and M_R by substitution of (3.27, 3.28) to (3.33, 3.34) we can able to reduce four differential equations (3.27, 3.28) to (3.33, 3.34) .These six differential Equations (3.25, 3.26) -(3.33, 3.34), and two algebraic Equations (3.39, 3.40) containing eight state variables represent a mathematical description of the dynamic behavior of ideal differential drive wheel agricultural robot with losses linearly dependent on the revolutions or speed. Control signals, u_L and u_R, that control the supply voltages of the motors are input variables. To put the equation in state space, new parameters are introduced as follow.

$$a_r = k_r + \frac{k_v * L_L * {r_G}^2}{L_L + L_R}$$

$$a_l = k_r + \frac{k_v * L_R * {r_G}^2}{L_L + L_R}$$

$$b_l = J_m + \mathrm{m} * L_R * \frac{{r_G}^2}{L_L + L_R}$$

$$b_r = J_m + \mathrm{m} * L_L * \frac{{r_G}^2}{L_L + L_R}$$

$$c_l = k_v * L_L + \frac{k_\omega * {r_G}^2}{L_L + L_R}$$

$$c_r = k_r * L_R + \frac{k_\omega * {r_G}^2}{L_L + L_R}$$

$$d_l = J_m * L_L + \frac{J_b * {r_G}^2}{L_L + L_R}$$

$$d_r = J_m * L_R + \frac{J_b * {r_G}^2}{L_L + L_R}$$

$$\frac{\mathrm{dxd}}{\mathrm{dt}} = \mathrm{Adxd} + \mathrm{Bdu}, \mathrm{yd} = \mathrm{Cdxd} \tag{3.51}$$

Where,

$$\mathrm{u} = [u_L \;\; u_R]^{\mathrm{T}}$$

$$\mathrm{xd} = [i_L \;\; i_R \;\; \omega_L \;\; \omega_R \;]^T$$

$$\mathrm{Yd} = [v_B \;\; \omega_B]\mathrm{T}$$

With matrices $Ad, Bd \; and \; Cd$ as

$$\mathrm{Ad} =$$

$$\begin{bmatrix} \frac{-(R_a + \mathrm{Rz})}{L_a} & -\frac{\mathrm{Rz}}{L_a} & -\frac{k_b}{L_a} & 0 \\ -\frac{\mathrm{Rz}}{L_a} & \frac{-(R_a + \mathrm{Rz})}{L_a} & 0 & -\frac{k_b}{L_a} \\ k_b \frac{d_r + b_r L_L}{b_l d_r + b_r d_l} & k_b \frac{d_r - b_r L_R}{b_l d_r + b_r d_l} & \frac{d_r a_l + b_r c_l}{b_l d_r + b_r d_l} & \frac{-d_r a_r - \; b_r c_r}{b_l d_r + b_r * d_l} \\ k_b \frac{d_l - b_l L_L}{b_l d_r + b_r d_l} & \frac{d_l + b_l L_R}{b_l d_r + b_r d_l} & \frac{-d_l a_l - \; b_l c_l}{b_l d_r + b_r \mathrm{dl}} & \frac{-d_l a_r + b_l c_r}{b_l d_r + b_r d_l} \end{bmatrix}$$

$$Bd = \begin{bmatrix} \frac{\mathrm{va}}{L_a} & 0 \\ 0 & \frac{\mathrm{va}}{L_a} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\mathrm{Cd} = \begin{bmatrix} 0 & 0 & r_G \frac{L_R}{(L_L + L_R)} & r_G \frac{L_L}{L_L + L_R} \\ 0 & 0 & -\frac{r_G}{L_L + L_R} & \frac{r_G}{L_L + L_R} \end{bmatrix}$$

A linearized discrete error model is obtained from the kinematic model and the dynamic model is converted to a discrete error based model. Since the dynamic model is linear time invariant, the error model will be the same as that of Equation (18). The following equation is the discrete time state-space dynamic model

$$\mathrm{xd(k+1) = Ad(k)xd(k) + Bd(k)ud(k)} \quad (3.52)$$

$$\mathrm{yd(k) = Cd(k)xd(k)} \quad (3.53)$$

The matrices $Ad, Bd\ and\ Cd$ are discretized matrices of the dynamic model (Equation (3.51)). The state variables and control inputs are deviation variables from the reference points, $xdr\ and\ udr$, as $x_B, y_B, \alpha, i_L, i_R, \omega_L, \omega_R, v_B, \omega_B$.

$$\mathrm{Xd(k)} = \begin{matrix} \mathrm{i_L(k) - i_{Lr}(k)} \\ \mathrm{i_R(k) - i_{Rr}(k)} \\ \mathrm{\omega_R(k) - \omega_{Rr}(k)} \\ \mathrm{\omega_L(k) - \omega_{Lr}(k)} \\ \mathrm{ud(k) = u_L(k) - u_{Lr}(k)} \\ \mathrm{u_R(k) - u_{Rr}(k)} \end{matrix}$$

This dynamics model and linearized kinematic time variant model, Equation (3.49, 3.50), can be combined into a single state-space time-variant model with nine states (currents, wheel speeds, linear and angular velocities and coordinates), two control variables (motor voltage control input) and three outputs (position in x and y direction and orientation measured from x direction).

$$xk(k+1) = Ak(k)xk(k) + Bk(k)uk(k) \quad (3.54)$$

$$yk(k) = C(k)x(k) \quad (3.55)$$

Where, $Ak(k) = \begin{bmatrix} Ad(4x4) & 0(4x5) \\ Cd(2x4) & 0(2x5) \\ 0(3x4) \quad Bk(3x2) & Ak(3x3)] \end{bmatrix}$, $Bk(k) = \begin{bmatrix} Bd(4x2) \\ 0(5x2)] \end{bmatrix}$, $Ck(k) = [0(3x6) \quad Ck(3x3)]$

$x = [xd(k) \quad uk \quad xk]^T, \ u = ud(\mathrm{k})$

Or

Ak(k)

$$= \begin{bmatrix} \frac{-(R_a+Rz)}{L_a} & -\frac{Rz}{L_a} & -\frac{k_b}{L_a} & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{Rz}{L_a} & \frac{-(R_a+Rz)}{L_a} & 0 & -\frac{k_b}{L_a} & 0 & 0 & 0 & 0 & 0 \\ k_b\frac{d_r+b_rL_L}{b_ld_r+b_rd_l} & k_b\frac{d_r-b_rL_R}{b_ld_r+b_rd_l} & \frac{d_ra_l+b_rc_l}{b_ld_r+b_rd_l} & \frac{-d_ra_r-\ b_rc_r}{b_ld_r+b_r*d_l} & 0 & 0 & 0 & 0 & 0 \\ k_b\frac{d_l-b_lL_L}{b_ld_r+b_rd_l} & \frac{d_l+b_lL_R}{b_ld_r+b_rd_l} & \frac{-d_la_l-\ b_lc_l}{b_ld_r+b_r\mathrm{dl}} & \frac{-d_la_r+b_lc_r}{b_ld_r+b_rd_l} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & r_G\frac{L_R}{(L_L+L_R)} & r_G\frac{L_L}{L_L+L_R} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{r_G}{L_L+L_R} & \frac{r_G}{L_L+L_R} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & cos(ar*T) & 0 & 1 & 0 & -vr*sin(ar*T) \\ 0 & 0 & 0 & 0 & sin(ar*T) & 0 & 0 & 1 & cos(ar*T) \\ 0 & 0 & 0 & 0 & 0 & T & 0 & 0 & 1 \end{bmatrix}$$

$$Bk(k) = \begin{bmatrix} \frac{v_a}{L_a} & 0 \\ 0 & \frac{v_a}{L_a} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \mathrm{C(k)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, D(k) = [0]$$

Values of the parameters listed in the following tables are used in all of the calculation.

Table 2: Chassis parameters

Notation	*Value*	Dimension	Meaning
L_L	0.5	m	Distance of the left wheel from point B
L_R	0.5	m	Distance of the right wheel from point B
L_T	0.40	m	Distance of center of gravity from join between wheels
L_k	0.40	m	Distance of caster wheel from join between wheels
m	2	kg	Total weight of the robot

k_v	0.0005	kg.s-1	Coefficient of the resistance against robot linear motion
j_t	0.550	kg.m^2	Moment of inertia of robot with respect to center of gravity
k_ω	0.0350	kg.m^2.s-1	Coefficient of the resistance against robot rotating

Inserting the parameter values into state space model of the robot verified above the state matrix, output state matrix and input matrix are shown below.

$$A = \begin{bmatrix} -176 & -16 & -0.0028 & 0 & 0 & 0 & 0 & 0 & 0 \\ -16 & -176 & 0 & -0.0028 & 0 & 0 & 0 & 0 & 0 \\ 0.3312 & 0.3312 & 0.3196 & 0.3428 & 0 & 0 & 0 & 0 & 0 \\ 0.3312 & 0.3312 & 0.3428 & 0.3196 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.1 & 2.1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -3.7 & 3.7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0.8 \\ 0 & 0 & 0 & 0 & 0 & 0.1 & 0 & 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} 960 & 0 \\ 0 & 960 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, D = 0$$

3.5.1. Disturbance affecting the DDWAR

The above state space representation of the differential drive wheel agricultural robot is modeled without consideration of the disturbance .The state space representation of the model with the disturbance is written as :

$$xm(k+1) = Amxm(k) + Bmu(k) + Bd\omega(k) \quad (3.56)$$

$$y(k) = Cmxm(k)$$

Where, ω (k) is the input disturbance.

$$\omega(k) - \omega(k-1) = \epsilon(k) \quad (3.57)$$

Note that from (3.5) the following difference equation is also true:

$$xm(k) = Amxm(k-1) + Bmu(k-1) + Bd\omega(k-1) \quad (3.58)$$

By defining $\Delta xm(k) = xm(k) - xm(k-1)$ and $\Delta u(k) = u(k) - u(k-1)$, then

Subtracting (3.58) from (3.56) leads to

$$\Delta xm(k+1) = Am\Delta xm(k) + Bm\Delta u(k) + Bd(k) \quad (3.59)$$

In order to relate the output $y(k)$ to the state variable $\Delta xm(k)$, deduce that

$$\Delta y(k+1) = Cm\Delta xm(k+1) = CmAm\Delta xm(k) + CmBm\Delta u(k) + CmBd(k) \quad (3.60)$$

Where $\Delta y(k+1) = y(k+1) - y(k)$.

Then the augmented state space of the system with disturbance is given as:

$$\begin{bmatrix} \Delta xm(k+1) \\ y(k+1) \end{bmatrix} = \begin{bmatrix} Am & Om^T \\ CmAm & I_{q\times q} \end{bmatrix} \begin{bmatrix} \Delta xm(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} Bm \\ CmBm \end{bmatrix} [\Delta u(k)] + \begin{bmatrix} Bd \\ CmBd \end{bmatrix} \varepsilon(k) \quad (3.61)$$

$$y(k) = [0m \quad I_{q\times q}] \begin{bmatrix} \Delta xm(k) \\ y(k) \end{bmatrix}$$

So far, there is no doubt that mpc can control this robot system very well under no a disturbances condition. It is impossible to have an ideal system without any disturbance. The disturbance test is very important. It can directly demonstrate the properties of the controllers to engineers. In the disturbance tests, the mass change of the robot tanker with time has been used as a disturbance and added in to the input. So, the graph shown in figure 3.14 simulates the disturbance of the robot and its amplitude is 0.4.

The robot carries herbicide tanker for the purpose of spring the herbicide to protect the vegetables from damage. Since the total mass of the robot is the sum of the mass of the herbicide in the tank and the body of the robot, as the robot starts the operation the mass of the liquid decreases from time to time. The change in mass acts as a disturbance to the system operation.

$$m = m_h + m\text{b} \quad (3.62)$$

Where, m_h is mass of the herbicide in the tank.m is total mass of the robot in operation. m_b is mass of the robot body. Consider the structure of the pipe as shown below.

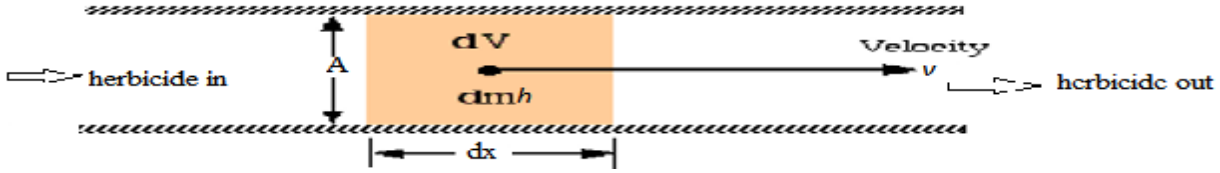


Figure 3.13: Herbicide flow in small pipe

$$dm_h = \rho dV \quad (3.63)$$

$$dV = Adx$$

$$dm_h = \rho A dx \tag{3.64}$$

$$\frac{dm_h}{dt} = \frac{\rho A dx}{dt} = \rho A v \tag{3.65}$$

Where,

$\rho = density\ of\ the\ herbicide$

$dmh/dt = \ Mass\ flow\ rate$

$v = velocity\ of\ the\ herbicide\ flowing$

$V = volume\ of\ the\ herbicide\ flowing$

$A = pipe\ cross\ sectional\ area$

Considering zero initial condition the Laplace form of equation (3.65) is

$$sm_h(s) = \rho A v(s), \frac{m_h(s)}{v(s)} = \frac{\rho A}{s} \tag{3.66}$$

Taking v(s) variation is (0.4, 0.4:0) m\s and $\rho = 1.02kg\backslash m^3, A = 0.4m^2$

$$\rho A = \frac{0.408kg}{m}, \frac{m_h(s)}{v(s)} = \frac{0.408\text{kg/m}}{s}$$

Using this transfer function, it is possible to know the effect of disturbance on the output. Using Simulink tool box, the graph of the change in mass of the herbicide tank is shown below.

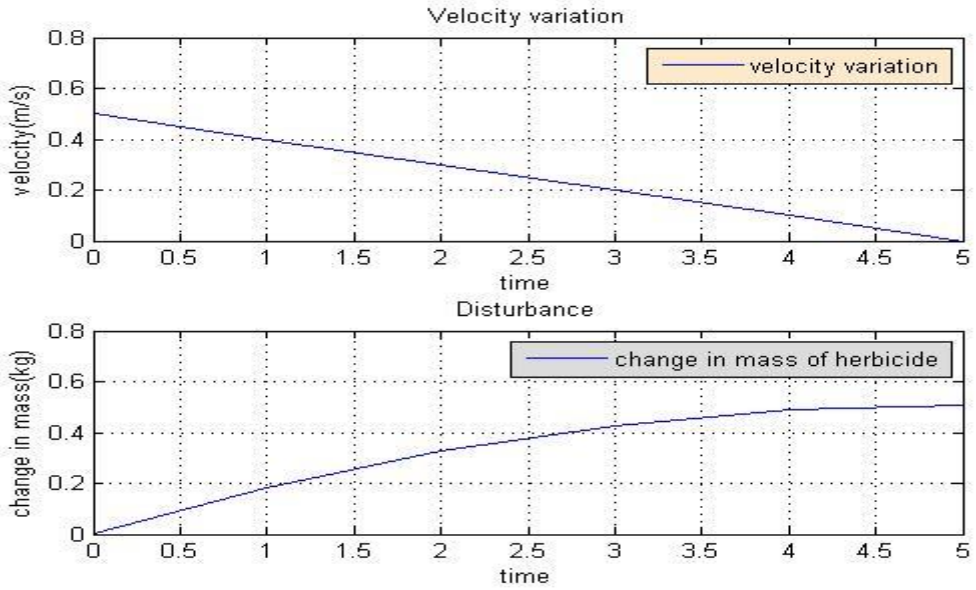


Figure 3.1.14: Graph of the change in mass of the herbicide tank

To involve the disturbance in the dynamic equation; first, knowing that for which state is exactly it is affecting is important. Now call the equation bellow.

$$\begin{bmatrix} d\omega_L/dt \\ d\omega_R/dt \end{bmatrix} = \begin{bmatrix} k_b \dfrac{d_r + b_r L_L}{b_l d_r + b_r d_l} & k_b \dfrac{d_r - b_r L_R}{b_l d_r + b_r d_l} & \dfrac{d_r a_l + b_r c_l}{b_l d_r + b_r d_l} & \dfrac{-d_r a_r - b_r c_r}{b_l d_r + b_r * d_l} \\ k_b \dfrac{d_l - b_l L_L}{b_l d_r + b_r d_l} & \dfrac{d_l + b_l L_R}{b_l d_r + b_r d_l} & \dfrac{-d_l a_l - b_l c_l}{b_l d_r + b_r \mathrm{dl}} & \dfrac{-d_l a_r + b_l c_r}{b_l d_r + b_r d_l} \end{bmatrix} \begin{bmatrix} i_L \\ i_R \\ \omega_L \\ \omega_R \end{bmatrix}$$

Where, $b_l = J_m + \mathrm{m} * L_R * \frac{r_G{}^2}{L_L + L_R}$, $b_r = J_m + \mathrm{m} * L_L * \frac{r_G{}^2}{L_L + L_R}$. The effect of the mass is clearly shown in the parameter b_l and b_r .the input disturbance modeled above affects for the state variables left wheel angular speed (ω_L) and right wheel angular speed (ω_R).

$b_l = J_m + (m_h + m_b) * L_R * \frac{r_G{}^2}{L_L + L_R}$, $b_r = J_m + (m_h + m_b) * L_L * \frac{r_G{}^2}{L_L + L_R}$

m_h is mass of the herbicide in the tank. It varies as the robot spray the herbicide to the crop.

$b_l = J_m + \left(m_h * L_R * \frac{r_G{}^2}{L_L + L_R}\right) + m_b * \mathrm{Lr} * \frac{\mathrm{rG}^2}{\mathrm{Ll+Lr}}$, $b_r = J_m + m_h * L_L * \frac{r_G{}^2}{L_L + L_R} + m_b * L_L * \frac{r_G{}^2}{L_L + L_R}$

Since m_b is mass of the robot, it is constants. $m_h * L_L * \frac{r_G{}^2}{L_L + L_R}$ and $m_h * L_R * \frac{r_G{}^2}{L_L + L_R}$ are considered as disturbance.

Disturbance , $\begin{bmatrix} dl \\ dr \end{bmatrix} = \begin{bmatrix} L_L * \frac{r_G{}^2}{L_L + L_R} \\ L_R * \frac{r_G{}^2}{L_L + L_R} \end{bmatrix} m_h = Bd\omega(k$, where $mh = \omega(k)$ is input disturbance.

$$Bd = \begin{bmatrix} 0 \\ 0 \\ L_L * \dfrac{r_G{}^2}{L_L + L_R} \\ L_R * \dfrac{r_G{}^2}{L_L + L_R} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The total model of the differential drive wheel robot becomes as fellow.

$$xm(k+1) = Amxm(k) + Bmu(k) + Bd\omega(k)$$

$$y(k) = Cmxm(k)$$

Where

$$\text{Am} = \begin{bmatrix} \frac{-(R_a + Rz)}{L_a} & -\frac{Rz}{L_a} & -\frac{k_b}{L_a} & 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{Rz}{L_a} & \frac{-(R_a + Rz)}{L_a} & 0 & -\frac{k_b}{L_a} & 0 & 0 & 0 & 0 & 0 \\ k_b \frac{d_r + b_r L_L}{b_l d_r + b_r d_l} & k_b \frac{d_r - b_r L_R}{b_l d_r + b_r d_l} & \frac{d_r a_l + b_r c_l}{b_l d_r + b_r d_l} & \frac{-d_r a_r - b_r c_r}{b_l d_r + b_r * d_l} & 0 & 0 & 0 & 0 & 0 \\ k_b \frac{d_l - b_l L_L}{b_l d_r + b_r d_l} & \frac{d_l + b_l L_R}{b_l d_r + b_r d_l} & \frac{-d_l a_l - b_l c_l}{b_l d_r + b_r \text{dl}} & \frac{-d_l a_r + b_l c_r}{b_l d_r + b_r d_l} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & r_G \frac{L_R}{(L_L + L_R)} & r_G \frac{L_L}{L_L + L_R} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{r_G}{L_L + L_R} & \frac{r_G}{L_L + L_R} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & cos(ar * T) & 0 & 1 & 0 & -vr * sin(ar * T) \\ 0 & 0 & 0 & 0 & sin(ar * T) & 0 & 0 & 1 & cos(ar * T) \\ 0 & 0 & 0 & 0 & 0 & 0.1 & 0 & 0 & 1 \end{bmatrix}$$

$$Bm = \begin{bmatrix} v_a/L_a & 0 \\ 0 & v_a/L_a \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, Bd = \begin{bmatrix} 0 \\ 0 \\ \text{Ll} * \frac{\text{rG}^2}{\text{Ll+Lr}} \\ \text{Lr} * \frac{\text{rG}^2}{\text{Ll+Lr}} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \text{Cm} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, Dm = [0]$$

Inserting parameter values of the robot we have the following state space of the plant with its disturbance.

$$Am = \begin{bmatrix} -176 & -16 & -0.0028 & 0 & 0 & 0 & 0 & 0 & 0 \\ -16 & -176 & 0 & -0.0028 & 0 & 0 & 0 & 0 & 0 \\ 0.3312 & 0.3312 & 0.3196 & 0.3428 & 0 & 0 & 0 & 0 & 0 \\ 0.3312 & 0.3312 & 0.3428 & 0.3196 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.1 & 2.1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -3.7 & 3.7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0.8 \\ 0 & 0 & 0 & 0 & 0 & 0.1 & 0 & 0 & 1 \end{bmatrix}$$

$$Bm = \begin{bmatrix} 960 & 0 \\ 0 & 960 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, Cm = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, Bd = \begin{bmatrix} 0 \\ 0 \\ 1.2800 \\ 1.2800 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

3.6. Reference generator model

The model of the plant is derived as an error model. The states are the deviations between actual robot and reference robot. A reference robot can be considered as a robot with reference (desired) values of the states are steady state of the dynamic equations.

Once the camera detects the position (x_r, y_r) of the crop row, the linear velocity, orientation angle and angular velocity of the reference robot can be derived based on equation (3.44,3.45,3.46). The other reference variables i_{Lr}, i_{Rr}, ω_{Lr}, ω_{Rr}, v_{Br} and ω_{Br} are obtained from calculation of steady-state values for constant engine power voltages .Steady-state in matrix representation is shown below.

$$\begin{bmatrix} U_L \\ U_R \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R_a + R_Z & R_Z & k_b & 0 & 0 & 0 & 0 & 0 \\ R_Z & R_a + R_Z & 0 & k_b & 0 & 0 & 0 & 0 \\ k_b & 0 & -K_r & 0 & -1 & 0 & 0 & 0 \\ 0 & k_b & 0 & -K_r & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -r_G k_v & 0 \\ 0 & 0 & 0 & 0 & 0 & -L_L L_R & 0 & -K_\omega \\ 0 & 0 & L_R & L_L & 0 & 0 & -L_R + L_L/r_G & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & -L_R + L_L/r_G \end{bmatrix} \begin{bmatrix} i_{Lr} \\ i_{Rr} \\ \omega_{Lr} \\ \omega_{Rr} \\ M_L \\ M_R \\ v_{Br} \\ \omega_{Br} \end{bmatrix} \quad (3.67)$$

Inserting the values of parameter of the robot the steady state equation is shown below.

$$\begin{bmatrix} 24u_{Lr} \\ 24u_{Rr} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 4.4000 & 0.4000 & 0.0001 & 0 & 0 & 0 & 0 & 0 \\ 0.4000 & 4.4000 & 0 & 0.0001 & 0 & 0 & 0 & 0 \\ 0.0001 & 0 & -0.0001 & 0 & -1.0000 & 0 & 0 & 0 \\ 0 & 0.0001 & 0 & -0.0001 & 0 & -1.0000 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0000 & 1.0000 & -0.0008 & 0 \\ 0 & 0 & 0 & 0 & 0 & .5000 & 0 & -0.0560 \\ 0 & 0 & 0.5000 & 0.5000 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.0000 & -1.0000 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} i_{Lr} \\ i_{Rr} \\ \omega_{Lr} \\ \omega_{Rr} \\ M_L \\ M_R \\ v_{Br} \\ \omega_{Br} \end{bmatrix}$$

After the desired position $(x_r, y_r$), orientation (α_r), $\omega_r and\ v_r$ are obtained .The rest states and desired control signals (u_L, u_R) are generated from the equation (3.67) as fellow.

$$\omega_{Rr} = \frac{v_{Br} + L_R * \omega_{Br}}{r_G}$$

$$\omega_{Lr} = -\frac{\omega_{Br} * (L_L + L_R) - \omega_{Rr} * r_G}{r_G}$$

$$M_R = \frac{L_L * r_G * k_v * v_{Br} + -K_\omega * \omega_{Br}}{L_L + L_R}$$

$$M_L = r_G * k_v * v_{Br} - M_R$$

$$i_{Lr} = \frac{M_L + K_r * \omega_{Lr}}{K} * 0.001$$

$$i_{Rr} = \frac{M_R + K_r * \omega_{Rr}}{K} * 0.001$$

$$u_{Lr} = \frac{(R + R_Z) * i_{Lr} + Rz * i_{rr} + K * \omega_{Lr}}{u_0}$$

$$u_{Rr} = \frac{(R + R_Z) * i_{Rr} + Rz * i_{Lr} + K * \omega_{Rr}}{u_0}$$

The reference signals are illustrated in chapter five.

3.7. Controller design

One who want to design a controller for a given system, knowledge of the actuating devices, plant, controlling device, output variables (actual values) and desired values is very important. This chapter deals with the MPC design for the differential drive wheeled agricultural robot. Computing a trajectory of future control inputs that optimizes the future behavior of plant output, where the optimization is carried out within a limited time window is objective of a model predictive control strategy. Controllability is needed in order for the design of MPC to access all dynamic modes in the process. A given model that describes the dynamics of the system is paramount in predictive control. A good dynamic model gives a consistent and accurate prediction of the future. Mathematical model of the plant is very important for the design of model predictive control systems. The model used in the control system design is taken to be a state-space model. The current information required for predicting ahead is represented by the state variable at the current time. Already I have formulated the state space model of the robot in chapter three. Since MPC needs augmented and controllable model the new augmented state space model formulated and realized to obtain controllable model. Finally, the augmented model is used to design mpc for the purpose of controlling the robot trajectories ($y_{B,}\ x_B\ , \alpha_B$).

3.7.1. Requirements to design MPC for DDWAR

- Augmented model of the robot
- Prediction horizon
- Control horizon
- Objective criterion (objective function)
- Constraint (input and output constraint)
- Weight on input and output variables

3.7.2. Augmented state space representation of DDWAR

The augmented model is an error based model where the state variables are deviations from reference variables. The reference variables can be seen as an ideal agricultural robot following a time-varying reference trajectory. These reference velocities v_r , ω_r and orientation angle , α_r can be calculated from Equations (3.44, 3.45 and 3.46) with the reference inputs (positional coordinates of the robot (x_r, y_r). The other reference variables i_{Lr} , i_{Rr} , ω_{Lr}, ω_{Rr} and (u_{Lr}, u_{Rr}) can be pre-calculated from the model, Equation (3.67). The MPC algorithm calculates the optimal control inputs (motor voltage control inputs u_L and u_R at every time instance. The following diagram shows the overall control scheme.

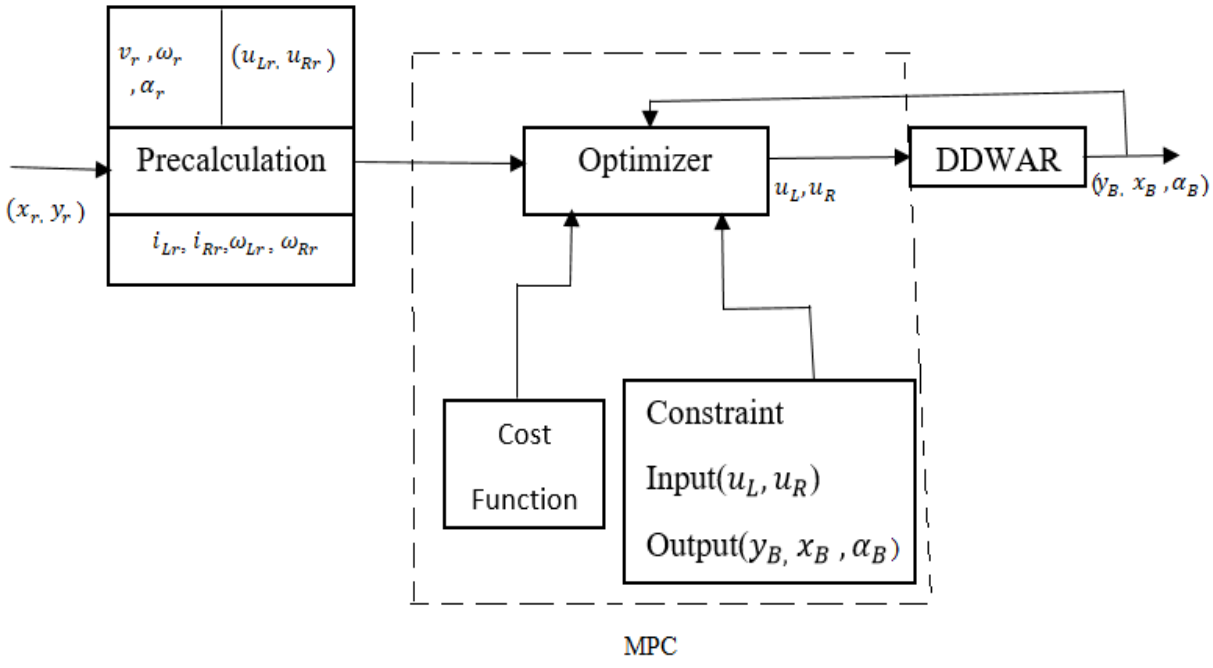


Figure 3.15: Overall control system block diagram

From the study of feedback control that in order to remove steady-state error from a control system consisting integral action is required. For model predictive controllers, a better way to have integral action is modifying the state-space model. By removing the disturbance in the state-space model in equation (3.61) we have.

$$\begin{vmatrix} \Delta xm(k+1) \\ Y(K+1) \end{vmatrix} = \begin{bmatrix} Am & 0m^T \\ CmAm & I_{q\times q} \end{bmatrix} \begin{bmatrix} \Delta xm(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} Bm \\ CmBm \end{bmatrix} \Delta u(k)$$

$$y(k) = [0m \quad I_{q\times q} \begin{bmatrix} \Delta xm(k) \\ y(k) \end{bmatrix}$$

The matrix Am, Bm, Cm are shown below .The state space is the augmented model of the agricultural robot written in equation (3.54) and (3.55) with the parameter values shown in table one and two.

$$Am = \begin{bmatrix} 3.187e-08 & -8.067e-08 & -1.65e-05 & 9.668e-07 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -8.067e-08 & 3.187e-08 & 9.668e-07 & -1.65e-05 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.001837 & 0.001837 & 1.033 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.001837 & 0.001837 & 0.0354 & 1.033 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.0007089 & 0.0007089 & 0.2171 & 0.2171 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1.959e-08 & 1.959e-08 & -0.3696 & 0.3696 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3.443e-05 & 3.443e-05 & 0.0111 & 0.0111 & 0.1052 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2.388e-11 & 2.388e-11 & -0.0005069 & 0.0005069 & 0 & 0.00415 & 0 & 1.105 & 0.08455 & 0 & 0 & 0 \\ -9.258e-10 & 9.258e-10 & -0.01855 & 0.01855 & 0 & 0.1005 & 0 & 0 & 1.01 & 0 & 0 & 0 \\ 3.553e-05 & 0.01147 & 0.01147 & 0.1105 & 0 & 0 & 1.105 & 0 & 0 & 1.105 & 0 & 0 \\ -2.446e-11 & 2.446e-11 & -0.0005197 & 0.0005197 & 0 & 0.004291 & 0 & 0.08884 & 0.08884 & 0 & 1.105 & 0 \\ -9.556e-10 & 9.556e-10 & -0.01918 & 0 & 0.1057 & 0 & 0 & 0.01057 & 0 & 0 & 0 & 1.105 \end{bmatrix}$$

$$Bm = \begin{bmatrix} 5.5 & -0.5 \\ -0.5 & 5.5 \\ 0.162 & 00.162 \\ 0.162 & 00.162 \\ 0.03201 & 0.03201 \\ -8.859e-07 & 8.859e-07 \\ 0.001036 & 0.001036 \\ -5.409e-10 & 5.409e-10 \\ -2.799e-08 & 2.799e-08 \\ 0.001062 & 0.001062 \\ -5.513e-10 & 5.513e-10 \\ -2.866e-08 & 2.866e-08 \end{bmatrix} \text{cm} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

This augmented state space is not controllable. The model used for design model predictive must be controllable in order to take the states everywhere we want. A minimal realization is used over the above state space.

Then the controllable model of the robot used in the design of MPC is shown below.

$$A = \begin{bmatrix} 1.089 & -0.1013 & 0.02791 & 0.05923 & 0.05626 & 0.0007039 & 0.07298 & 0.03133 & -0.0706 \\ -0.04817 & 0.7995 & 0.5446 & -0.126 & 0.01563 & 0.00101 & 0.0225 & 0.06282 & -0.02648 \\ 0.05557 & 0.5453 & 0.3864 & 0.004829 & 0.006014 & .00067 & 0.01522 & 0.04883 & -0.02886 \\ -0.08706 & -0.07249 & 0.03041 & 1.117 & -0.03546 & -0.001191 & -0.004215 & 0.05406 & 0.01377 \\ -0.1036 & -0.08606 & -0.06589 & 0.1229 & 0.9589 & 0.0001838 & -0.1202 & 0.0008081 & 0.09308 \\ -1.351e-05 & -1.123e-05 & -8.555e-06 & 1.329e-05 & -6.737e-05 & 3.068e-08 & 3.132e-05 & 0.0002882 & -0.001861 \\ -0.1329 & -0.1104 & -0.08425 & 0.133 & -0.1118 & -0.0001219 & 1.008 & -0.1613 & 0.1342 \\ -0.06143 & -0.05105 & -0.03894 & 0.06147 & -0.0517 & -5.637e-05 & -0.05353 & 1.042 & 0.3327 \\ -0.001748 & -0.001452 & -0.001108 & 0.001749 & -0.001471 & -1.603e-06 & -0.0009068 & -0.002592 & 1.096 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.09161 & -0.3304 \\ 0.3522 & -3.055 \\ -0.3428 & 4.538 \\ -0.06609 & -0.4587 \\ 0.04152 & 0.08156 \\ 5.501 & -0.5184 \\ 0.01446 & 0.0516 \\ 0.00264 & 0.01981 \\ 0.18 & 0.1805 \end{bmatrix}$$

$$C = \begin{bmatrix} -0.3649 & 0.4488 & 0.3154 & 0.4372 & 0.4999 & -4.647e-06 & -0.3507 & 0.04198 & -0.01312 \\ 0.05084 & -0.4084 & -0.3101 & -0.366 & 0.5026 & 1.314e-05 & -0.5838 & 0.0813 & -0.02233 \\ 0.6207 & -0.05106 & 0.0523 & 0.4127 & -0.07269 & -3.27e-05 & -0.2175 & 0.3185 & 0.06575 \end{bmatrix}$$

3.7.3. Prediction and control horizon

Choosing the prediction horizon for the differential drive wheel agricultural robot needs careful consideration without using exponential data weighting. The design parameters are $N = 5$, $p = 26$. Where, N is control horizon and p is prediction horizon. Because of the embedded integrators, the prediction horizon p as a design parameter plays a useful role in a predictive control system. If it is chosen too short, the closed-loop system is not necessarily stable, and if it is too large, the predictive control system will encounter a numerical stability problem the plant has two inputs, three outputs and nine states. The number of outputs is greater than the number of inputs. If the number of outputs is greater than the number of inputs, it is not possible to control each of the measured outputs independently with zero steady-state errors. The maximum total delay in the prediction model is zero full sampling periods. There are 22 steps between the last MV movement and the end of the prediction horizon.

3.7.4. Penalty Weights On Manipulated Variable Rates

Increasing penalties on manipulated variable is guaranteed to have a positive-definite Hessian by changes (MV rates). A potential disadvantage is a more sluggish controller response in comparison to the original controller. The following table lists the controller's minimum MPC object, Weight, manipulated variable rate parameter for each manipulated variable (u_L and u_R).

Table 3 Manipulated variable

MV (manipulated variable)	**Weights.MVRate**
Left motor control signal(u_L)	0.1
right motor control signal(u_R)	0.1

3.7.5. Penalty Weights On Output Variables

The output variables($x_{B,}\ y_B, \alpha_B$) penalty weights also affect the Hessian. Non-zero values emphasize the importance of output variables target tracking, making a unique QP (quadratic programming) solution more likely. The following table lists the minimum weight for each output variables along the prediction horizon.

Table 4: Output variables

OV (output variables**)**	**Weights.OV**
x_B position	3000
y_B position	3000
α_B heading angle	3000

3.7.6. Cost function

The model predictive controller produces an optimal control sequence by optimizing a quadratic cost function at each sampling time. The system receives the first control action. At the next sampling time, the optimization problem is solved again using the updated process measurements and a shifted horizon. The cost function formulation depends on the control requirements. Mostly the common cost function is written in the form of

$$J = \sum_{i=1}^{ny} \sum_{j=1}^{n2} ri[yi(k+j) - wi(k+j)]^2 + \sum_{i=1}^{nu} \sum_{j=0}^{n3} qi[\Delta ui(k+j-1)]^2 \qquad (3.68)$$

Where $yi\ (k+j)$ is an optimum *j*-step ahead prediction of the system *i*-th output, n*2* is the control error horizon n*3* is the control horizon and $Wi\ (k+j)$ is the future reference or set point for the *i*-th controlled variable. The parameters *ri* and *qi* are the weighting coefficient for control errors and

control increments respectively and $\triangle ui\ (k+j-1)$ is the control increment of the *i*-th input. *nu* and *ny* are the number of inputs and number of outputs (manipulated and controlled variables).

The cost function in Equation (3.68) can be represented in matrix form as

$$J = (Y - W)^T R\ (Y - W + U^T\ \mathrm{QU}) \quad (3.69)$$

Where, R (output weight) and Q (input weight) are diagonal matrices with diagonal elements *ri* and *qi* respectively. W is a column vector of N future set points and Y is the outputs of the plant The differential derive agricultural robot has three out puts and two inputs. The input and output diagonal weights are$Q = diag\ (1,1)\ and\ R = diag\ (1,1,1)$.

3.7.7. Constraints

In a long range predictive control, the controller has to anticipate constraint violation and correct control actions in a correct way. The input constraints and output constraints are:

$$Umin \leq U(i) \leq Umax, i \in \{K, K+N-1\} \quad (3.70)$$

$$Ymin \leq Y(i) \leq Ymax, i \in \{K+1, K+N\} \quad (3.71)$$

Where:

$Umin$ is input control or manipulated variable lower limit.

Umax is input control or manipulated variable upper limit.

Ymin is process output lower limit.

$Ymax$ is process output upper limit.

The constraints or limits are shown in table 5 and 6 bellow.

Table 5: Output constraint

Output constraint	**Values**
Lower limit:α_B heading	0 deg
Upper limit: α_B heading	180 deg
Lower limit: x_B position	-2m
Upper limit: x_B position	40m
Lower limit: y_B position	-2m

Upper limit: y_B position	4m

Table 6: Input constraint

Input constraint	**Values**
Lower limit: left control signal (u_L)	-2v
Upper limit: left control signal (u_L)	2v
Lower limit: right control signal (u_R	-2v
Upper limit: right control signal (u_R)	2v

CHAPTER FOUR

4. SIMULATION AND RESULTS DISCUSSION

4.1. Behavior of the agricultural Robot without controller

Before designing a controller for the robot knowing its behavior is very important. The robot is represented by state space model with nine states ($x_B, y_B, \alpha, i_L, i_R, \omega_L, \omega_R, v_B, \omega_B$).

Basic verification of the above derived model was made by calculation for situations where we can guess the behavior of the real device. First value of the state variables in steady states is given for some combinations of parameters and motor supply voltages.

To know the robot character a unit step input voltage is used and the graphs are obtained using MATLAB. Dynamic behavior is demonstrated on the time courses of currents and angular speeds of the motors starting from zero initial conditions.

Figure 4.1 shows the model of the robot with step inputs (u_L, u_R) and output variables (x_B, y_B, α). As the model indicates the robot position and orientation is controlled by varying the input voltage control signals.

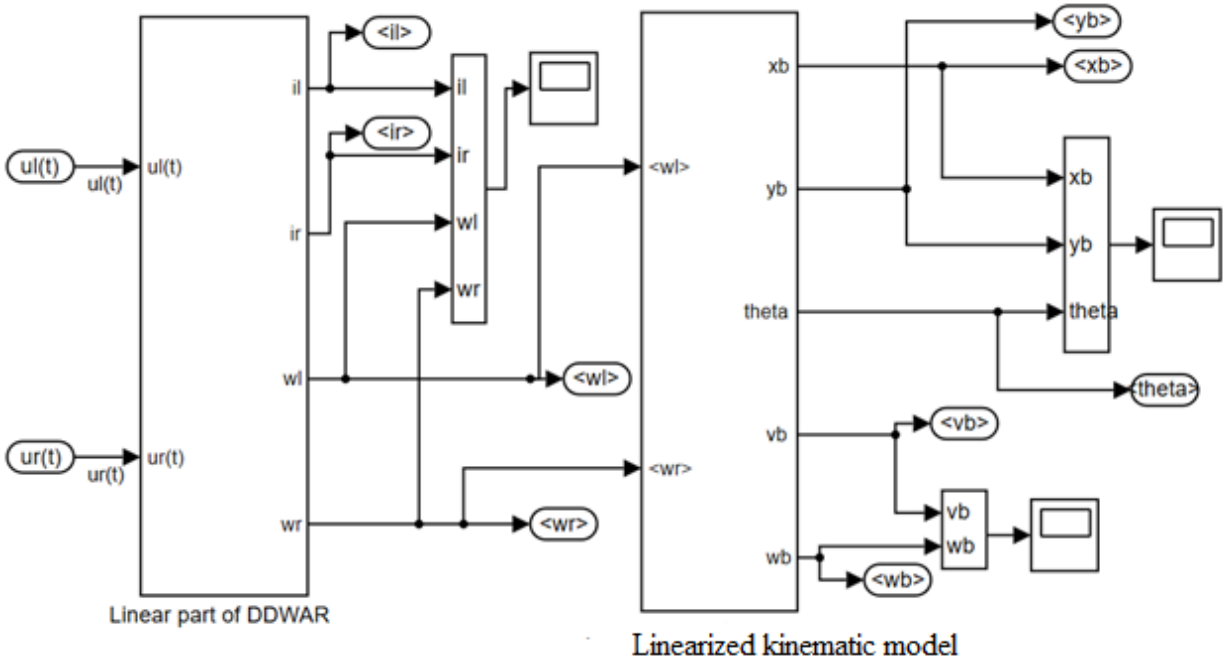


Figure 4.1: Model of differential drive wheeld agricultural robot

The model of the robot involves two DC series motor dynamics, chassis dynamics and kinematics as shown in Fig 4.1(a) and 4.1(b). Fig 4.1 (a) is the dynamic of the robot and Fig 4.1(b) is the kinematics of the robot. The output $(v_{B,}\omega_B)$ of the dynamic model is an input to the kinematic model of the robot.

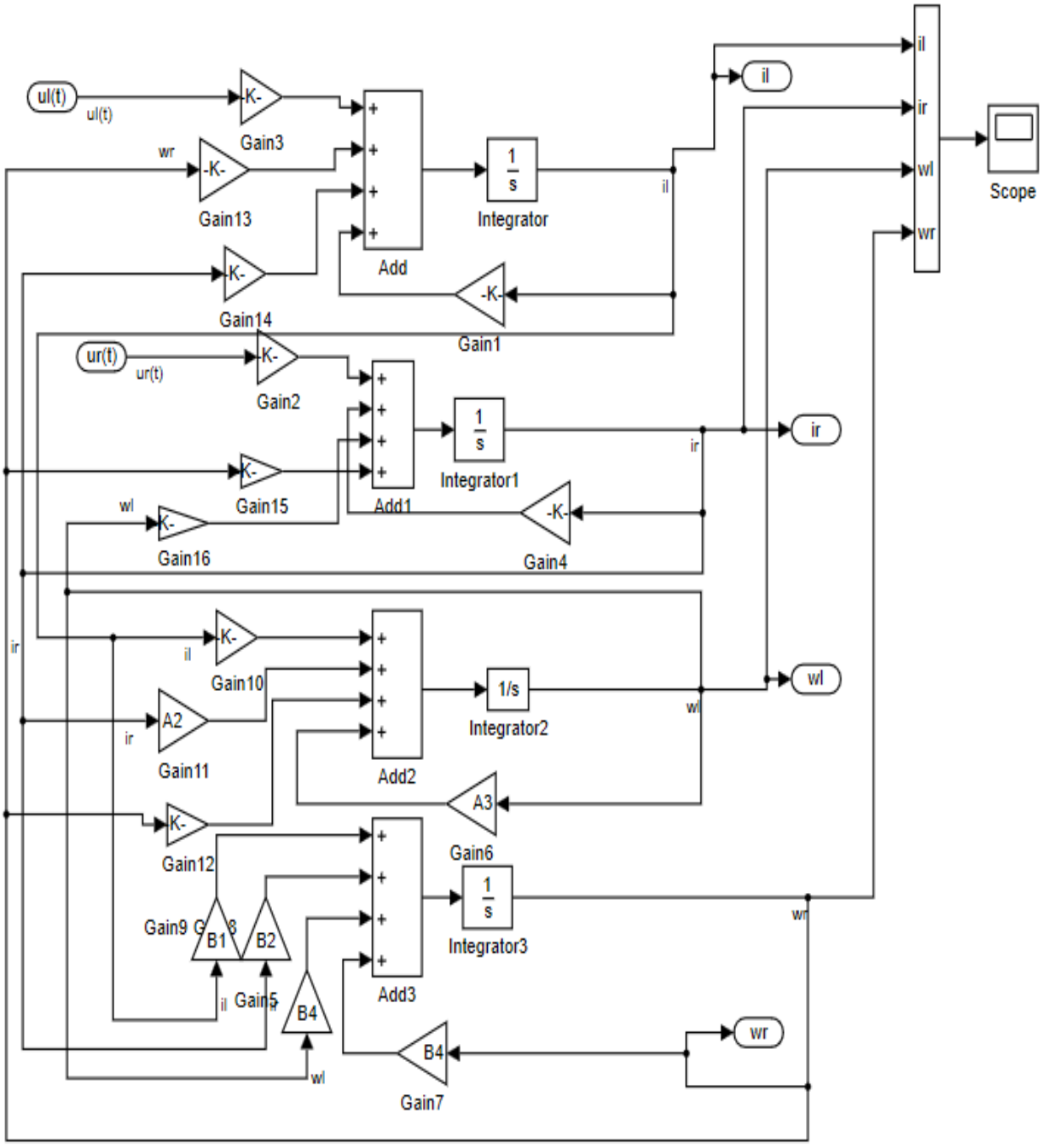


Figure 4.1 (a): Dynamic model of differential drive wheeld agricultural robot

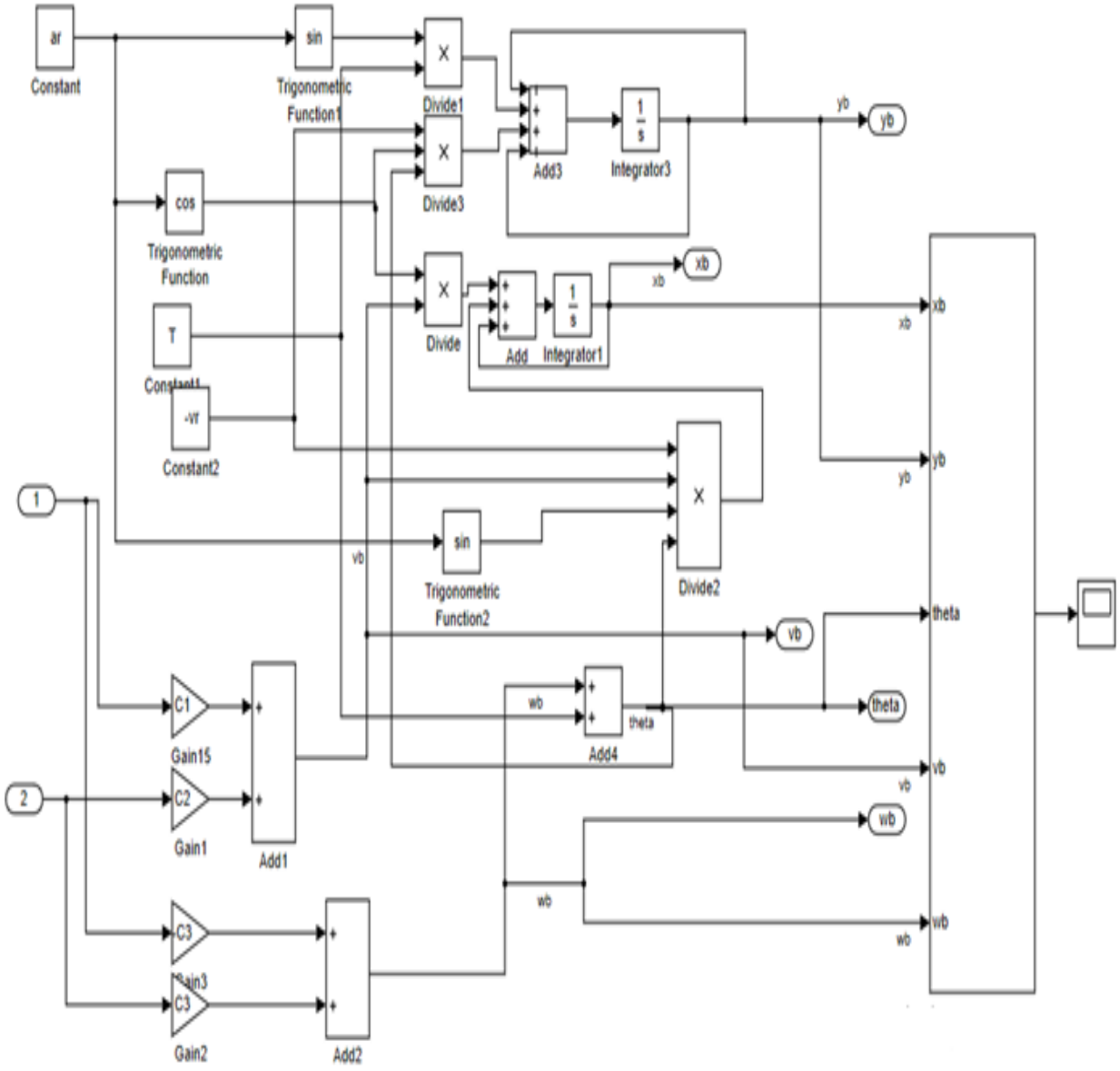


Figure 4.1(b): kinematic model of differential drive wheeld agricultural robot

To test the open loop system of the robot a unit step input signal is used as shown in fig 4.2 bellow.As the graph of wheeel speed virsuss time and speed at chasis point shown in fig 4.5 the robot is at rest intially and increases the magnitudee of the angular speed and forward speed with time.The current that makeks the robot are bigger than the the individual motor currents due to the effect of robot body on the model.

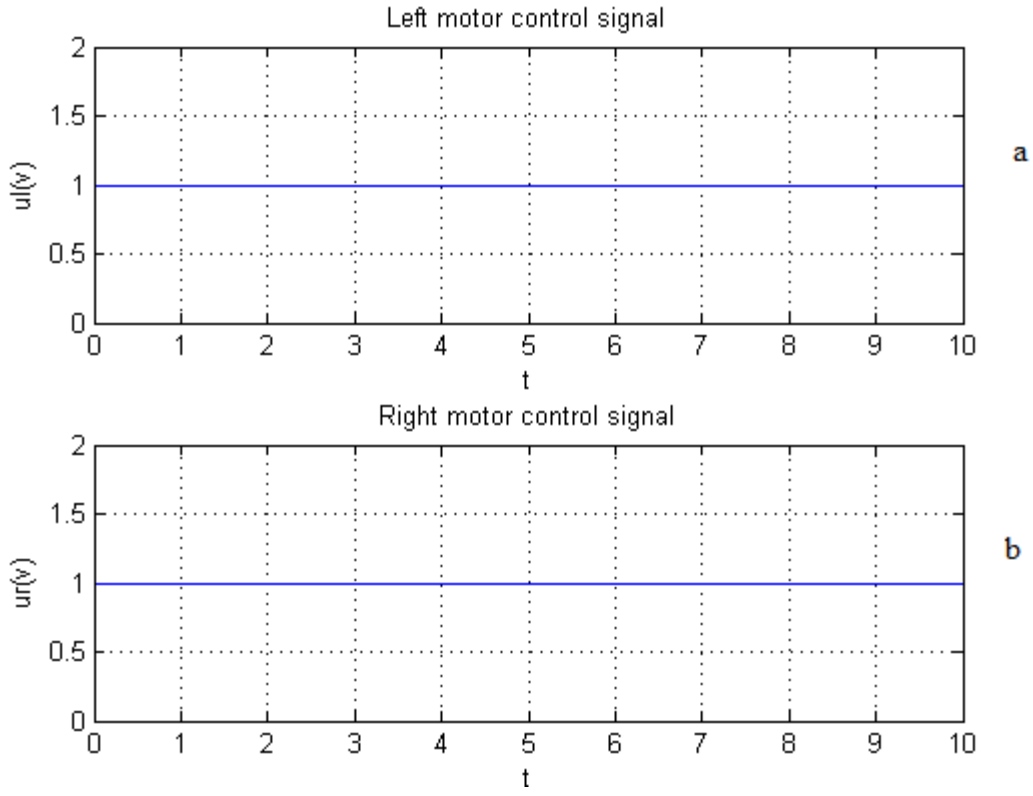


Figure 4.2: Test input signals of the of differential drive wheel agricultural robot

Fig 4.2 (a) is the control signal applied for the DC series motor installed in the right wheel of the differential drive wheeld agricultural robot. Fig 4.2 (b) shows left motor control signal that actuated the wheel to stir the robot as right motor does. Both signals are a unit step test inputs for a given ten seconds operating time. For the given control signals the currents flowing in to the motors of the robot is shown in fig 4.3 bellow . Once the motor actuated the left and right motor current increases posesetivelly for one second .The current kept constant untill the end of the time in seconds. The robot involves two dc series motors with common power supply.The two motors actuated differentially.As shown in figure 4.3 a and b both motors have the same current and speed for the untip step actuating signal. The maximum current and speed for both motors 0.25A and 0.17 rad/s respectivelly.Figure 4.3 a shows left motor current and speed. Figure 4.3 b shows right motor current and motor speed.

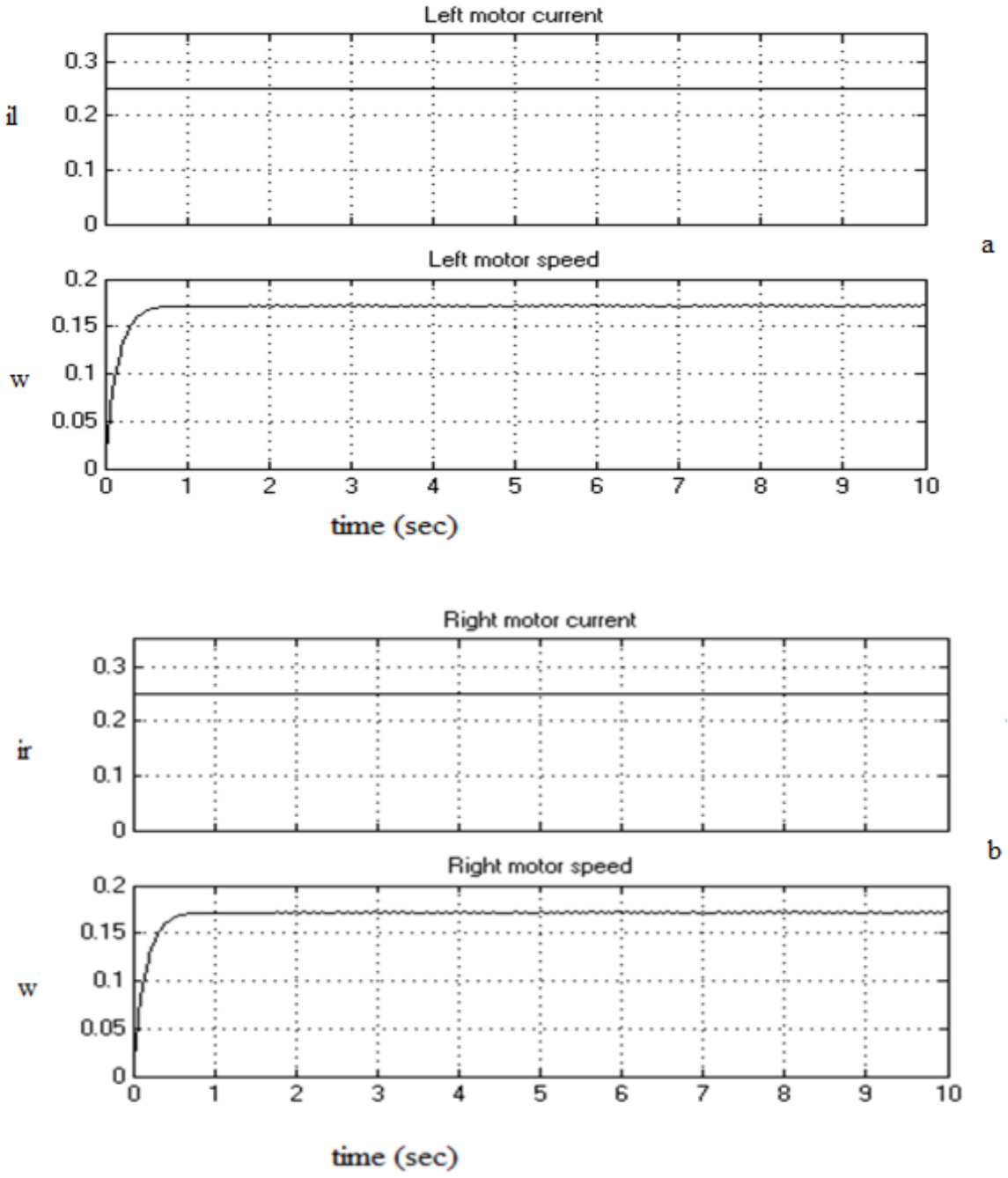


Figure 4.3: Current flow to DC series motor and its speed

Fig 4.4 below shows that as the robot DC series motor are actuaed with the same magnitude input signal the left and right wheel speeds have the same magnitude . The angular speed is increasing with time.This indicates that a controller design is neccesary to tune the angular speed to the desired value for the best operation in the irrigation.

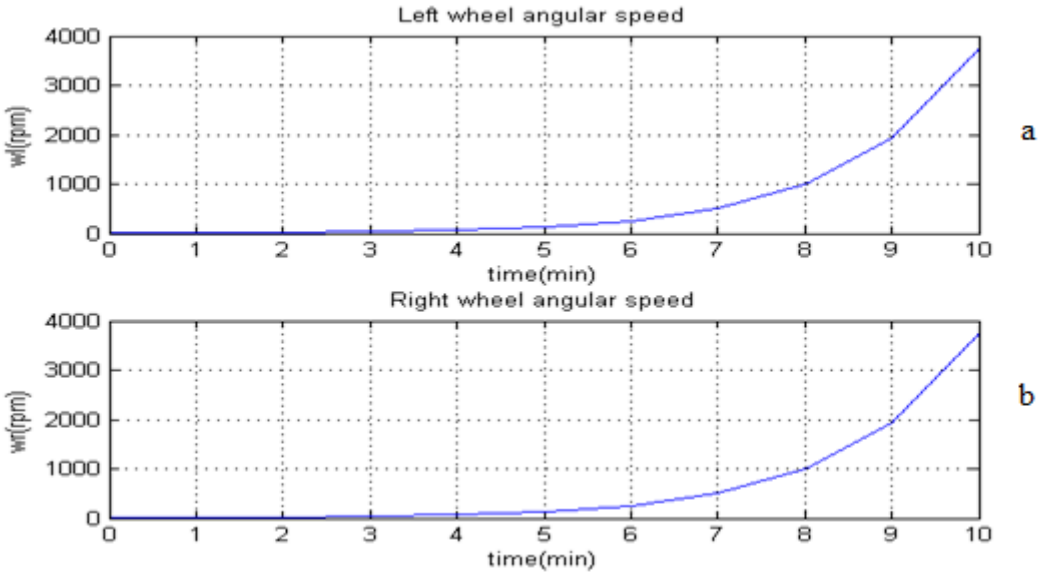


Figure 4.4: Angular speed of DC series motors connected to wheel with gear

Fig 4.5. bellow shows the robot angular speed and forward speed of the robot wich indicates its total movment .The robot is moving with different magnitude of angular and forward speeds at a different time. Figure 4.5 (a) shows the robot angular speed .It is varing with very small magnitude with time.This indicates that the robot is moving in straight path .Figure 4.5 (b) shows the robot forwad speed .It is increasing with time hence some one who wants to fix the robot to the desired value a controller must be designed.

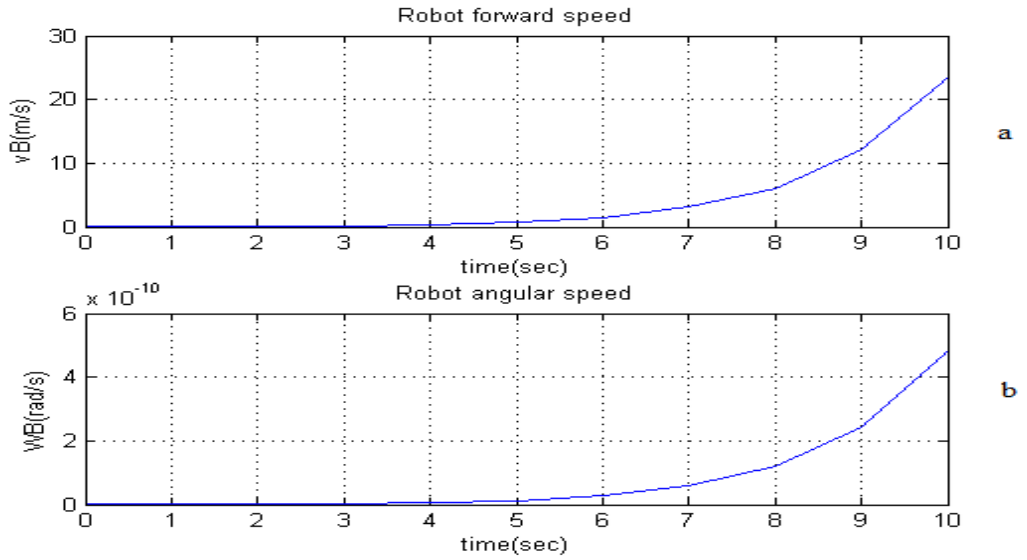


Figure 4.5: Robot angular speed and forward speed at chassis point

Figure 4.6 bellow shows the robot postion and oreaintation vs time graph. The robot starts from zero initial values of the states ($x_{B,}y_{B,}\alpha, i_{L,}i_{R,}\omega_{L,}\omega_{R,}v_{B,}\omega_{B}$).The output of the robot is the position($x_{B,}y_{B}$) and orientation or heading angle (α).The graph indicates that the robot is moving in between x and y position with small angle at chassis point B.

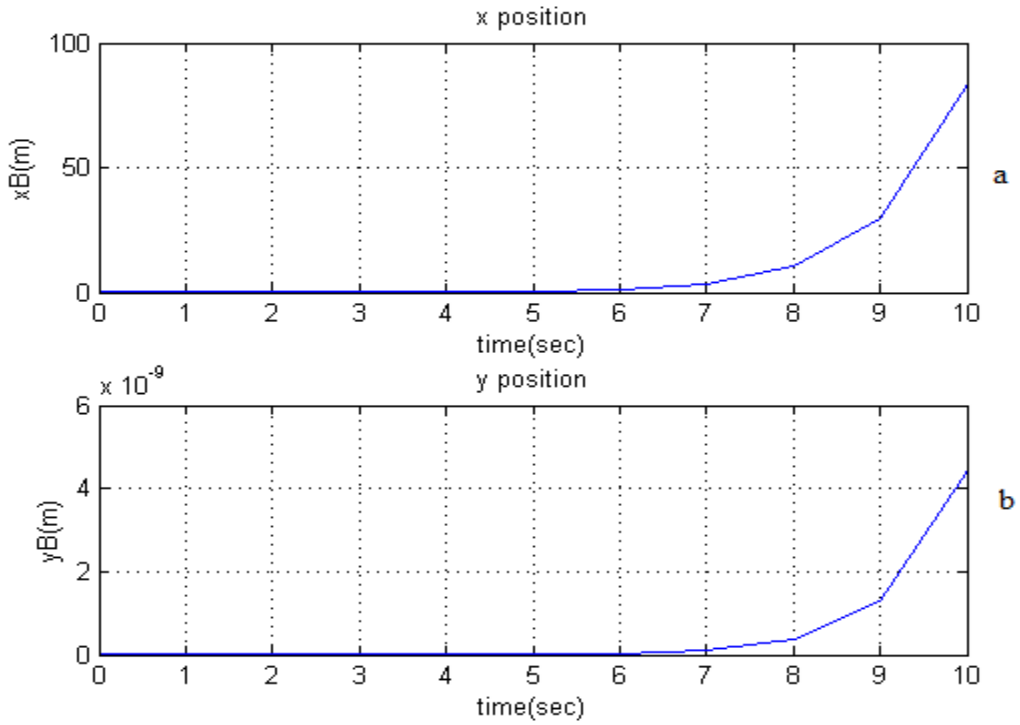


Figure 4.6: Robot position vs time graph.

The open loop simulation results the following heading angle of the robot. It shows that the robot is moving at angle of different magnitude with time. One how want to stir the robot in straight line it is necessary to make the angle zero at all time during the movement.

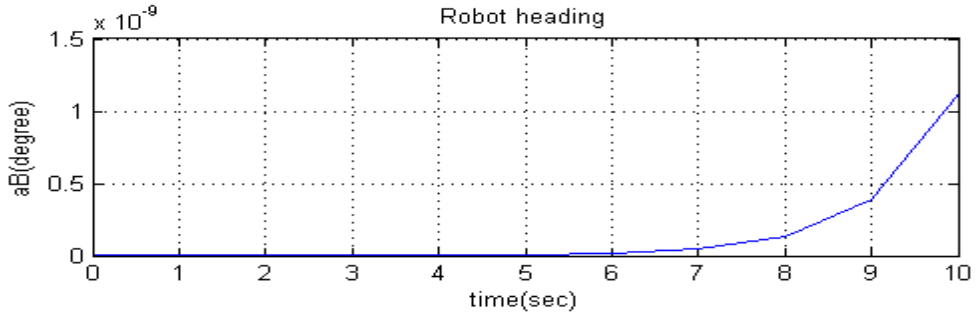


Figure 4.7: Robot angle at chassis point

The graph shown bellow in figure 4.8 is the robot trajectory with out controller .The robot is moving at angle illustrated in fig 4.7 above .As final goal of this thesis is to control the position and orientation, a good contrroller is needed to stirr the robot to the desired crop row that it receive from camera after Hough tranpsform method .

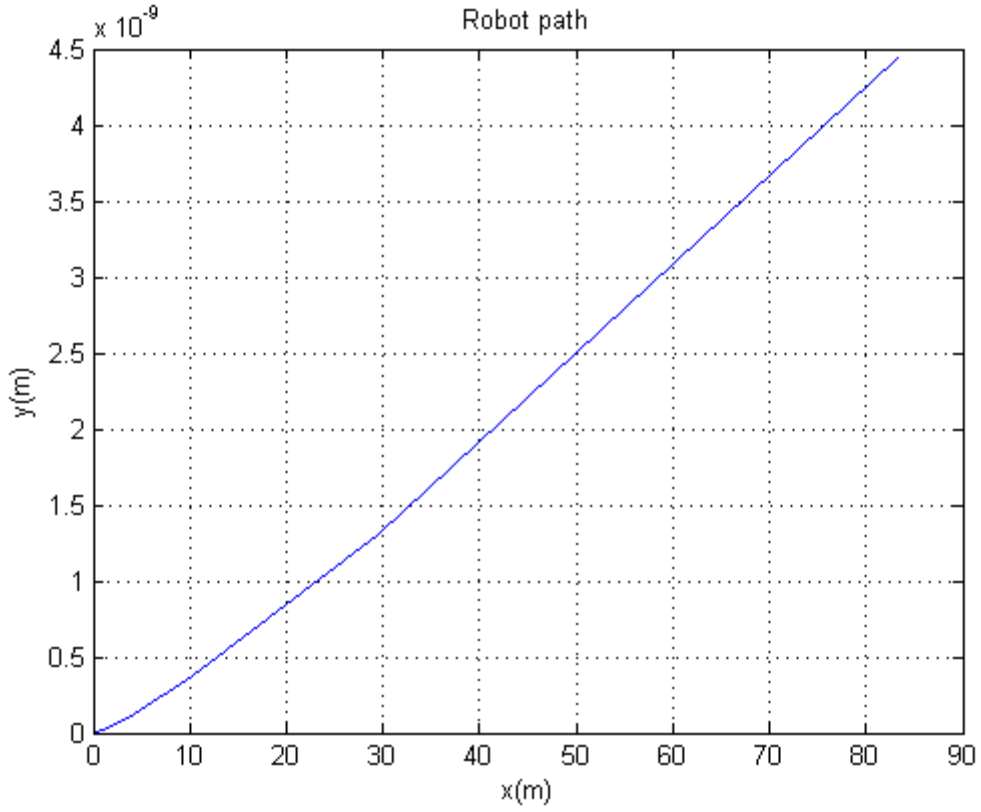


Figure 4.8: Open loop robot path for unit step input signal

The open loop trajectory indicates the robot is moving to the $x_{B,}$ direction with small distance to y_B) axsis .To the final goal of this thesis the x_B and y_B postions have to be controlled to the desired values of ($x_{B,}y_B$) postion and robot oraintation α to be navigated in the crop row.

4.2. Reference generator

As the final goal of the control design is stirring the robot without damaging the crop in the row by the wheels, the controller must know the path of the robot. The pinhole camera receives the position of the crop in pixel in the form of ($x_{B,}$ y_B) coordinate. The detection process is explained in chapter three.

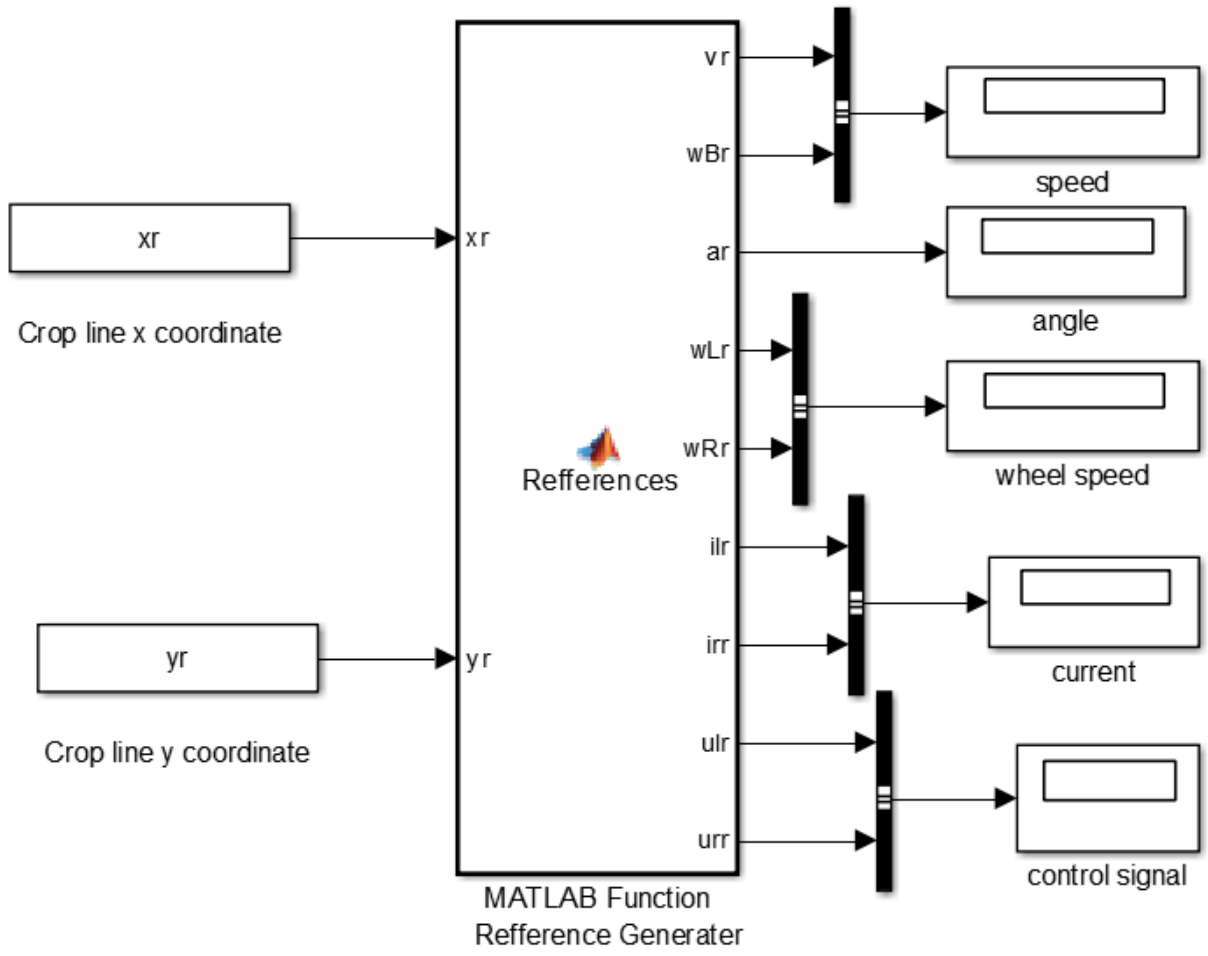


Figure 4.9: Reference generator based on crop row

Now take the camera detects the line as:

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} 4t \\ 0 \end{bmatrix}$$

$$v_{r(t)} = \sqrt{\left(\frac{\mathrm{d}x_r(\mathrm{t})}{\mathrm{dt}}\right)^2 + \left(\frac{\mathrm{d}y_r(\mathrm{t})}{\mathrm{dt}}\right)^2}, \quad v_{r(t)} = \sqrt{\left(\frac{\mathrm{d}}{\mathrm{dt}}4t\right)^2 + \left(\frac{\mathrm{d}0}{\mathrm{dt}}\right)^2}$$

$$v_{r(t)} = \sqrt{4^2} = \frac{4m}{s}, \qquad \alpha_r = arctan^2(0{,}4) = 0,$$

$$\omega_r = \frac{\mathrm{d}\alpha_r(\mathrm{t})}{\mathrm{dt}} = 0 \ \text{Rad/s.}$$

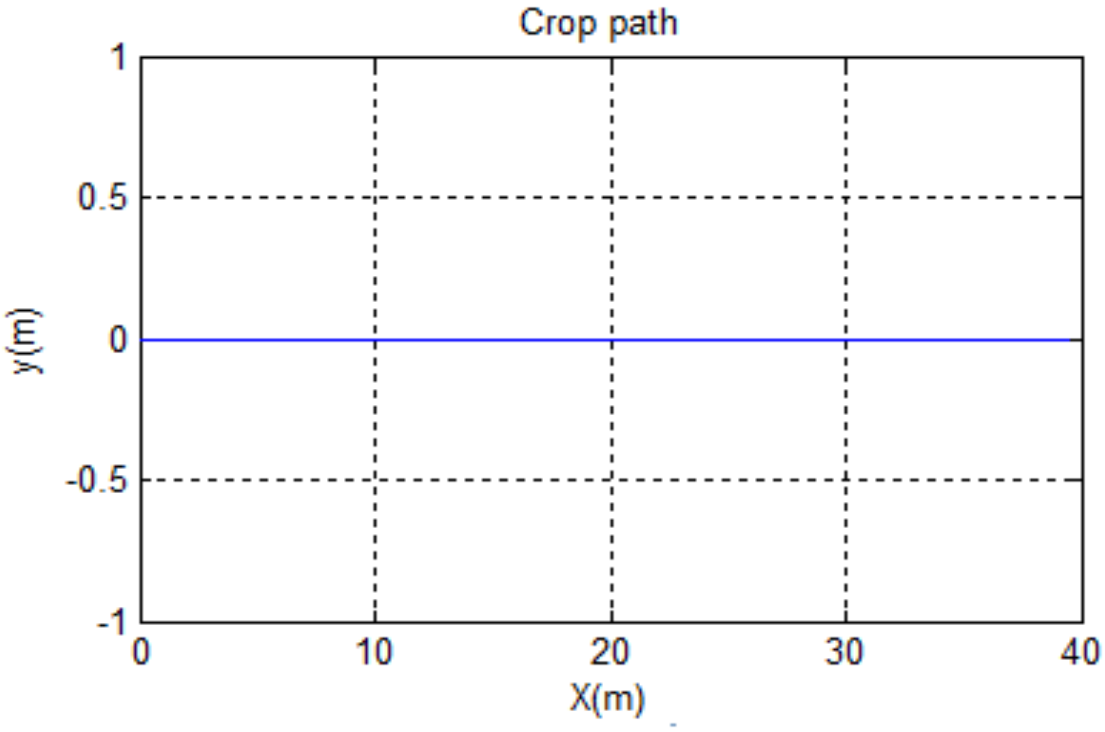


Figure 4.10: Crop row in the irrigation

This line is the desired line in which the differrntial drive wheeld agricaltural robot has to fellow . MPC navigates the robot to operate in the line.The line informations $(x_{r,}y_r)$ are obtained from the pinhol camera .

Since the plant Model used in the model predictive controller is an error model the nine states $(i_L, i_R, \omega_L, \omega_R, v_B, \omega_B, x_{B,}, y_{B,}\alpha_B)$ and two control signals or manipulated variables (u_L, u_R) needs a reference to form error states as explained mathematically in chapter three.

Fig 4.11 below is the desired motor current, the left and right DC series motors angular speed that the robot has to obey for the best trajectory tracking of crop line. Equal angular speed of the robot shows that it is moving in straight path.

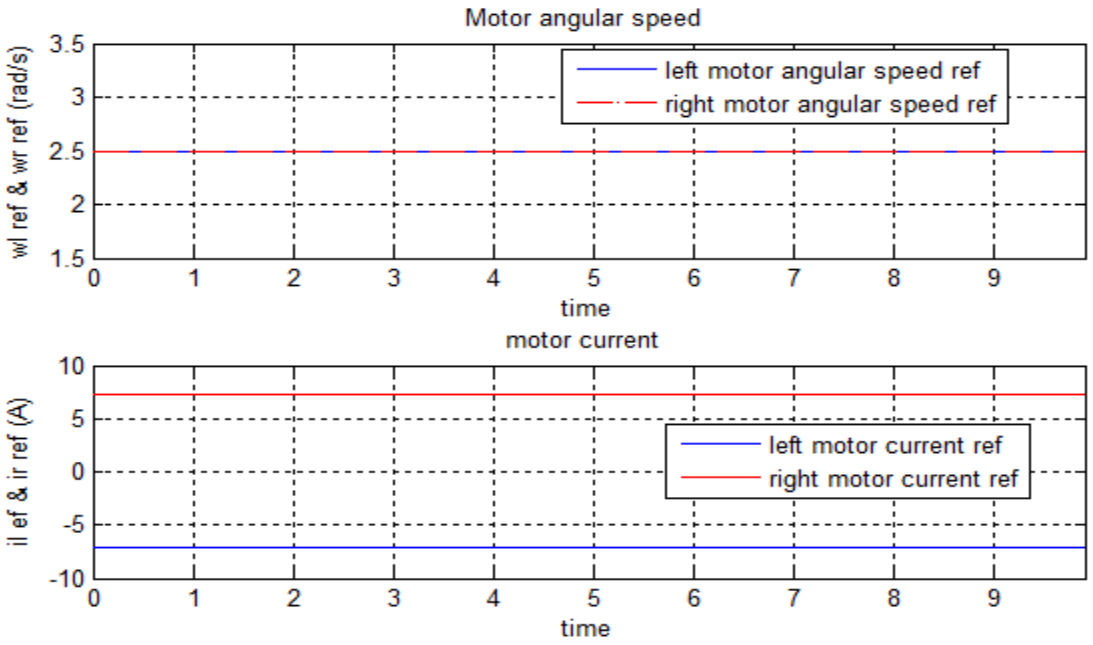


Figure 4.11: Reference current and motor angular speeds

The figure bellow is the graph generated using MATLAB to act as desired robot forward speed, angular speed and left, right motor control signals that the actual robot has to be actuated to fellow the desired crop row obtained from the image in the camera.

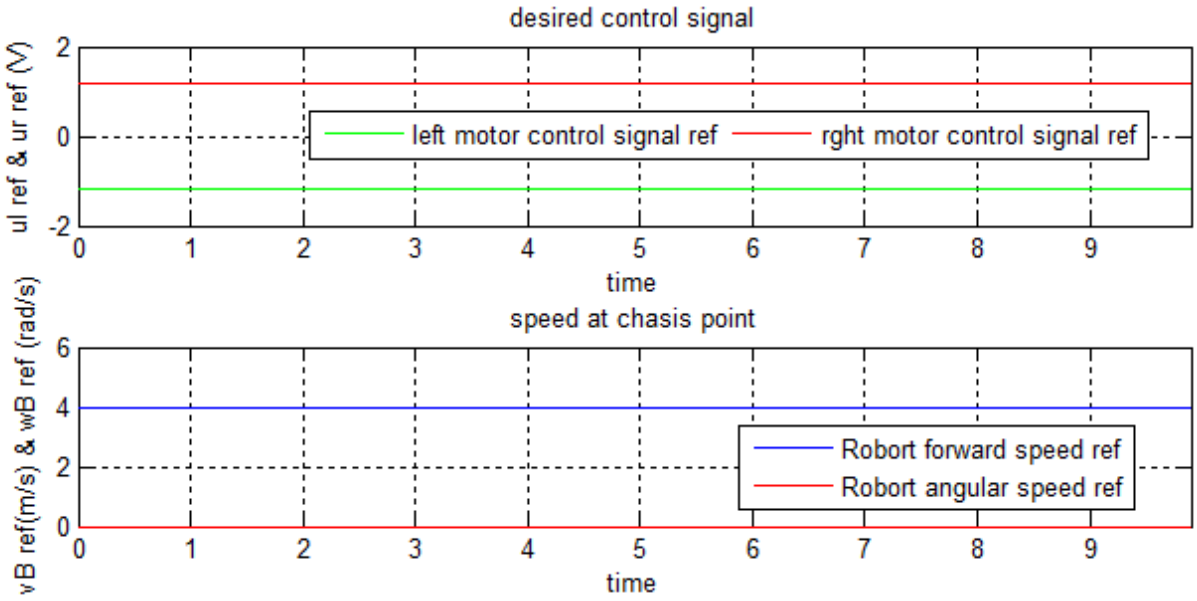


Figure 4.12: Desired manipulated variables and robot speed

4.3. System response with model predictive controller

As the open loop response is explained above. A unit step input control signal is used as test input. Unlimited position ($x_{B,}, y_B$) and an angle (α) which increases with time is obtained. Specifically the angle measured from the desired crop path (x position only) is $1 * 10^{-8}\ rad$ .It is good that the angle is very small and negligible in the crop row but the robot is moving with varied aggressive forward speed at different time. The angular Speed of the robot also varies with time. As the final goal of the thesis is to control the position and orientation of the robot and fixing the maximum angle difference between crop path ($x_{r,}y_r$) and the robot actual chassis position ($x_{B,}, y_B$) a model predictive control is designed as shown below. The robot forward speed (v_B) and angular speed (ω_B) must be fixed to the reference speed ($4m/s, 0rad/s$) respectively.

4.3.1. Closed loop response without disturbance

This section deals with the analysis of the controller performance to track the reference trajectory without the disturbance. A Simulink block based model predictive control is shown below. The MPC performance is measured based on tracking error and cost function minimization which explained in chapter four. The design of the MPC for the differential drive wheeld agricultural robot shown below in figure 4.13 bellow. The diagram is draw in MATLAB Simulink environment which has the MPC block. It consists the plant dynamic and kinematic model inside. The controller input ports involves minimum and maximum constraint of the outputs ($x_{B,}, y_{B,}, \alpha_B$), minimum and maximum constraints of the input control signals or manipulated variables ($u_{L,}u_R$), reference variable ports, weights on the manipulated variables and output variables and quadratic program switch. The output port consists manipulated variable and cost displayer port.

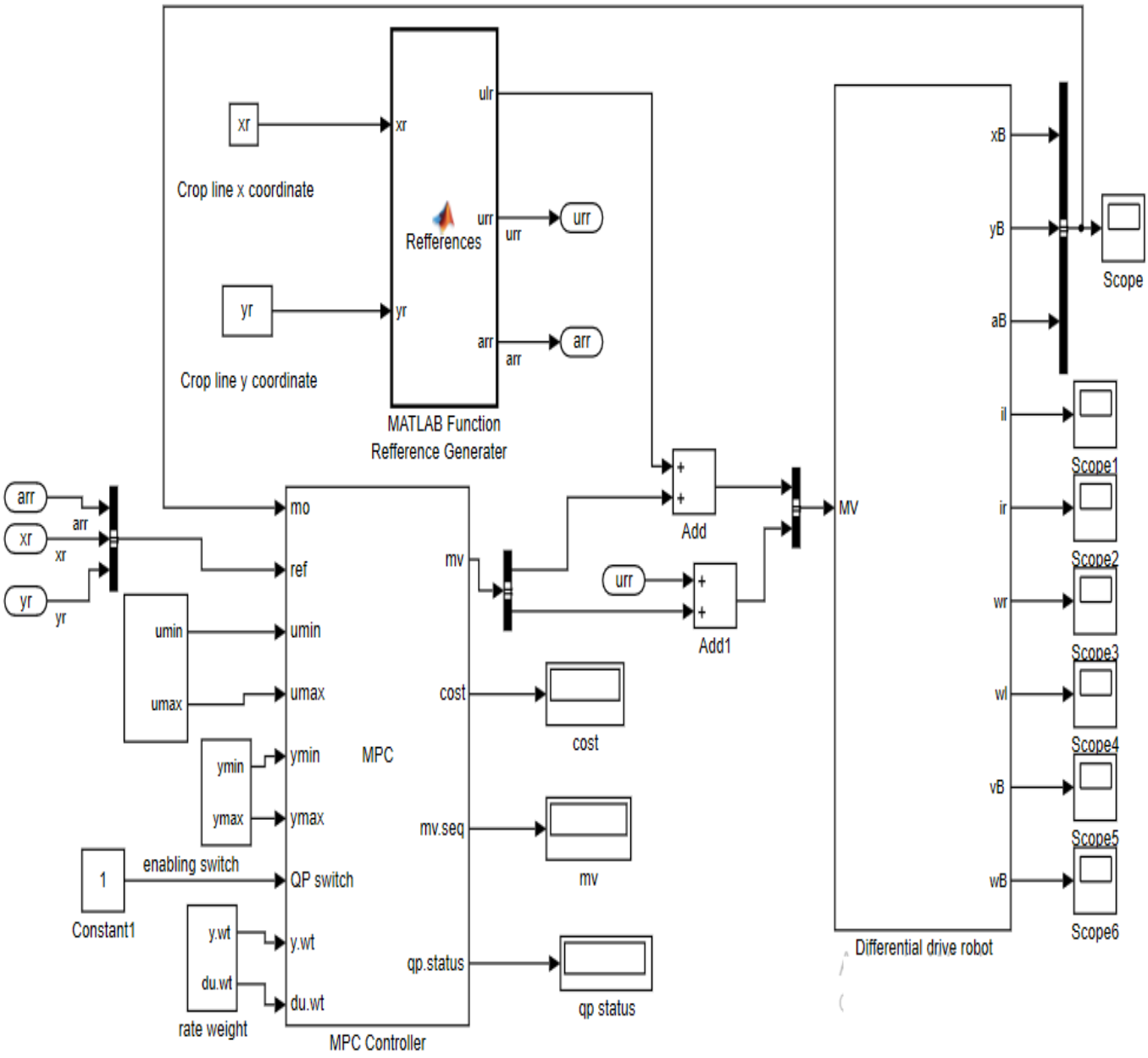
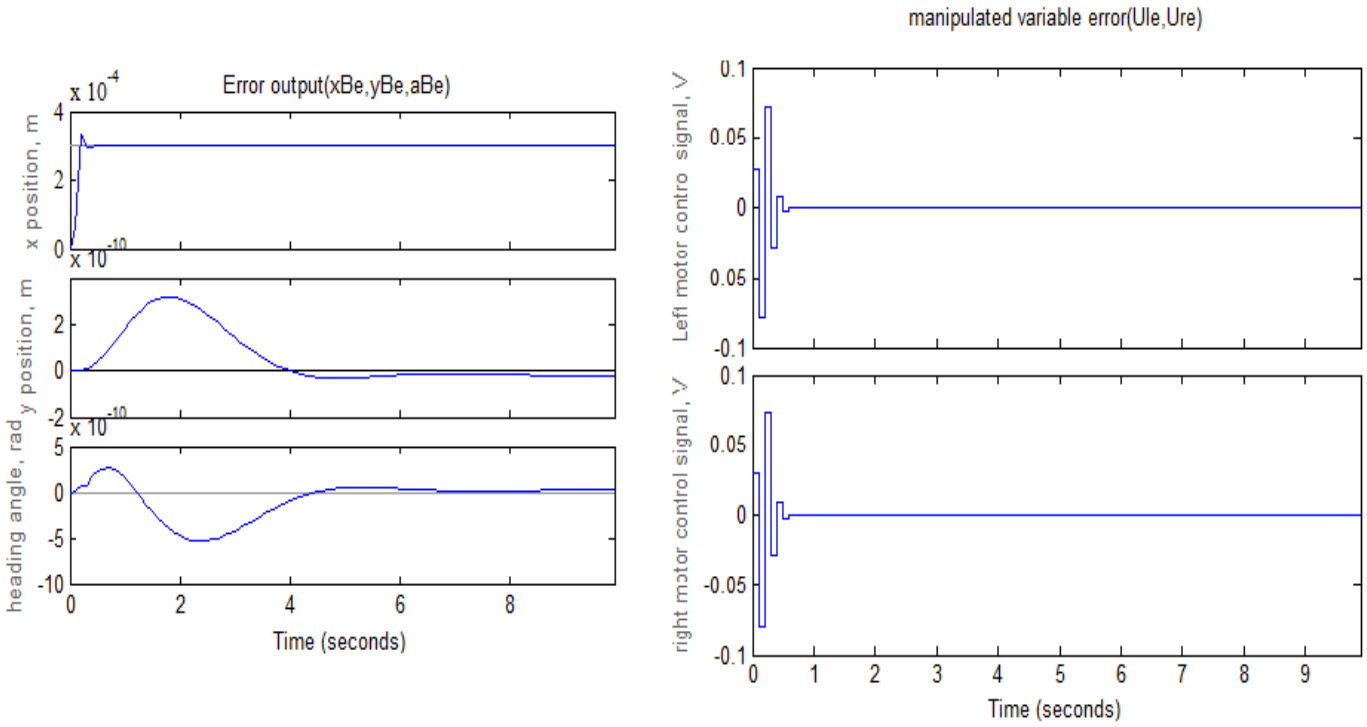


Figure 4.13: MPC design for differential drive wheeld agricultural robot without disturbance

The plant (differential drive wheeld agricaltural robot) design model used in the design of MPC is an error model wich mean the state variables are the difference between desired states and actual states where the actual states are obtained from the real robot and the desired staates are obtained from the refference that assumed to exactlly with crop row.The figures 4.14 a and b below shows the error output of the system and error manipulated variables that actuated the motor adding with the refference control signals illustrated above in figure 4.12.



(a) (b)

Figure 4.14: Error manipulated variables and outputs

The figure 4.15 bellow shows the comparision between desired control signal and actual control signal. The right and left control signals are varing between (-1.1,-1.3) and (1.3,1.1) volt for 0.5 seccond respectivelly and meets the desired value (-1.1893, 1.1992) volt from 0.5 second upto the end of simulation time.The left and right control signals have almost similar magnitude but different polarity.This situation is happened because both motors are supplied from common supply 24 V .Figure 4.16 illustriates the left and right angular speed of the robot .The desired value is obtained at 0.5 seecond after the control signals are applied.The left and right motors are moviing with(2.5,2.502) rad/s resecpectively until the end of the simulation time.This shows that the robot is moving in striaght line.

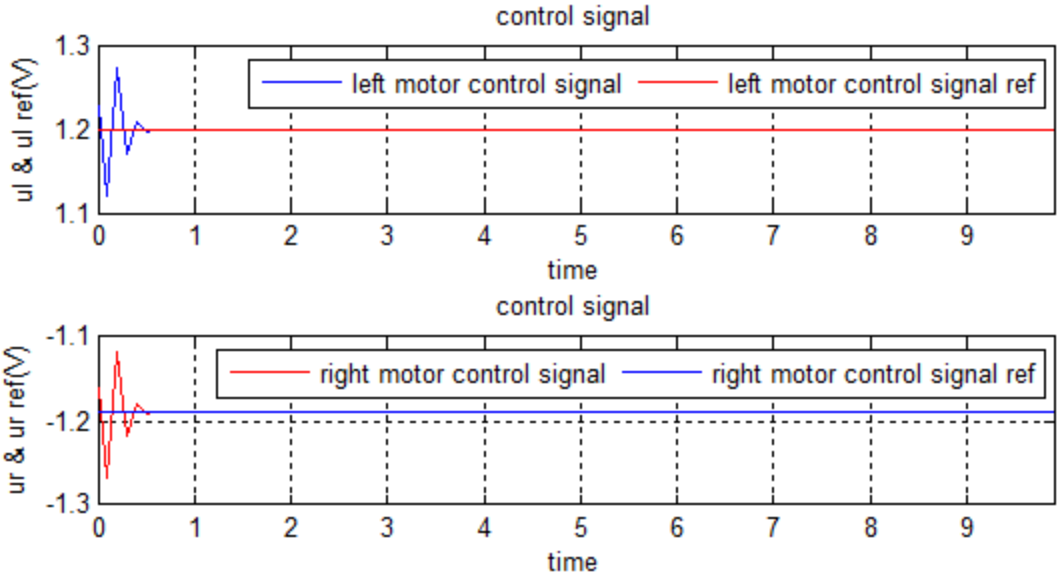


Figure 4.15: Comparison graph of actual control signal with desired control signal

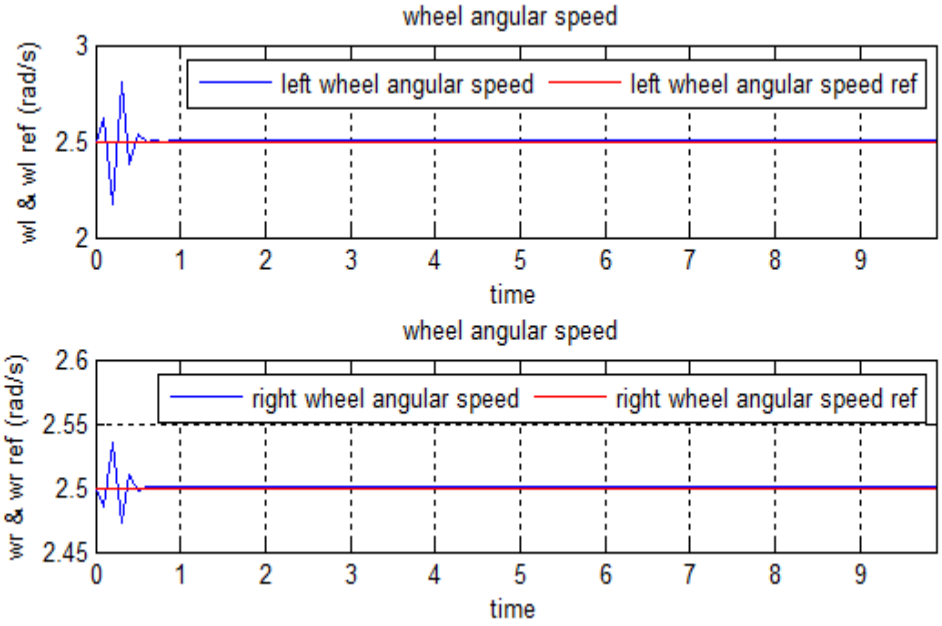


Figure 4.16: Comparison graph of actual wheel motor angular speed with desired angular speed

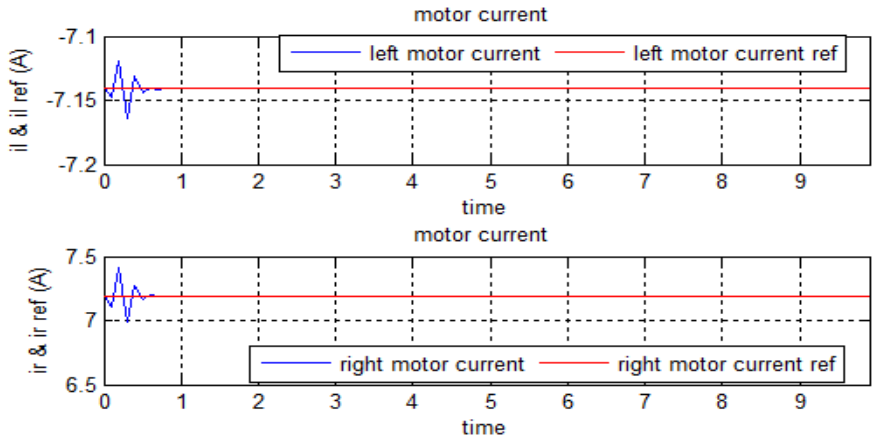


Figure 4.17: Desired sand actual current comparison

As fig 4.17 illustrates the desired left and right currents are obtained at 0.5 second exactly after the control signal is applied. Its magnitude is varying for 0.5 second with the order of the manipulated variable or control signals. Figure 4.18 bellow verifies that the forward speed and angular speed of the robot are kept constant with value 4 m/s and 0rad/s respectively after 0.5 second of applying the control signal. Unlike the open loop system the closed response shows the robot is moving with constant forward velocity.

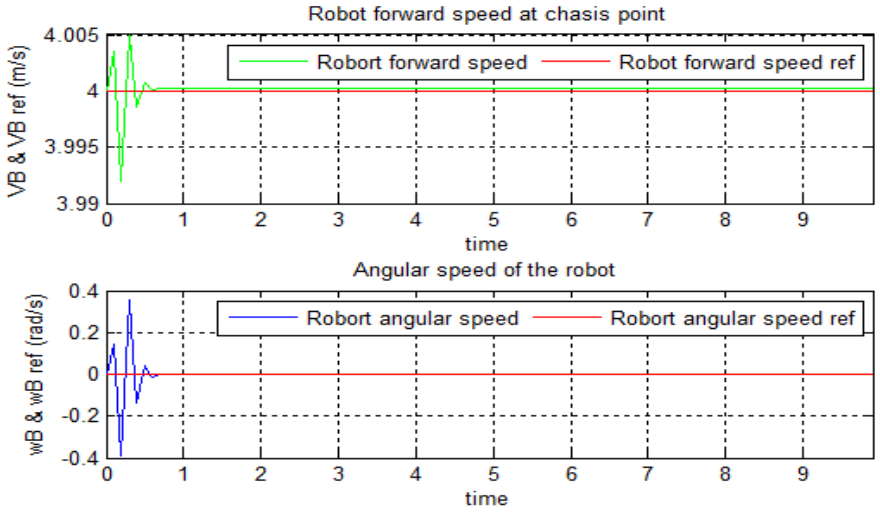
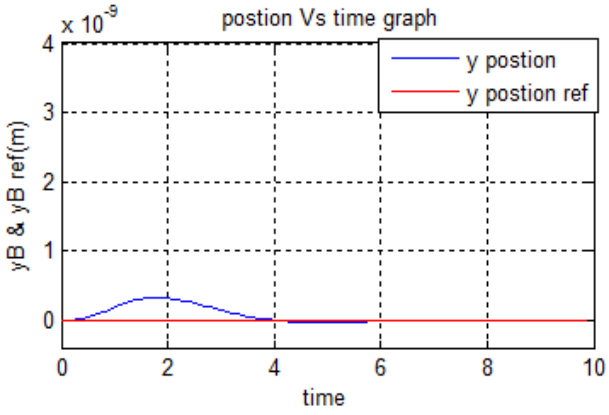


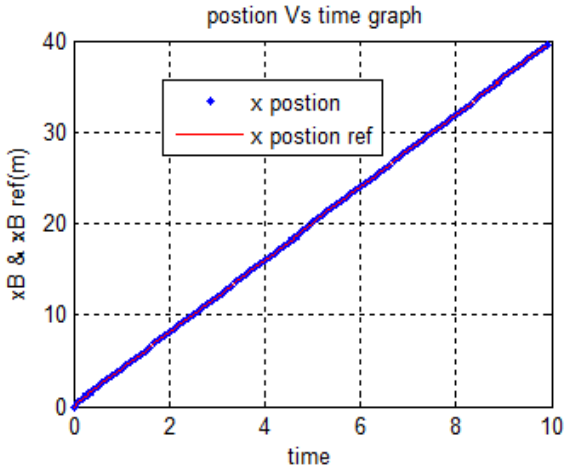
Figure 4.18: Robot forward speed and angular speed graph

The following graph (a, b, c, d) shows the outputs of the differential drive wheeld agricultural robot x_B position is exactly fits to the desired value whereas y_B position does not track the reference value. It deviates by $0.5 * 10^{-9}m$ from its reference but this is negligible in robot operating

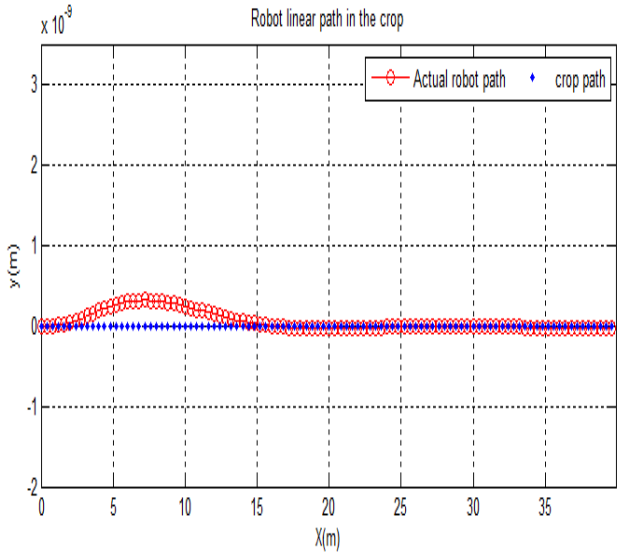
in the crop row. The maximum heading or orientation error is $-0.5 * 10^{-10} degree$.The negative sign indicates that rotation for the error is to the left of the desired crop row.



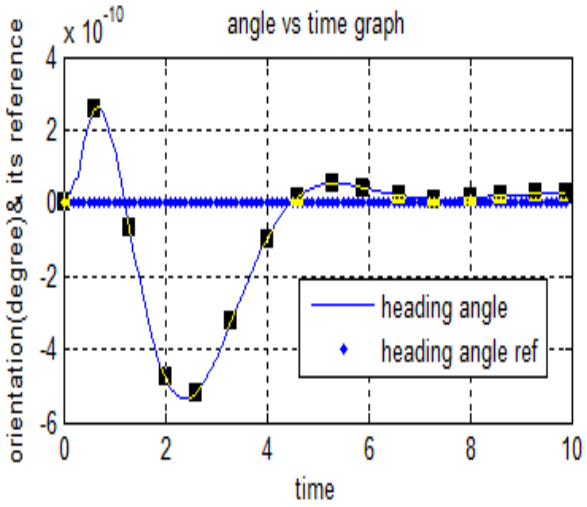
(a)



(b)



(b)



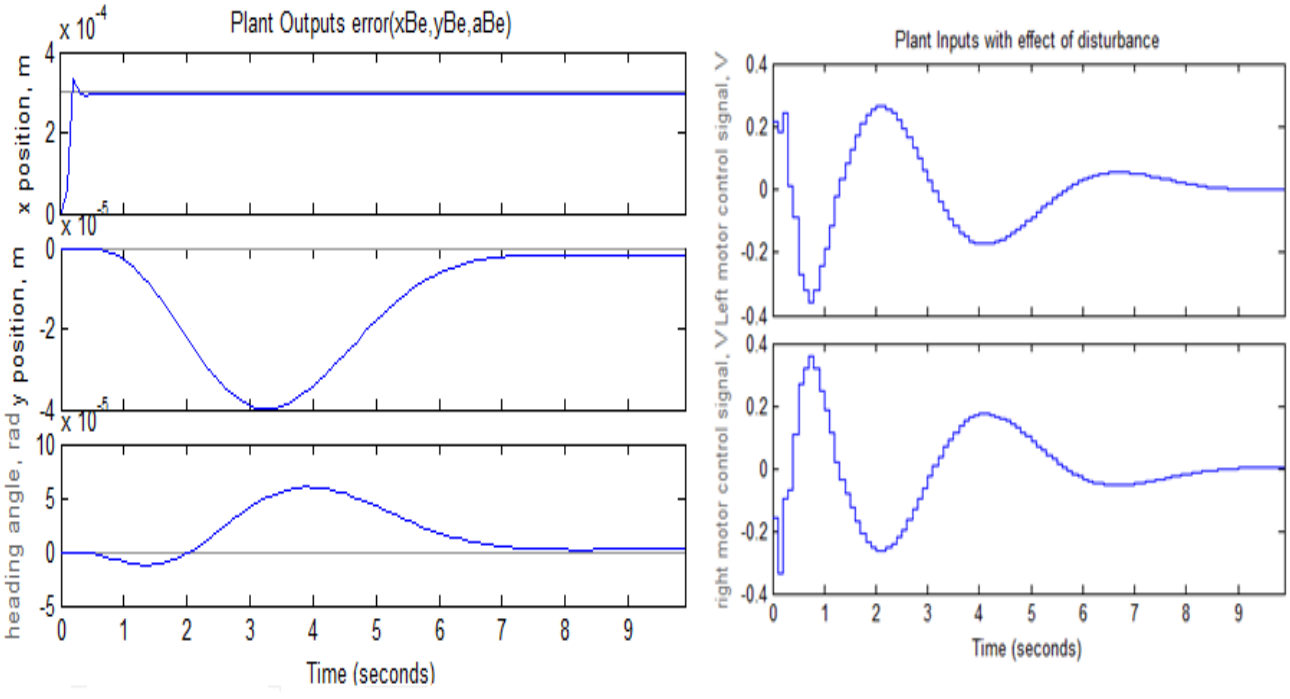
(d)

Figure 4.19: Robot position and orientation

4.3.2. System response with disturbance

This section deals with the effect of disturbance to the output $(x_{B,}, y_{B,}\alpha_B)$ of the differential drive wheeld agricultural robot as explained the disturbance model in chapter three .It is true that since the model of the robot involves mass of the robot; the robot mass is varied with the herbicide tanker variation. In this sub topic the effect of the disturbance is illustrated. The initial values of states of the robot is set to zero i.e. $x_B = 0, y_B = 0, \alpha = 0, i_L = 0, i_R = 0, \omega_L = 0, \omega_R = 0, v_B = 0 \; and \; \omega_B = 0$.As shown in the graph 5.19 below the disturbance does not affect the x position. Only the y position and heading error is increased with comparison of the response without disturbance explained above. The time to settle to zero change in control signal is increased, but the robot is moving in the desired path with small error. The peak change in angular error is $5 * 10^{-5}$ degree.

The design of the control system involves disturbance port in addition to the system in figure 4.13. The plant model involves the kinematic and dynamic model of the differential drive wheeld agricultural robot.



(a) (b)

Figure 4.20: Effect of disturbance on output error and input error

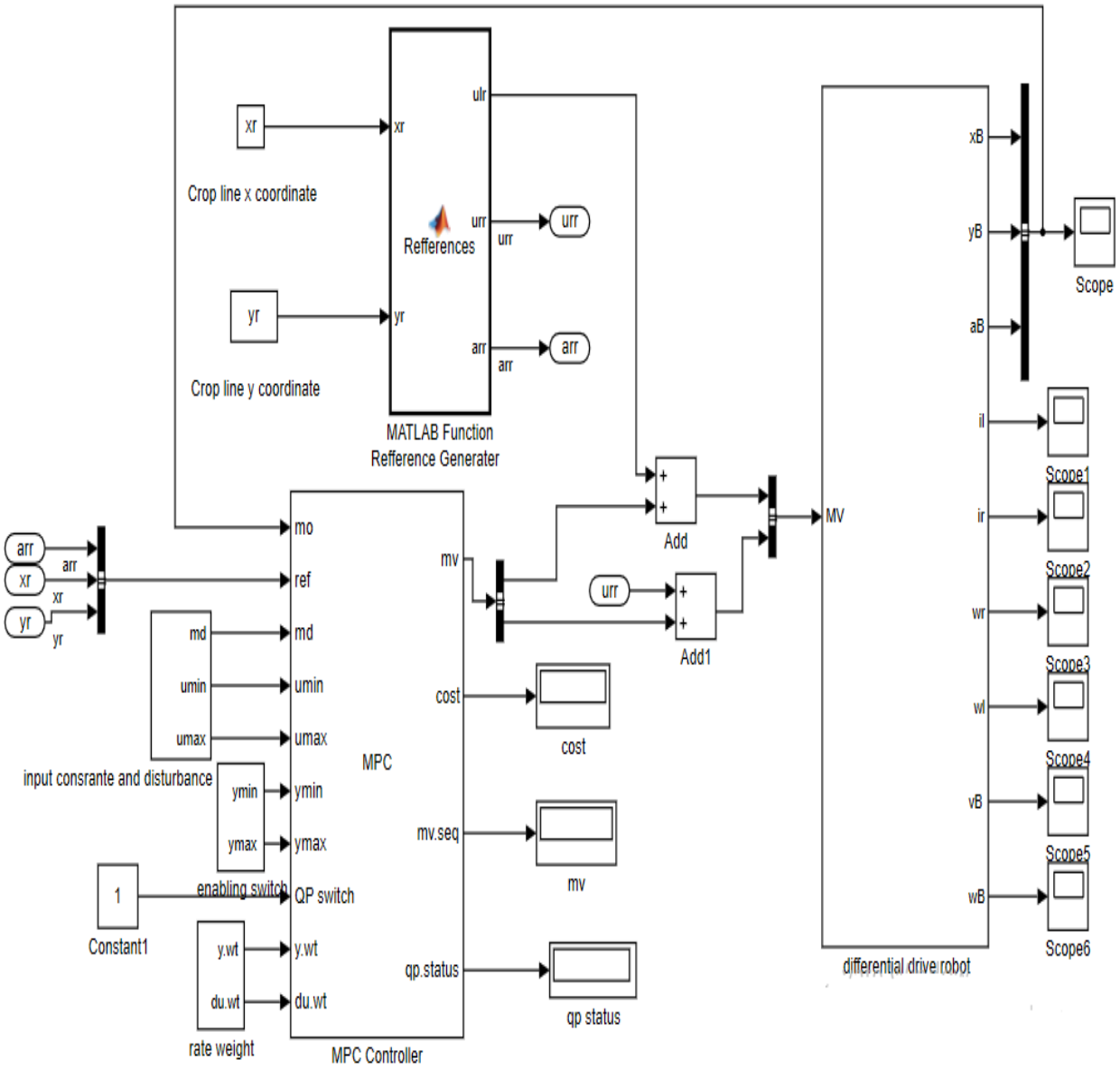
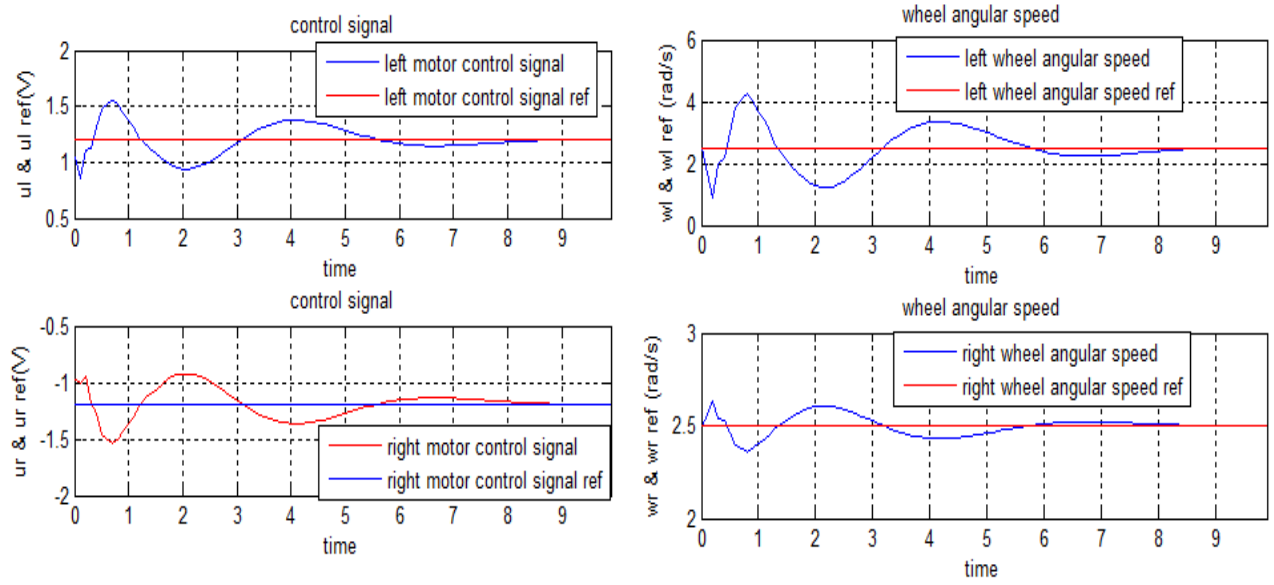


Figure 4.21: MPC design for differential drive wheeld agricultural robot with disturbance

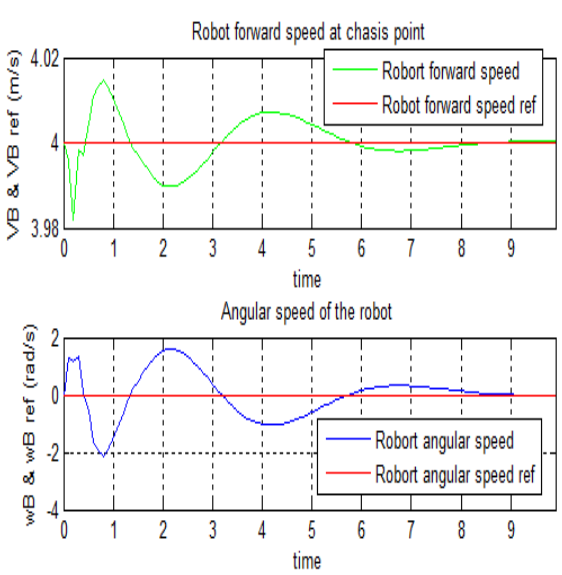
The actual control signal and desired control signal comparison graph is shown in figure 4.22 bellow. Once the error control signal is generated from the model predictive controller the robot motor actuated by sum of the error signal and the desired values or references control signals. It is possible to decrease the steady state error by changing the weight of the output variables. But this is designed to know the effect of disturbance without any change to controller parameters. As shown in graph 4.22 (b) the right and left wheels angular speed is varied with time for six seconds.

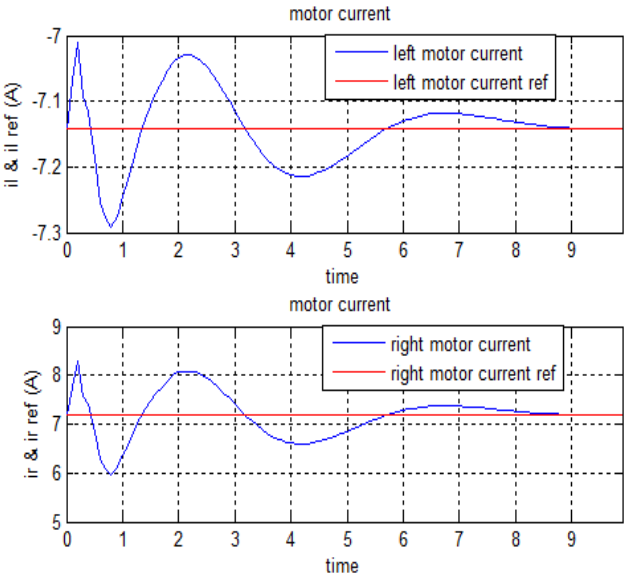


(a) (b)

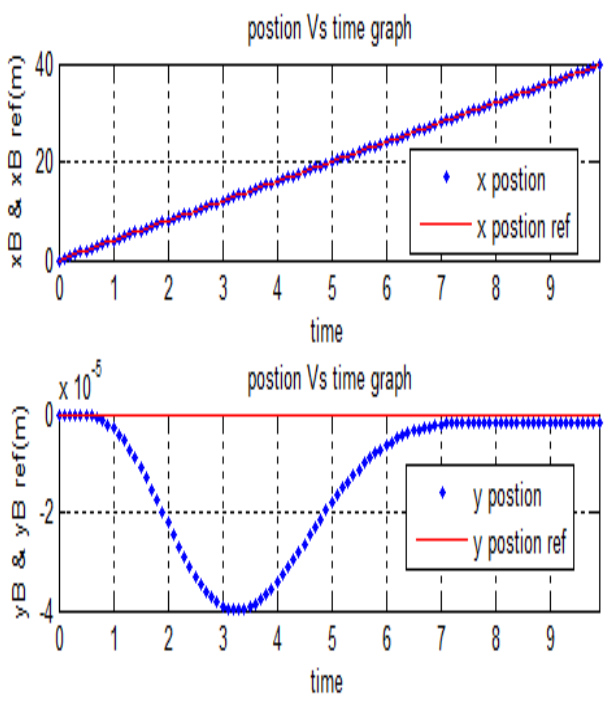
Figure 4.22: Robot wheel speed and control signals graph

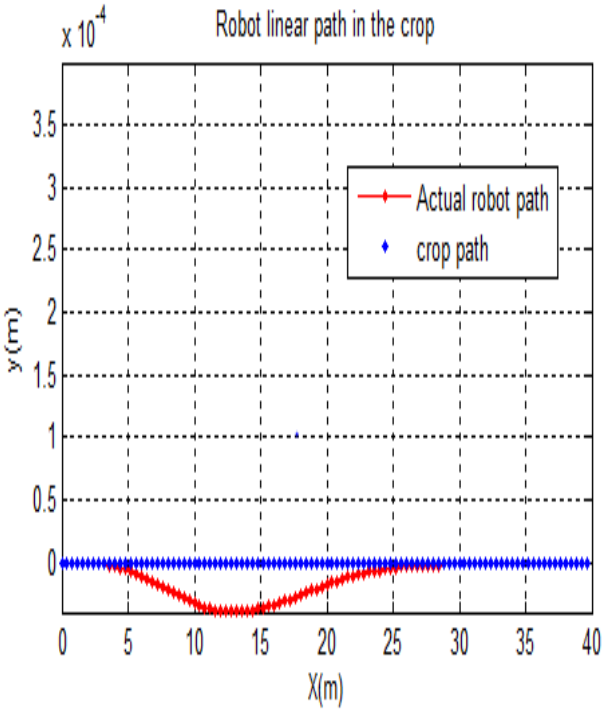
Fig 4.23 below shows the illustration of robot forward speed ,and angular speed ,left motor current ,right motor current and the trajectory of the robot in the irrigation .The maximum forward speed error is $0.02\ m/s$ .Almost the robot is moving with constant forward speed. The angular speed of the robot varies between -2 and 1.92 rad /s for the first six seconds but, it settles to the desired value after the perturbation. The linear path graph (d) indicates that the maximum error distance between the crop path is $-0.5 * 10^{-4} m$.it is clear that figure 4.23 (e) shows the angle tracks the desired heading with $0.5 * 10^{-4}$ degree maximum angular error.


Robot forward speed at chasis point
Robort forward speed
Robot forward speed ref
VB & VB ref (m/s)
time
Angular speed of the robot
Robort angular speed
Robort angular speed ref
wB & wB ref (rad/s)
time

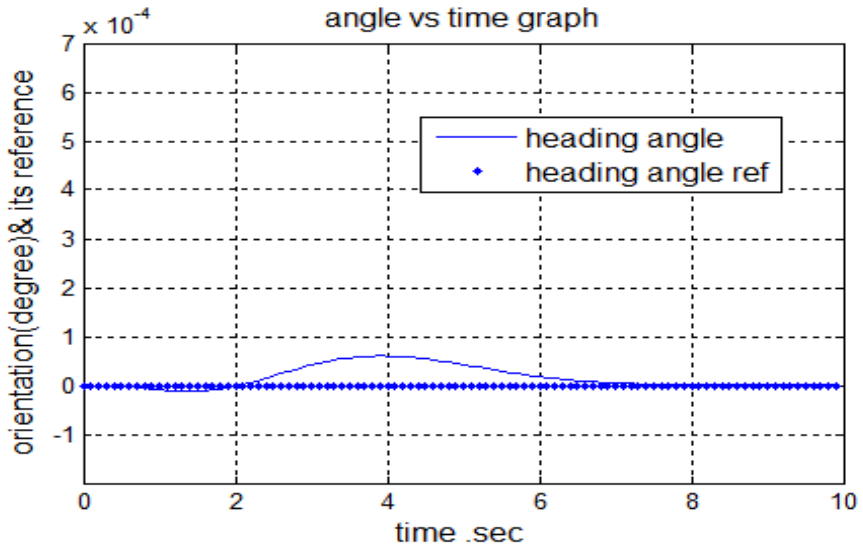

motor current
left motor current
left motor current ref
il & il ref (A)
time
motor current
right motor current
right motor current ref
ir & ir ref (A)
time

(a) (b)


postion Vs time graph
x postion
x postion ref
xB & xB ref(m)
time
x 10^-5
postion Vs time graph
y postion
y postion ref
yB & yB ref(m)
time


x 10^-4
Robot linear path in the crop
Actual robot path
crop path
y (m)
X(m)

(c) (d)



(e)

Figure 4.23: Robot angular speed, forward speed, current, position and orientation graph (a, b, c, d, e)

From the simulation of both the system with disturbance and without disturbance it is showing that the desired path is tracked with very small position error and angular error. Since the operation of the robot is to spray herbicide to the crops in the irrigation it has to move with constant forward speed. Both simulation verifies that the robot is moving with $4m/s$ in the ideal crop row.

4.4. System performance description

As the target of the robot is to follow the crop line error minimization and cost function minimization are the measures of the performance of the controller. The maximum angular error for the system with disturbance is $0.5 * 10^{-4}$ degree, but without considering the disturbance the controller achieves $0.5 * 10^{-10}$ degree angular error for the straight line crop row following robot. Those quantities' are measurable ideally, but it is not possible to measure on ground. It will be assumed as no error during string the robot.

The state variables of the robot are the difference between the actual state variables and the reference states. There for, the controller out puts are error control signals.

The change in error control signals (Δu_e) generated from the controller, the change in error output of the plant and the cost function that describes the control performance are shown below in table

7. Zero cost function is obtained at 6th sample time instant of the total step of the simulation. The total sample time instants is hundred, but the cost function is put for $k = 0:8$. Zero cost functions obtained at $k = 5$, which mean that the change in control signal is zero and the set point is tracked.

Table 7: Cost function

k	Cost(j)	Δu_{Le}	Δu_{Re}	$x_{Be,}10^{-3}(m)$	$y_{Be}\,(m)$	$\alpha_{Be}(deg)$
0	0.0008	0.0286	0.0290	0	0	0
1	0.0063	0.0786	-0.0797	0.0612	0.0000	0.0000
2	0.0053	0.0719	0.0734	0.3349	0.0000	0.0000
3	0.0008	-0.0282	-0.0295	0.2986	0.0000	0.0000
4	0.0001	0.0085	0.0089	0.2982	0.0000	0.0000
5	0.0000	-0.0021	-0.0022	0.3009	0.0000	0.0000
6	0.0000	0.0003	0.0004	0.2995	0.0000	0.0000
7	0.0000	0.0001	-0.0000	0.3001	0.0000	0.0000
8	-0.0000	-0.0000	0.0000	0.3000	0.0000	0.0000

As the controller tracks the set point errors the error control signals is added with reference control signals to actuate the robot. The actual values of the plant output and actual control signals that actuates the motors are shown in the table below The total simulation time is $k = 0:100$. But , results of the actual plant output ($y_{B,}, x_B, \alpha$) and control signals ($u_L\ ,\ u_R$) are put for $k = 0:38$. The robot can only stop movement if the crop row trajectories not captured.

CHAPTER FIVE

5. CONCLUSIONS, RECOMMENDATIONS AND FUTRE WORKS

5.1. Conclusions

The model predictive controller was designed to control the motion of the robot in the ideal crop row. Hough transform method of fitting a line to the crop image was described and developed. Mathematical model of the robot and the disturbance was developed as paramount of the MPC design for the control of the differential drive wheel agricultural robot. The model of the robot is described in state space form and its augmented model was derived and the same was used in the simulation. The input and output of the differential drive wheeld robot is constrained and input and output weights are selected. The controller used N=5, control horizon, P=26, prediction horizon with 0.1 sampling period.

The main focus the work was how to design model predictive controller for agricultural robot. This thesis reviewed and discussed some existing agricultural robot controlling techniques and showed the ability to track the straight path of the crop path gathered from the camera through image processing.

The performance of the controller was analyzed using the cost function taken to track the desired path or reference of the crop row. An error minimization ability was also taken as performance minimization of the controller.

The simulation results were divided into three parts. First, the open loop response was simulated for unit step input and the robot forward speed was different from time to time. The error between the reference line and actual line was big. Second, the closed loop response without consideration of the modeled disturbance was simulated and analyzed. Third, the closed loop response with consideration of the effect of the disturbance was simulated.

The result showed that the closed loop system without consideration of the disturbance was very good to track the path with cost function $j = 1.0653 * e^{-10}$.This shows that the controller tracks the desired path with small angular error $0.5 * 10^{-10}$ degree. Unlike the open loop response, the robot able to move with constant forward speed 4m/s.The control system with consideration of

disturbance showed that the cost function taken to track the desired path was $j = 1.0653 * e^{-10}$ approximately but the angular error was $0.5 * 10^{-4}$ degree And the robot able to move with constant forward speed with maximum error $0.02m/s$ from the desired forward speed $4m/s$.

5.2. Recommendations

In this thesis work simulation model was developed and used to the performance of the straight line crop row tracking. However, practical hardware test has to be conducted to verify the ability of the controller to track the desired path.

The results obtained in this work pushed me to do the robot practically. This work will lead me to study and investigate skill on how the image processing of the crop row do and interconnected with the controller. The performance of the controller shows that MPC is appropriate controller for MIMO systems with constraint.

5.3. Suggestions for future work

In this work the model used for the design of the model predictive controller consists dynamics of the chassis, kinematics and motor dynamics. For the better tracking the dynamics of the power stage will be consisted. In this work the controller tracks straight path and in the future robot turning strategy will be consisted. The constraint is only applied to the input and output, but in the future work constraints on wheel speeds will be involved.

In this thesis the power supply is considered as given. But, for future, model predictive control based solar power tracking will be studied. In this work the crop row detection starts from camera model and studies Hough transform method fitting line to crop. For future details of image processing of line detection algorism will be developed.

REFERENCES

[1] B. Benet, R. Lenain. Multi-sensor fusion method for crop row tracking and travers ability operations 18 Apr 2018

[2] F. Dong, O. Petzold, W. Heinemann, et al. Time-optimal guidance control for an agricultural robot with orientation constraints. Compute. Electron. Ag. 99(2013), 124-131

[3] Jarle Dørum. Autonomous Navigation and Row Detection in Crop Fields Using Computer Vision, January 2015

[4] Introduction to Model Predictive Control. Lecture notes prepared by M. Scott Trimboli. M. Scott Trimboli ECE5590, 1–2. 2015

[5] Robotics in Agriculture. Types and Applications By Robotics Online Marketing Team POSTED 12/12/2017

[6] Lauren Lieu CMU-RI-TR-18-47. Predictive Control of Constrained Nonlinear Systems, December 2018.

[7] Jin-lin XUE, Bo-wen FAN, Xin-xin ZHANG and Yong FENG. An Agricultural Robot for Multipurpose Operations in a Greenhouse 15 August 2017

[8] Vishnu R Desaraju. "Safe, Efficient and Robust Predictive Control of Constrained Nonlinear Systems", In (2017)

[9] Samer Abdelmoeti and Raffaella Carloni. "Robust control of UAVs using the parameter space approach". International Conference on. IEEE. 2016, pp. 5632–5637

[10] Jian Wang et al. "Non-cascaded dynamic inversion design for quadrotor position control with L1 augmentation". Navigation & Control, Delft, Netherlands. 2013

[11] Evgeny Kharisov, Naira Hovakimyan, and Karl A° stro¨m. "Comparison of several adaptive controllers according to their robustness metrics". Navigation, and Control Conference. 2010, p. 8047

[12] Samir Bouabdallah and Roland Siegwart. "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor", In 2005

[13] Daewon Lee, H Jin Kim, and Shankar Sastry. "Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter", 2009, pp. 419–428

[14] Guilherme V Raffo, Manuel G Ortega, and Francisco R Rubio. “An integral predictive/nonlinear control structure for a quadrotor helicopter”, 2010, pp. 29–39

[15] Robert F Stengel. Optimal control and estimation. Courier Corporation, 1994.feedback control of constrained nonlinear stochastic systems”, 2005, pp. 300–306

[16] Jay H Lee. “Model predictive control: Review of the three decades of development”,2011, pp. 415.

[17] Shuyou Yu et al. “Tube MPC scheme based on robust control invariant set with application to Lipschitz nonlinear systems”,2013, pp. 194–200

[18] Patrick Bouffard, Anil Aswani, and Claire Tomlin. “Learning-based model predictive control on a quadrotor: Onboard implementation and experimental results”, 2012, pp. 279–284

[19] Chris J Ostafew, Angela P Schoellig, and Timothy D Barfoot. “Learning-based nonlinear model predictive control to improve vision-based mobile robot path-tracking in challenging outdoor environments, 2014, pp. 4029–4036

[20] Sethu Vijayakumar, Aaron D’souza, and Stefan Schaal. “Incremental online learning in high dimensions”,2005, pp. 2602–2634

[21] Arjan Gijsberts and Giorgio Metta. “Real-time model learning using incremental sparse spectrum gaussian process regression”,2013, pp. 59–69

[23] Yoshiaki Kuwata, Arthur Richards, and Jonathan How. “Robust receding horizon control using generalized constraint tightening”, 2007, pp. 4482–448

[24] Wilbur Langson et al. “Robust model predictive control using tubes”, 2004, pp. 125–133

[25] Guilherme V Raffo, Manuel G Ortega, and Francisco R Rubio. “An integral predictive/nonlinear control structure for a quadrotor helicopter”, 2010, pp. 29–39

[26] Arthur Richards and Jonathan How. “Robust model predictive control with imperfect information”,2005, pp. 268–273

[27] Yulin Zhang, Daehie Hong, Jae H. Chung, and Steven A. Velinsky. Dynamic Model Based Robust Tracking Control of a Differentially Steered Wheeled Mobile Robot, June 1998

[4] Trygve Utstumo, Therese W. Berge, Jan Tommy Gravdahl.Non-linear Model Predictive Control for constrained robot navigation in row crops. 978-1-4799-7800-7/15/S31.00 ©2015 IEEE

[28] Shubham Dhage1, Pradip Patil2, Data Kande3, Dr. Prakash Patil4.Wireless Controlled Multipurpose Agricultural Robot. Vol. 7, Issue 5, May 2018.

[29]. Urdal, F., Utstumo, T., Ellingsen, S. A., Vatne, J. K. & Gravdahl, T. Design and control of precision drop-on-demand herbicide application in agricultural robotics in The 13th International Conference on Control, Automation, Robotics and Vision, Singapore (2014)

[30] K¨uhne, F., Lages, W. F., and Silva, J. Mobile Robot Trajectory Tracking using Model Predictive Control, 2005, pp 1–7

[31] Vougioukas, S.G. Reactive trajectory tracking for mobile robots based on nonlinear model predictive control. 2007, p. 3074

[32] Klancˇar, G., Škrjanc, I., Tracking-error model-based predictive control for mobile robots in real time. Robotics and Autonomous Systems 55 (6), 460–469. 2007

[33] F. Dušek D. Honc P. Rozsíval Mathematical model of differentially steered mobile robot. 18th International Conference on Process Control June 14–17, 2011

[34] J. J. Craig, Introduction to robotics: mechanics and control, 3/E. Pearson Education India, 2009

[35] A. Z. Alassar, I. M. Abuhadrous, and H. A. Elaydi, "Comparison between FLC and PID Controller for 5DOF robot arm," in 2010 2nd International Conference on Advanced Computer Control, vol. 5, pp. 277-281

[36] Kuhne, F., Lages, W.F. and da Silva Jr, J.M.G., 2004. Model predictive control of a mobile robot using linearization. In Proceedings of mechatronics and robotics (pp. 525-530)

APPENDIX

A. Simulation of mpc using mfile command in matlab

```
% Control of a Multi-Input Multi-Output DDWAR
% This command shows how to design model predictive controller with three
% measured output, two manipulated variables in a typical workflow.
% Parameters of differentially drive wheeld agricultural robot (DDWAR)
R=4; %R [ohm] is motor winding resistance.
U0=24; %U0 [V] is source voltage.
Rz=0.4; %Rz[ohm] is internal resistance.
L=0.025; %L [H] is motor inductance.
K=0.00007; %[K] [kg.m2. s-.2A-1] is the back EMF constant
J=0.0025; %J [kg.m2] is moment of inertia.
ar=0; %desired angle
kr=0.00005; %Kr [kg.m2. s-1] is coefficient of rotation resistance
Ll=0.5; %Ll [m] is distance of the left wheel from point b,
Lr=0.5; %Lr [m] is distance of the right wheel from point b
kw=0.0350; %K? [kg.m2. s-1] is resistance coefficient against rotary motion
kv=0.0005; %Kv [kg. s-1] is resistance coefficient against linear motion
Jb=0.05; %JT + mlT.JT [kg.m2] is moment of inertia with respect to center of
% gravity and lT [m] is distance between center of gravity and point b.
Jr=0.03; %rotational moment of inertia
T=0.1; %control interval or sample time
vr=0.8; %reference velocity
PG=0.25; %PG is the gear box transmission ratio
r=0.4; %r [m] is semi-diameter of the wheels
m=2; %m [kg] is robot mass
rG=r/PG;
% Introducing new parameters
al=kr+(kv*Lr*rG^2)/(Ll+Lr);
aR=kr+(kv*Ll*rG^2)/(Ll+Lr);
bl=J+(m*Lr*rG^2)/(Ll+Lr)
```

```
bR=J+(m*Ll*rG^2)/(Ll+Lr)
cl=kr*Ll+(kw*rG^2)/(Ll+Lr);
cR=kr*Lr+(kw*rG^2)/(Ll+Lr);
dl=J*Ll+(Jb*rG^2)/(Ll+Lr);
dR=J*Lr+(Jb*rG^2)/(Ll+Lr);
A1=K*(dR+bR*Ll)/bl*dR+bR*dl;
A2=K*(dR-bR*Lr)/bl*dR+bR*dl;
A3=-(dR*al+bR*cl)/bl*dR+bR*dl;
A4=-(dR*aR-bR*cR)/bl*dR+bR*dl;
B1=K*(dl-bl*Ll)/bl*dR+bR*dl;
B2=K*(dl+bl*Lr)/bl*dR+bR*dl;
B3=-(dl*al-bl*cl)/bl*dR+bR*dl;
B4=-(dl*aR+bl*cR)/bl*dR+bR*dl;
C1=rG*Lr/Ll+Lr;
C2=rG*Ll/Ll+Lr;
C3=rG/Ll+Lr;
bb=Ll*rG^2/(Ll+Lr)
bb=Lr*rG^2/(Ll+Lr)
```

% actual robot state space representation

```
A=[-(R+Rz)/L -Rz/L -K/L 0 0 0 0 0; -Rz/L -(R+Rz)/L 0 -K/L 0 0 0 0;
   A1 A2 A3 A4 0 0 0 0; B1 B2 B3 B4 0 0 0 0;0 0 C1 C2 0 0 0 0;
   0 0 -C3  C3 0 0 0 0;0 0 0 0 cos(ar*T) 0 1 0 -vr*sin(ar*T);
   0 0 0 0 sin(ar*T) 0 0 1 vr*cos(ar*T);0 0 0 0 0 1 0 0 T];
B= [U0/L 0;0 U0/L;0 0;0 0;0 0;0 0;0 0;0 0];
C= [0 0 0 0 0 0 1 0 0;0 0 0 0 0 0 0 1 0;0 0 0 0 0 0 0 0 1];
D=0;
sys3=ss (A, B, C, D)
eig(A)
co= ctrb(sys3);
controllability = rank(co)
% stablity
```

```
isstable(sys3)
% observability
ob = obsv(sys3);
observability=rank(ob)
eig(A)
```

% augmented state space for actual robot

```
[m1, n1] =size(C);
[n1, n_in] =size(B);
A_e=eye (n1+m1, n1+m1);
A_e (1: n1,1: n1) =A;
A_e (n1+1: n1+m1,1: n1) =C*A;
B_e=zeros (n1+m1, n_in);
B_e (1: n1, :)=B;
B_e (n1+1: n1+m1, :)=C*B;
C_e=zeros (m1, n1+m1);
C_e (: n1+1: n1+m1) =eye(m1,m1);
states = {'ildot' 'irdot' 'wldot' 'wrdot' 'VB' 'WB' 'xb' 'yb' 'al' 'xbdot' 'ybdot' 'aldot'}
inputs = {'left motor control signal'; 'right motor control signal'}
outputs = {'xb' 'yb' 'al'}
sysa=ss (A_e, B_e, C_e, D,'State name', states,' input name', inputs, 'output name', outputs);
SS=c2d(sysa)
sysreal=ss(SS,'min')
% controllability
co= ctrb(sysreal);
controllability = rank(co)
% stablity
isstable(sysreal)
% observability
ob = obsv(sysreal);
observability=rank(ob)
% roots(sysreal)
```

```
modlaa=sysreal
% Assign names and units to I/O variables.
modlaa. Input Name= {'Left motor control signal'; 'right motor control signal'};
modlaa. Output Name= {'x position';' y position'; 'heading angle'};
modlaa. Input Unit = {'V'; 'V'};
modlaa. Output Unit = {'m'; 'm'; 'rad'};
% Define constraints on the manipulated and measured output variables.
Modlaa.Plant1. Output Group = {[3],'Measured'};
clear Input Specs Output Specs
Input Specs (1) =strut('Min',-2,'Max',2,'RateMin',-10,'Ratemax',10);
Input Specs (2) =strut ('Min', -2,'Max',2,'RateMin', -10,'Ratemax',10);
Output Specs (1) =strut ('Min', -2,'Max',40);
Output Specs (2) =strut ('Min', -2,'Max',4);
Output Specs (3) =strut('Min',0,'Max',180);
Weights=strut ('Manipulated Variables', [.1 .1], ...
  'ManipulatedVariablesRate', [.1 .1], ...
  'Output Variables', [3000 3000 3000]);
% setup an mpc controller object.
clear Model
Modla.Plant=modlaa;
Prediction Horizon =26;
Control Horizon =5;
MLLn =mpc (modlaa, T, Prediction Horizon, Control Horizon, Weights, Input Specs, Output Specs)
% Closed-loop MPC Simulation Using the Command SIM
Tstop=5; % simulation time
Tf=round(Tstop/T); % number of simulation steps
r1= [0.0003 0 0]; % Reference trajectory
% Specify noise signals. In order to do this,
% I create the MPC simulation object 'SimOptions'.
```

```
xmpc=mpcstate(MLLn);
d= [];
v= [0.000003 0.000003];
x=zeros (9,1);
SimOptions=mpcsimopt(MLLn);
SimOptions.plantInitialState =x; % plantInitialState
SimOptions.ControllerInitialState=xmpc; %Controller Initial State
SimOptions.RefLookAhead='on';
SimOptions.unmeasuredDisturbance =d;
SimOptions.OutputNoise= []; % output measurement noise
SimOptions.InputNoise=[v]; % noise on manipulated variables
% Run the closed-loop simulation and plot results.
% Run the closed-loop simulation and save the results to workspace.
sim (MLLn, Tf, r1, SimOptions);
[y2, t2, u2, all] =sim (MLLn, Tf, r1, SimOptions)
% MPC Control Action (Step-by-step Simulation)
% I may just want to compute the MPC control action inside our simulation
% First I get the discrete-time state-space matrices of the plant.
% I store the closed-loop MPC trajectories in arrays YY, UU, XX.
[Al, Bl, Cl, Dl] =data(modlaa);
YY= [];
UU= [];
XX= [];
jj= [];
% Main simulation loop
for k=0: round(Tstop/T)
        XX= [XX, x];
% Define measured disturbance signal
        v=0.03;
        d=0.03;
        Q=diag (1,1);
```

```
        R=diag (1);
        % Compute MPC law
        y=Cl*x;
        YY= [YY, y]
% Plant equations state update
        ud=mpcmove (MLLn, xmpc, y, r1);
        x=Al*x+Bl*ud;
        UU= [UU, ud];
        j=(y-r1')'*R*(y-r1') +ud'*Q*ud;
        jj= [jj, j];
end
plot (k, jj (,1) *size(k))
legend('cost')
[~, Info] =mpcmove (MLLn, xmpc, y, r1)
topt=Info. Topt;
yopt=Info. Yopt;
uopt=Info. Opt;
close all
% Reference generating model
xr=4*t2;
yr=zeros(size(t2));
vBr=4*ones(size(t2));
aref=zeros(size(t2));
wBr=zeros(size(t2));
wRr=(vBr+Lr*wBr)/rG;
wLr=-(wBr*(Ll+Lr)-wRr*rG)/rG;
MRr=(Ll*rG*kv*vBr+kw*wBr)/Ll+Lr;
MLr=rG*kv*vBr-MRr;
ilr=((MLr+kr*wLr)/K) *0.001;
irr=((MRr+kr*wRr)/K) *0.001;
ulr=((R+Rz) *ilr+Rz*irr+K*wLr)/U0;
```

```
urr=((R+Rz) *irr+Rz*ilr+K*wRr)/U0;
All= [wRr wLr MRr MLr ilr irr ulr urr]';
% error plus references
udr=urr+u2(:,2);
udl=ulr+u2(:,1);
uhg=[udl udr]
xrd=xr+y2(:,1)
yrd=yr+y2(:,2)
ard=aref+y2(:,3)
yuubb=[xrd yrd ard]
ild=all(:,1)+ilr;
ird=all(:,2)+irr;
wld=wLr+all(:,3);
wrd=wRr+all(:,4);
vBd=vBr+all(:,5);
wBd=wBr+all(:,6);
% Analysis
get(MLLn)
display(MLLn)
X=trim (MLLn, y, ud)
% How can we know if the designed MPC controller will be able to reject
% constant output disturbances and track constant set-point with zero offsets in steady-state?
% Compute the DC gain from output disturbances to controlled outputs using CLOFFSET.
DC=cloffset(MLLn)
% A zero gain means that the output will track the desired set-point.
review(MLLn)
```

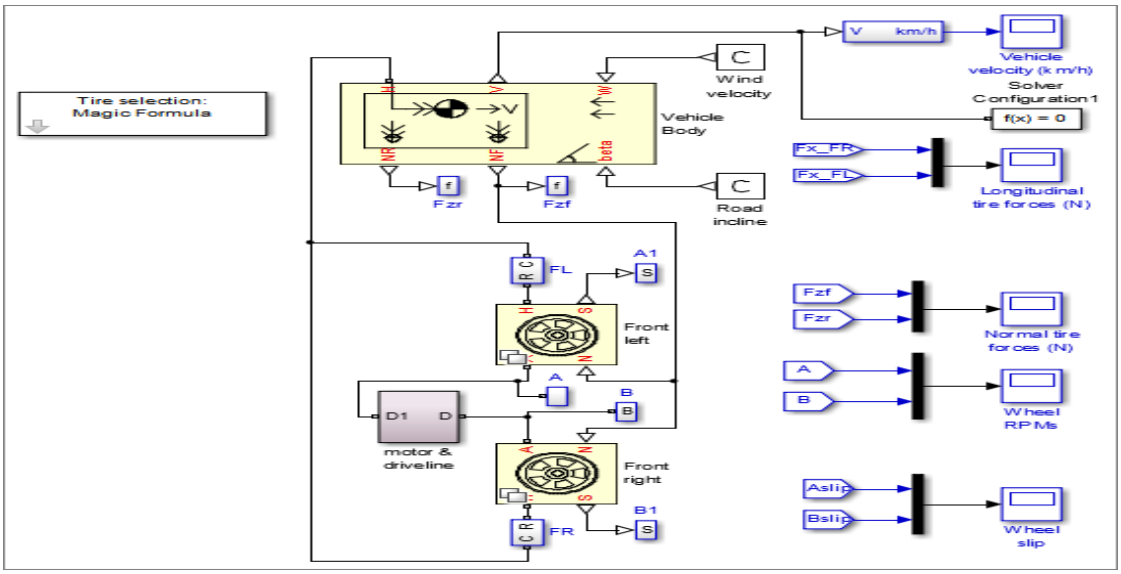
Soft Constraints

Constraint	Assumed Violation	Impact Factor	Sensitivity Ratio
Lower limit: heading angle	18	18.37	1000
Upper limit: heading angle	18	18.37	1000
Lower limit: x position	4.2	1	54.44
Upper limit: x position	4.2	1	54.44
Lower limit: y position	0.6	0.02041	1.111
Upper limit: y position	0.6	0.02041	1.111

Closed-Loop Steady-State Gains

Disturbed	Affected	OVGain
y position	x position	2.23915e-05
y position	y position	0.141267
heading angle	y position	0.113092
y position	heading angle	1.07342
heading angle	heading angle	0.858635

B. Robot body connection with wheels



C. Robot motor connection with supply and gear box