PERFORMANCE COMPARISON BETWEEN PID AND LQR CONTROLLERS FOR DRONE APPLICATION

ALIF IMRAN BIN ZAMREE

UNIVERSITI SAINS MALAYSIA

2021

PERFORMANCE COMPARISON BETWEEN PID AND LQR CONTROLLERS FOR DRONE APPLICATION

by

ALIF IMRAN BIN ZAMREE

A thesis submitted in fulfilment of the requirements for the Bachelor Degree of Engineering (Honours) (Aerospace Engineering)

June 2021

ENDORSEMENT

I, Alif Imran bin Zamree, declare that I have checked and revised the whole draft of the dissertation as required by my supervisor.

(Signature of Student)

Date : 10th July 2021

(Signature of Supervisor)

Name : Ir. Dr. Ahmad Faizul Hawary

Date : 10th July 2021

Assoc. Prof. Dr. ELMI BIN ABU BAKAR
Lecturer & Researcher
Room 1.26, School of Aerospace Engineering,
Engineering Campus, Universiti Sains Malaysia,
14300 Nibong Tebal, Penang
Tel : +604-599 5954 / 5938 Fax : +604 599 6911

(Signature of Examiner)

Name : Associate Professor Dr. Elmi bin Abu Bakar

Date : 11th July 2021

DECLARATION

This thesis is the result of my investigation, except where otherwise stated and has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any other degree.

(Signature of Student)

Date : 10th July 2021

ACKNOWLEDGEMENT

First and foremost, I would like to thank my adviser, Ir. Dr. Ahmad Faizul Hawary, for giving me the opportunity to explore the control system-related field. It was a fantastic experience for me to complete my project with his help, and I would not have been able to do it without his instruction. He is the type of person who is willing to put in the time and effort to assist me, especially in the context of the pandemic situation.

Next, I would like to express my gratitude to the School of Aerospace Engineering's employees and technicians, particularly Mr. Mohd Amir bin Wahab, who provided invaluable support and assistance in the fabrication of the platform structure. Furthermore, the expertise and expertise he shared are priceless, and I have gained from it.

In addition, I would like to thank the University of Southern Mississippi's School of Aerospace Engineering for providing the project with laboratories, resources, and equipment. Furthermore, I am grateful to my other pals who are always willing to share their thoughts and ideas with me anytime I require it.

Finally, I want to express my gratitude to my family members for their unconditional support, assistance, and patience throughout my education. I could not picture my life without everyone with me.

PERFORMANCE COMPARISON BETWEEN PID AND LQR CONTROLLERS FOR DRONE APPLICATION

ABSTRACT

In industries of unmanned aerial vehicle (UAV), the implementation of a motor control system is essential to ensure the system mechanism can be operated efficiently. In addition, DC servo motor systems are widely applied in a variety of fields of UAV. They are used to generate electrical power in power plants and to supply mechanical motive power to operate the UAV and manage numerous industrial operations in industrial settings. In some application of the DC servo motor, when load is applied, or disturbance occur during the operation, the DC servo motor is required to maintain its desired speed to ensure the stability and efficiency of the system. This system can be controlled using PID, Fuzzy, LQR and other more. The PID algorithm becomes a closed loop system when it is added to the motor. The system is developed in MATLAB software, and the PID algorithm is tuned by adjusting the values of proportional gain, Kp, integral gain, Ki, and derivative gain, Kd to get a motor speed and position that is less overshoot, has a longer settling time, and has a longer rise time. To control the Dc servo motor speed and position, the Linear Quadratic Regulator (LQR) controller is introduced. The LQR controller is designed and tuned using MATLAB/Simulink, and it is simulated using a mathematical model of a DC servo motor. A new approach of controlling the motor is the Linear Quadratic Regulator (LQR) controller. The Linear Quadratic Regulator (LQR) is an optimum control theory that focuses on controlling a dynamic system at the lowest possible cost. The purpose of the Linear Quadratic Regulator (LQR) is to minimize the deviation of the motor's speed and position. The input voltage of the motor will be specified by the motor's speed, and the output will be compared to the input. The advantages of using LQR are that it is simple to build and that it improves the accuracy of state variables by estimating them. When contrasted to pole placement, the LQR control has the advantage of specifying a

set of performance weighting rather than needing to define where eigenvalues should be positioned, which may be more intuitive.

PERBANDINGAN PRESTASI ANTARA PENGAWAL PID DAN LQR UNTUK APLIKASI DRONE

ABSTRAK

Dalam industri kenderaan udara tanpa pemandu (UAV), pelaksanaan sistem pengendalian motor sangat penting untuk memastikan mekanisme sistem dapat dikendalikan dengan efisien. Di samping itu, sistem motor servo dc digunakan secara meluas dalam pelbagai bidang UAV. Mereka digunakan untuk menghasilkan tenaga elektrik di loji janakuasa dan untuk membekalkan daya motif mekanikal untuk mengendalikan UAV dan menguruskan banyak operasi industri dalam persekitaran industri. Dalam beberapa aplikasi motor servo DC, ketika beban dikenakan, atau gangguan terjadi selama operasi, motor servo DC diperlukan untuk mempertahankan kecepatan yang diinginkan untuk memastikan kestabilan dan kecekapan sistem. Sistem ini dapat dikendalikan dengan menggunakan PID, Fuzzy, LQR dan lain-lain lagi. Algoritma PID menjadi sistem gelung tertutup apabila ditambahkan ke motor. Sistem ini dikembangkan dalam perisian MATLAB, dan algoritma PID disetel dengan menyesuaikan nilai keuntungan Proportional, Kp, Integral gain, Ki, dan Derivative gain, Kd untuk mendapatkan kelajuan motor dan kedudukan yang kurang overhoot, mempunyai penyelesaian yang lebih lama masa, dan mempunyai masa kenaikan yang lebih lama. Untuk mengawal kelajuan dan kedudukan motor servo Dc, pengawal Linear Quadratic Regulator (LQR) diperkenalkan. Pengawal LQR direka dan ditala menggunakan MATLAB / Simulink, dan disimulasikan menggunakan model matematik motor servo DC. Pendekatan baru untuk mengawal motor adalah pengawal Linear Quadratic Regulator (LQR). Linear Quadratic Regulator (LQR) adalah teori kawalan optimum yang memberi tumpuan untuk mengawal sistem dinamik dengan kos serendah mungkin. Tujuan Linear Quadratic Regulator (LQR) adalah untuk meminimumkan penyimpangan kelajuan dan kedudukan motor. Voltan input motor akan ditentukan oleh kelajuan motor, dan output akan dibandingkan dengan input.

Kelebihan menggunakan LQR adalah bahawa ia mudah dibina dan meningkatkan ketepatan pemboleh ubah keadaan dengan menganggarkannya. Apabila dibandingkan dengan penempatan tiang, kontrol LQR memiliki kelebihan menentukan set pemberat kinerja daripada perlu menentukan di mana nilai eigen harus diposisikan, yang mungkin lebih intuitif.

TABLE OF CONTENTS

LIST OF FIGURES

LIFT OF TABLES

LIST OF ABBREVIATIONS

PID	Proportional-Integral-Derivative
UAV	Unmanned Aerial Vehicle
MATLAB	Matrix Laboratory
LQR	Linear Quadratic Regulator
K_p	Proportional Gain
K_i	Integral Gain
K_d	Derivative Gain
E_{ss}	Steady-state Error
T_s	Settling Time
T_r	Rise Time
K_r	Reference Gain
K	Optimal Gain

CHAPTER 1

INTRODUCTION

1.1 Overview

The aim of this project is to design and implement a PID and a Linear Quadratic Regulator (LQR) controllers for a DC servo motor. With a given performance requirement, the PID and LQR controllers are utilized to regulate the speed and position of the DC servo motor. To verify that the controller meets the requirements, simulations will be run using the MATLAB/Simulink software to determine the best PID and LQR controller parameters. The PID controller's performance will then be compared to that of the LQR controller.

1.2 Research Background

Rotorcraft have been in development since the dawn of aviation and were initially designed as a means of achieving extremely low speeds. (Green, 2019) Rotorcraft was divided into two categories. One example is a helicopter with a single rotor configuration. A multirotor is another kind. The platform of a multirotor will has up to eight rotors. The additional motors improve stability, durability, and lifting power. (Security, no date) Quadcopters are classified as rotorcraft, as opposed to fixed-wing aircraft, because the lift is generated by a set of rotors (vertically oriented propellers). A quadcopter's thrust is provided by four propellers, which are positioned in a cross or plus pattern as in Figure 1 - 1. The quadcopter can take off and land vertically, which is a significant benefit because it reduces the landing platform requirements. (Ostojić *et al.*, 2015)

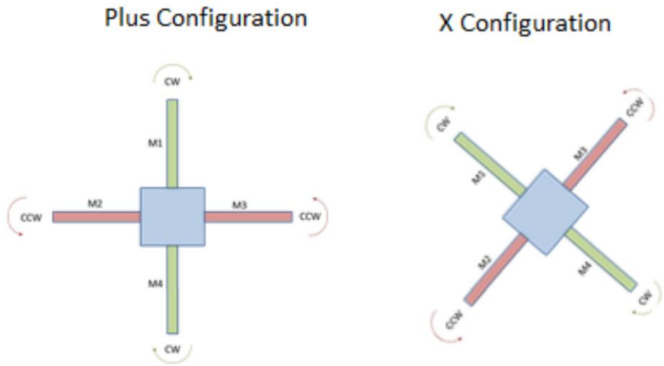

Figure 1 - 1 Cross and Plus Quadcopter Configurations (Fiaz and Mukarram, 2018)

Quadcopters, like most aircraft, are highly unstable systems, and any imbalance in their motion (especially in roll and pitch) generates angular and linear accelerations, which can cause a crash if not accounted for rapidly. Moreover, the quadcopter system is non-linear, and the movements are coupled with each other. (Romero, Pozo and Rosales, 2015) For all flying platforms, the wind is a key cause of disturbance. The ability to maintain stability while minimizing wind disturbances would broaden the scope of UAV applications. (Bannwarth *et al.*, 2016) According to the control strategy of quadcopters, implementation of a flight control system method can increase the stability of the quadcopter when there is a disturbance. The flight control systems can be divided into 3 types which is linear flight control system, learning based flight control system and non-linear flight control system. The feedback linearization approach, also known as the linear flight control system, is a very effective approach for decoupling and linearizing quadcopter attitude models.(Kuantama *et al.*, 2017)

A controller's function in a process control loop is to influence the control system via a control signal such that the controlled variable's value equals the reference variable's value. The controller is often referred to as the "Brain" of the process control room. Depending on the difference between the set point and the measured value of the controlled variable, the controller sends a control signal to the final control element. The controller's response to deviation is referred to as controller mode. Sensors, transmitters, and control valves are usually positioned around the process, while the controller is either on the panel or in the computer memory as a program. (Khan, Khan and Ghazali, 2015)

A control system is used to adjust the behaviour of a system over time such that it responds in a desired manner. The system can be presented with a control block diagram as Figure 1- 2 which involving input and output relationship. The input-output relationship represents the process' cause-and-effect relationship, which in turn represents the processing of the variable input signal to supply and output signal. (Dorf and Bishop, 2010) In a control system, a proportional integral derivative controller (PID) is a typical approach for linear control system as it is a very effective approach as mentioned previously. PID theory contributes to the development of a better control equation for the system. Simple structure, high performance for several processes, and customizable even without a specific model of the controlled system are only a few of the benefits. (Bin Awang Besar, 2013)

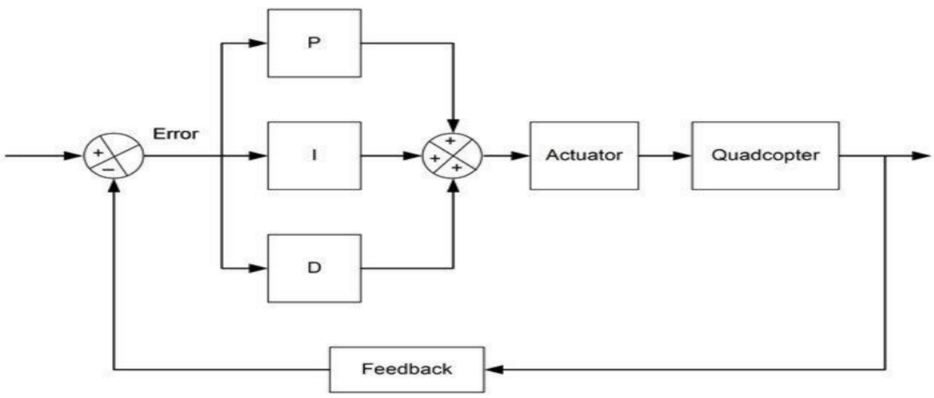

Figure 1 - 2 General PID block diagram (Hesham, no date)

A new method of controlling the motor besides PID is the Linear Quadratic Regulator (LQR) controller. The LQR is an optimal control theory that focuses on operating a dynamic system at the lowest possible cost. The purpose of the LQR is to reduce the deviation of the motor's speed and position. (Haron, 2013) The LQR is a controller that uses a state-space approach. This control is an optimal control that produces a steady state minimum error and has robust characteristics with regulator property. This method can also be used to quickly resolve any system errors that arise. As a result, the system can withstand environmental disturbances while remaining stable. (Dhewa, Dharmawan and Priyambodo, 2017) The quadratic cost function is minimized in LQR to provide optimal control. As a result, using LQR to control a quadcopter is interesting. (Ahmad *et al.*, 2020)

1.3 Problem Statement

The challenge of this project is to observe the system response of the propeller that causes the system to be unstable. It is proposed that the controllers that will be designed are the PID and the LQR controllers to monitor the stability and control the performance of the propeller's system. Then, analysis need be conducted to determine the best controller in improving the stability of the system's response.

1.4 Research Objectives

The general objective of this study is to design a controller for UAV/drone in order to obtain the best controller that can improved the quadcopter disturbance rejection. In order to achieve this aim, three specific objectives were set out as follows:

1. To design a controller structure that can control quadcopter.
2. To analyse the performance of the controller and compare different method.
3. To propose control strategy that is able to stabilize the platform system.

1.5 Thesis Outline

There are five chapters in this thesis report. First chapter will give a brief introduction about this project and objectives to be achieved for this project.

Chapter 2 covers the literature review performed for this project. Drone design and control system design theory are studied and reviewed. Control system design is researched in more detail on system identification, system linearization and controller design which included PID and LQR controller.

Chapter 3 contains methodology utilized in this project. Flow of work is first initiated by making research on related design of project system followed by design, fabrication, and simulation of the project in software.

Chapter 4 presents the results of simulation along with analysis of the project and discussions. Performance of both PID and LQR controller will be compared in terms of transient response, steady-state response, and design approaches.

Chapter 5 as the last chapter will embraces conclusions of the project which summarize the work throughout this project. Moreover, recommendations are included to enhance the further research on this project or title concerned with this project.

CHAPTER 2

LITERATURE REVIEW

2.1 Concepts of Quadcopter

Quadcopter control is a complicated but fascinating challenge. They have six degrees of freedom (three translational and three rotational), but only four independent inputs (rotor speeds), resulting in a mechanism that is underactuated. To obtain six degrees of freedom, translational (up, down, left, right, forward, and backward) and rotational motions (roll, pitch and yaw) are combined as shown in Figure 2 - 1. This produces highly non-linear dynamics, especially when complex aerodynamic effects are taken into account. (Lopez, 2019) The two pairs of opposite rotors rotate in the opposite direction, as seen in the Figure 2 - 2, which not only balances the torque but also eliminates the need for a third rotor for stabilization. The change in altitude is achieved by varying the speeds of all the motors at the same time. (Fiaz and Mukarram, 2018)

The yaw movement is achieved by decreasing the speed of one pair of propellers in the same direction while increasing the speed of the other pair; this produces an unbalanced torque on the quadrotor while maintaining constant downward thrust. (Fiaz and Mukarram, 2018) The quadrotor is a well-known example of an underactuated system. Although the quadrotor has six degrees of freedom (3 translational and three rotational), it only has four control inputs (the speeds of each motor). The quadrotor can move vertically on the z axis without changing its state, but it must adjust its attitude to move horizontally on the x and y axes. Since controlling six degrees of freedom with just four control inputs is impossible, the controllers are designed to stabilize around desired x, y, and z positions as well as a desired heading. The quarter should be able to reach this location safely while maintaining stable roll and pitch angles. (Selby, no date)

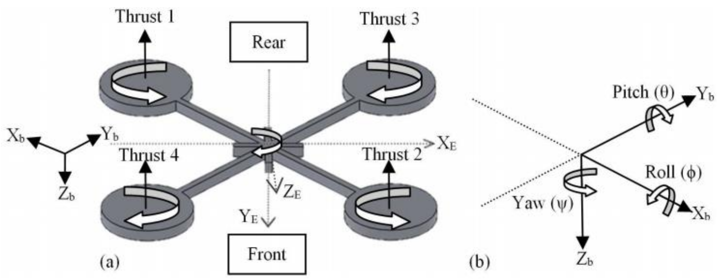

Figure 2 - 1 The Six Degrees of Freedom of The Quadcopter (Kuantama *et al.*, 2017)

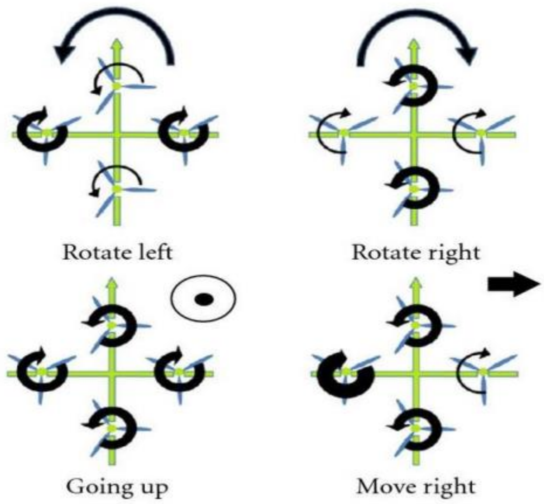

Figure 2 - 2 Basic Quadrotor Dynamics (Fiaz and Mukarram, 2018)

Quadrotor control is accomplished by differential control of each rotor's thrust. Vertical motion is achieved by simultaneously raising or lowering the speed of all four propellers.The differential in speed of the front and back sets of rotors creates pitch motion. Using a left-right set of rotors, roll motion is preserved in the same way. Quadrotors are underactuated, which implies that forward/backward and left/right motions are linked to pitch/roll motions and may be controlled through them. Increase the speed of the two clockwise rotating rotors while reducing. Quadcopters are very unstable systems, and any imbalance in their motion (particularly in roll and pitch) causes angular and linear accelerations, which can cause a collision if not adjusted for fast. Furthermore, the quadcopter system is non-linear, and the motions are synchronized. (Romero, Pozo and Rosales, 2015)

2.2 DC Motor of Quadcopter

DC motors are a type of electrical actuator that can be controlled very precisely and are commonly used in quadcopters. In terms of speed control, DC motors are extremely versatile. DC motors have generally been recognised as the most ideal choice for variable speed applications due to the precise speed control, controllable torque, excellent dependability, and simplicity. The field winding of a DC motor is directly connected to the power supply. Field resistance control, armature resistance control, and armature voltage control are the three most common speed control techniques. In the armature voltage control method, the field current is kept constant while a variable voltage is applied to the armature. The speed of a DC motor is directly proportional to the DC motor's applied armature voltage, according to the basic working principle of an armature controlled DC drive. (Khan, Khan and Ghazali, 2015)

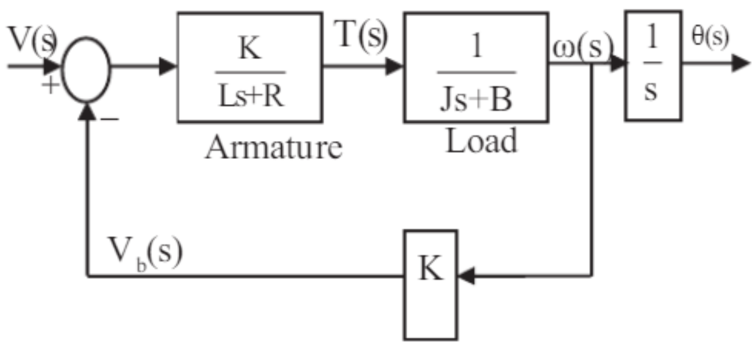

Figure 2 - 3 Block Diagram of DC Motor (Rajesh and Ananda, 2015)

The speed of a DC motor can be controlled by adjusting the terminal voltages proportionally. There have been various methods used to control the DC motor through proper tuned regulation of terminal voltages, including the use of PID, LQR and SMC controllers. Aside from the DC motor's speed, the DC motor's various loading effect is also a key consideration in the DC motor's operation. (Khan, Khan and Ghazali, 2015) PID controllers provide a basic but effective method for aircraft stabilization by allowing each variable to be treated individually within a narrow range where the quadcopter's behaviour is essentially linear. (Romero, Pozo and Rosales, 2015) The PID controllers are commonly used due to their simplicity and high performance in many situations. A well-tuned PID can solve the most of industry problems. The intuitive idea behind each of these three components is simple.

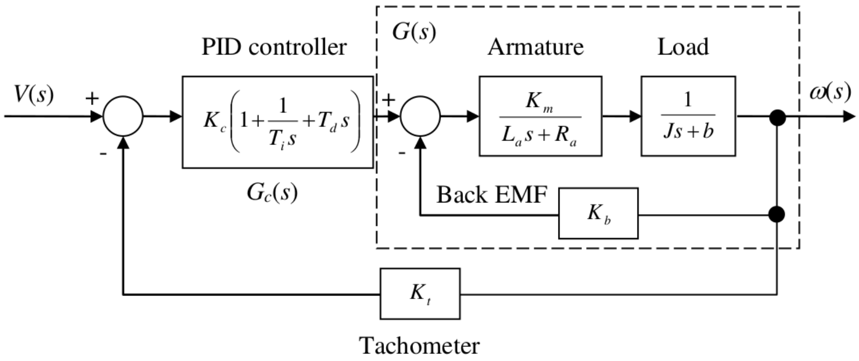

Figure 2 - 4 Block Diagram of The Quadcopter Whole System (Anirban, 2019)

2.3 PID Controller

The PID control scheme is named after the manipulated variable, which is made up of three correcting terms: proportional, integral, and derivative terms. PID controllers provide a simple but effective solution for quadcopter stabilization by allowing each variable to be treated independently within a limited range where the quadcopter's behaviour is approximately linear. (Romero, Pozo and Rosales, 2015)

2.3.1 PID Controller Theory

The proportional (P) controller produces a control signal proportional to the current error. To put it another way, as the error grows, the P block produces higher control signals to reduce it. The integral (I) term creates a signal proportional to the number of previous error values. When a single P controller fails to reach zero error in steady state, this block comes in handy. When this happens, a constant error remains in the controller's input, and a simple I block can integrate that constant to produce an increasing control signal, leading to the error being reduced to zero. Finally, the derivative (D) term generates a control signal proportional to the error signal's rate of change. Thus, it has a predictive effect. To avoid reacting to noisy data, this term usually has a built-in low-pass filter. The steady state is unaffected by this block. (Federal *et al.*, 2014) The PID algorithm's final form, which is defined as the controller output, is as follows (Swain, Mandavi and Behera, 2012) :

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad (1)$$

Taking the Laplace transform will obtain equation (2)

$$\frac{u(s)}{e(s)} = K_p (1 + \frac{1}{\tau_i s} + \tau_d s) \quad (2)$$

$$K_p + \frac{K_i}{s} + K_d s = \frac{K_d s^2 + K_p s + K_i}{s} \quad (3)$$

Where:

K_p = Proportional gain, a tuning parameter

K_i = Integral gain, a tuning parameter

K_d = Derivative gain, a tuning paramter

e : Error = Set point − Process value

t = Instantaneous time

τ_i = Integral time

τ_d = Derivative time

Greater values for proportional gain, K_p usually indicate a faster response since the proportional term compensation increases as the error increases. Process instability and oscillation will result from a relative gain that is too high. For integral gain, K_i larger values indicate that steady-state errors are eliminated faster. More significant overshoot is the trade-off: any negative error integrated during the transient response must be balanced by positive error before we achieve a steady state. Larger values for integral gain, K_i reduce overshoot while delaying transient response and potentially causing instability due to signal noise amplification in the error differentiation. (Haron, 2013) The table below summarizes the general impacts of each controller parameter on a closed-loop system. Note that while these criteria apply in many circumstances, they do not apply in all.

Closed-Loop Response	Rise Time	Overshoot	Settling Time	Steady-State Error	Stability
Increasing K_P	Decrease	Increase	Small Increase	Decrease	Degrade
Increasing K_I	Small Decrease	Increase	Increase	Large Decrease	Degrade
Increasing K_D	Small Decrease	Decrease	Decrease	Minor Change	Improve

Figure 2 - 5 Effect Of Independent P, I, And D During The Tuning Process (Alargt and Ashur, 2015)

2.3.2 PID Performance Criteria

Defining the performance criteria is the first step in the control design process. Using a step function as the set point command variable and then measuring the response of the process variable is a common way to evaluate control system performance. In most cases, the reaction is measured using waveform characteristics that have been determined. The rise time is when it takes for the system to move from 10% to 90% of the steady-state or final value. The amount by which the process variable overshoots the final value, represented as a percentage of the final value, is the percent overshoot. Settling time is the amount of time it takes for a process variable to settle to within a specific percentage of the final value (usually 5%). As illustrated in Figure 2 - 3, steady-state error is the ultimate difference between the process variable and the set point. It is important to note that industry and academics have different definitions of these terms. (Zomrawi, Mohammed and Jedo, 2017)

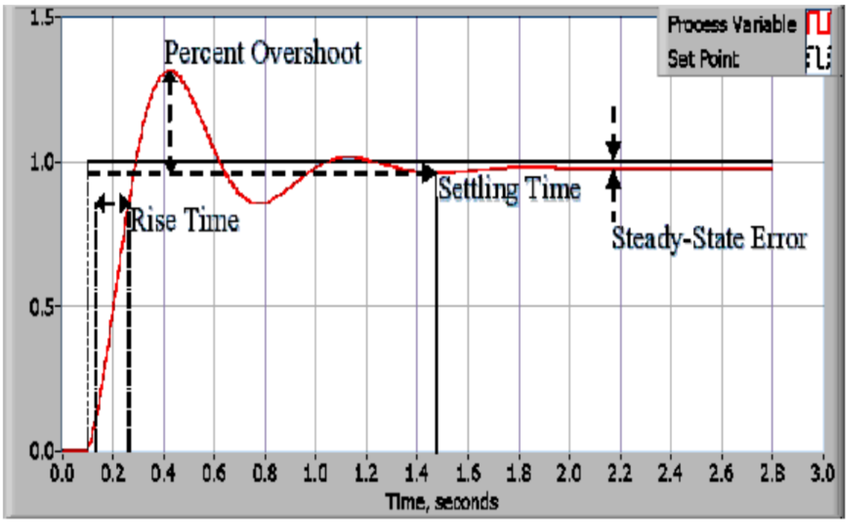

Figure 2 - 6 Response of a typical PID closed-loop system (Zomrawi, Mohammed and Jedo, 2017)

The PID control system can be designed using Matrix Laboratory (MATLAB) software to observe the response of the PID system. The PID Controller design may be carefully developed using MATLAB simulation. MATLAB is a programming language that may be used to create Since MATLAB offers a Control Toolbox facility, which makes it extremely easy to design in evaluating response design outcomes that are set from numerous variable input variables, it is straightforward to use in terms of design simulation and signal response analysis. As establishing the input variables might also combine many systems of PID control, this MATLAB program is required in the design to produce the desired output. Combining PI, PD, and PID in the system design is possible. (Anggoro *et al.*, 2020)

2.3.3 PID Tuning

In PID control design, the parameters of the controllers are tuned manually or through linear control-based tuning methods, such as Ziegler-Nichol tuning. Unfortunately, manual tuning has the drawback of being time-consuming and complex. Many other researchers have criticized the traditional Ziegler-Nichol technique, which has proved valuable and helpful in terms of practicality. The Ziegler-Nichol tuning method was the quickest and required the slightest effort to determine the ideal PID gains experimentally. The PID gain settings required to stabilize the quadrotor can be reached within a few minutes of implementation. The Ziegler-Nichol technique was the most efficient in terms of time and effort. (Khodja *et al.*, 2017)

Tuning is the process of analyzing transients in order to improve responses. Tuning methods are validated by plotted transient responses, even when motivated by other arguments. In addition to the design, the reaction time must be tuned to obtain a steady system output in line with the designer's intentions. Using MATLAB simulation, the design of the desired control system is analyzed, and the desired response output can be determined so that it is easier to implement and determine the input variables. In order to achieve the stability of the quadcopter, the controller concepts must be considered the system response to input variables such as step function, ramp function, and impulse function and noise, the stability of the system using the model: root locus, frequency of response, state-space and system response by combinations of PIDs. (Anggoro *et al.*, 2020)

2.4 LQR Controller

For a quadcopter, LQR is an ideal control regulator that is supposed to be more robust. Rather than the traditional linear equation method of PID, LQR focuses on non-linear models. The fundamental disadvantage of PID controllers is that they require linearization for every test on the existing system. This step is not required for LQR control; instead, the system equations can be fed into the controller, resulting in the desired response. (Saraf, Gupta and Parimi, 2020)

2.4.1 LQR Controller Theory

The linear quadratic regulator (LQR) is a common approach for determining the state feedback gain in a closed-loop system. For specified weighting matrices of the cost function, this is the best regulator for relocating the open-loop poles to provide a stable system with optimal control and the least cost. However, when utilizing the optimal regulator method, that flexibility of choice is lost for both discrete-time and continuous-time systems. There are some locations where the poles cannot be assigned to provide a positive-definite Riccati equation solution. The benefit of LQR is that it can improve the system's performance by managing the motor speed and position. To minimize the energy used by the control action itself to a minimum, the magnitude of the control action is frequently included in this amount. The LQR technique is essentially a computer-assisted search for a suitable state-feedback controller. (Haron, 2013)

On a fully dynamic model of the quadrotor, a primary path following the LQR controller was implemented. Despite the presence of wind and other disturbances, accurate path following was demonstrated in simulation utilizing optimal real-time trajectories. After evading a barrier, the controller appeared to lose track. Its performance in the face of several challenges was still being studied. The LQR technique becomes the Linear Quadratic Gaussian when combined with a Linear Quadratic Estimator (LQE) and a Kalman Filter (LQG). For systems with Gaussian noise and partial state information, this approach is used. In hover mode, the LQG with necessary action was used to stabilize the attitude of a quadrotor with good results. The advantage of this LQG controller is that it can be implemented without having complete state information. (Zulu and John, 2014)

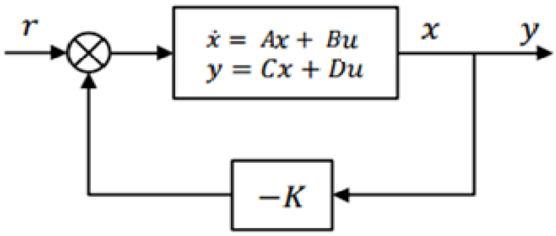

Figure 2 - 7 LQR Block Diagram (Gaol, Kurniawan and Setyawan, 2019)

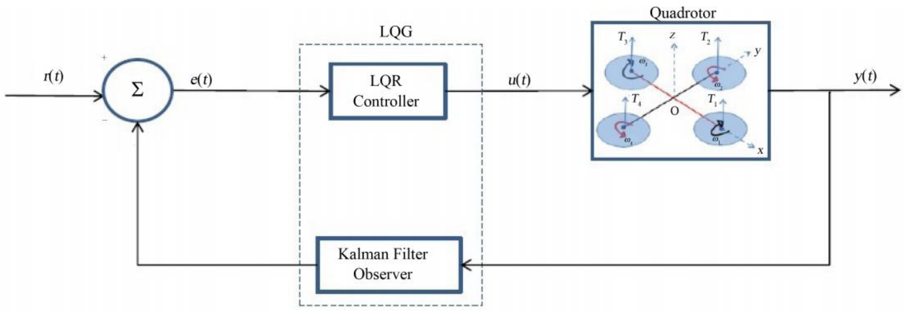

Figure 2 - 8 Block diagram of LQG controller applied to the quadrotor (Zulu and John, 2014)

2.4.2 LQR Control Technique

LQR is a popular optimum control strategy that involves a cost function that depends on the states of any system and the control inputs. As LQR requires a linearized model, the quadrotor's dynamic model is linearized using the Jacobian method as follows:

$$\delta\dot{X}_s = A\delta X_s + B\delta U \quad (4)$$

If the model is linearized at a nominal point $((X_{ss,} U_{ss})$ where $X_{ss} = [x_{ss}, y_{ss}, z_{ss}, \psi_{ss}]^T$ and other states are considered as zero with $U_{ss} = [mg\ 0\ 0\ 0]^T$

Where,

$$\delta X = X_d - X_a \tag{5}$$

$$U = U_{ss} + \delta U \tag{6}$$

$$X = X_{ss} + \delta X \tag{7}$$

A feedback control must be designed using the equation below when using the LQR control approach.

$$U = -K\delta X_s + U_{ss} \tag{8}$$

K. denotes the feedback gain matrix. It was determined using a cost function that was minimized.

$$J = \int_0^\infty (\delta X_s Q \delta X_s + \delta U R \delta U) dt \tag{9}$$

Q is a 12x12 semi-positive definite matrix for this quadrotor, while R is a 12x44 positive definite matrix. As a result, the closed-loop control system appears to be as follows:

$$\delta \dot{X}_s = (A - BK)\delta X_s \tag{10}$$

Where K has been determined by combining the Q and R matrices. The linear model is then substituted with a non-linear model to provide proper quadrotor control with the same gain K. (Islam, Okasha and Mohammad Idres, 2017)

2.4.3 LQR Controllability

Controllability requirements may govern the availability of a complete solution to the control system design problem. If the system under consideration is not controllable, there may be no solution to this problem. If it is possible to generate an unconstrained control signal that will transfer an initial state to any final state in a finite time interval $t_0 \leq t \leq t_1$, the system described by Equation (4) is said to state controllable at $t = t_0$. The system is said to be entirely state-controlled if every state is controllable. The system is given by equation (4) ultimately states controllable if and only if the vectors B, AB, $A^{n-1}B$ is linearly independent or the n x n matrix [B, AB, …, $A^{n-1}B$] is of rank n. (Mahdi, Al-Bermani and Jaafar, 2014)

In term of weighting matrices Q and R determination, the Q and R weighting matrices are crucial parts of the LQR optimization process. The Q and R element compositions have a significant impact on system performance. The designer has complete freedom in selecting the matrices Q and R; however, this is usually done in an iterative process based on experience and physical understanding of the challenges at hand. The matrices Q and R elements are commonly constructed using a trial-and-error process. This method is straightforward and well-known in the LQR application. However, selecting the best values for matrices Q and R take a lengthy time. The number of matrices Q and R elements is proportional to the number of state variables (n) and input variables (m). (Mahdi, Al-Bermani and Jaafar, 2014)

2.5 PID and LQR Comparison

Due to its versatility and easy implementation, the PID controller was chosen in this project while also providing consistent reaction to the attitudes of model dynamics. Also, because of its outstanding performance and robustness in the plant in question, the LQR controller seemed to be an excellent comparative controller to be used in this project. A few studies on the performances of LQR and PID show that each of these controllers provides unique features that make it hard to say which one is the best. The LQR controller is known to be robust and produce a minimal steady-state error. However, with a significant transition delay and six feedback gains, this makes them a wrong choice when the system needs to update fast parameters and has no direct access to all plant states. A PID controller, on the other hand, provides a faster response, but not with robust gains like the previous controller. The conventional theory of PID does not imply the development of a robust controller. (Argentim *et al.*, 2013)

CHAPTER 3

METHODOLOGY

3.1 Project Overview

This paper is conducted based on 1/4 of the quadcopter model, which means only one propeller will be analyzed to determine which controller is the best in stabilizing the PID or LQR controller. From the model, System Identification Toolbox is used to extract the transfer function of the plant so the control block diagram of the system can be designed. Then, the plant transfer function analysis is conducted, and the controller can be designed once the plant's response is obtained. The system can be analysed when the controller is implemented into the system to observe if the system can meet the design requirement and achieve the desired performance. Lastly, PID and LQR controllers can be compared to see which controller can achieve the best result in stabilizing the system.

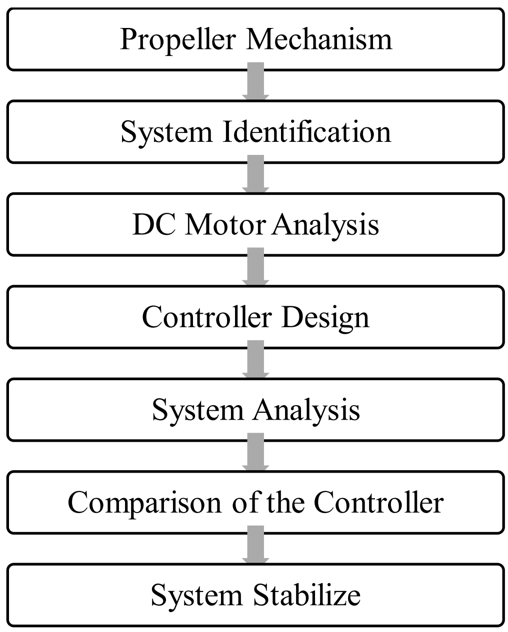

Figure 3 - 1 Overall Project Flowchart

3.2 Plant Analysis

Plant analysis is conducted first to observe the response of the system. Through this process, we can strategize the controllers that will be designed to improve the system's response so the system can be stabilized.

3.2.1 Transfer Function of the DC Motor

The flow diagram of the identification algorithm is shown in Figure 3 - 2. The data must first be prepared for system identification before the model can be obtained. The data from the experiments must then be imported into the MATLAB environment. Using the iddata function, the data must be expressed in time and frequency domains. The iddata object created by this function has a time-domain output signal y and an input signal u. In addition, the experimental data sample time must be given. Then, to construct independent data sets for estimate and validation, data is selected. The next step is to choose the model to employ to depict the linear system. A numeric model of a linear system with fixed numerical coefficients is the transfer function model. The numerator and denominator of the transfer function are those fixed coefficients. The transfer function model for the MATLAB System Identification Toolbox uses a ratio of polynomials to explain the link between the inputs and outputs of a dynamic system.

The transfer function model, in continuous time, has the following form:

$$\mathrm{Y(s)} = \frac{\mathrm{num\ (s)}}{\mathrm{den\ (s)}}\mathrm{U(s)} \tag{11}$$

Where Y (s) and U(s) are the output and input Laplace transforms, respectively. A single-input single-output (SISO) continuous transfer function is represented by equation (11). The validation model is the final step in the process of ensuring that the model obtained is accurate. The validation data is used to determine how the model system acts and, if necessary, to refine the model. (Aboytes Reséndiz and Rivas Araiza, 2016)

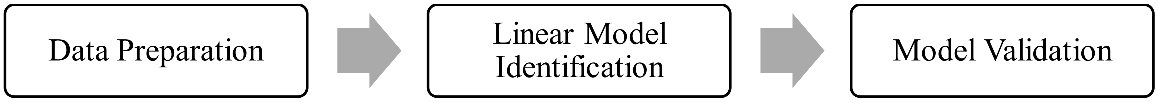

Figure 3 - 2 Flow Diagram of the System Identification

3.2.2 Step Response of the Plant

Before designing the controllers, the transfer function obtained from the System Identification Toolbox is required to analyze first to observe the system's stability. The model is imported into the MATLAB workspace after obtaining the optimal transfer function model of the propeller motor from the System Identification Toolbox. The analysis is conducted by observing the step response of the DC motor. When using MATLAB to determine step response, care should be taken to ensure a high level of accuracy. The analysis is conducted by converting the transfer function into the step response graph. A response that can be observed from the graph is steady-state error, settling time, rise time, and overshoot. By obtaining these data, the system's response can be improved by introducing a controller into the system. This controller can be used to stabilize the lacking of the propeller’s system.

3.3 Designing PID Controller

After the analysis of the uncontrolled DC motor is conducted, the controller can be designed to improve the control system of the propeller. Before PID controller is created, desired performance of transient and steady-state response should be set as a reference value for tuning such as settling time, T_s, peak time, T_p, overshoot percentage, OS% and steady-state error, E_{ss}. Repeating of adjusting K_P, K_i and K_d is occurred in tuning the gain values by using any approach since the approaches give reference values instead of optimal values. Simulink model of the propeller system is created to design the PID controller with the DC motor as a plant.

A PID controller can be created in Simulink using two alternative ways. In Simulink's library browser, there is a block called PID. We can use the built-in PID block to construct the PID controller or use the block diagram in Figures 1 - 2 to design our own PID controller. MATLAB and Simulink are launched first. Then, as indicated in Figure 3 - 3, open the library browser and choose the continuous sub-block from the library browser.

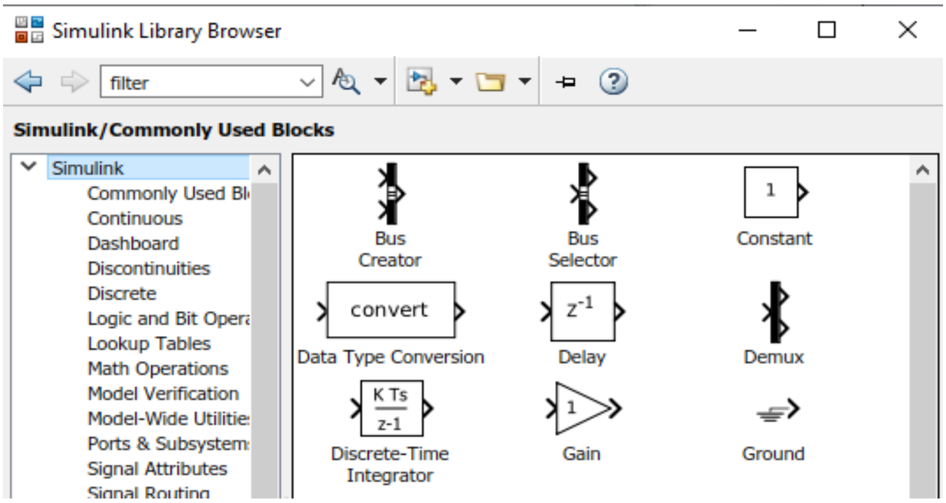

Figure 3 - 3 Continuous Sub Block

In the library browser, double-click the continuous block and then pick the PID block, as shown in Figure 3 - 4.

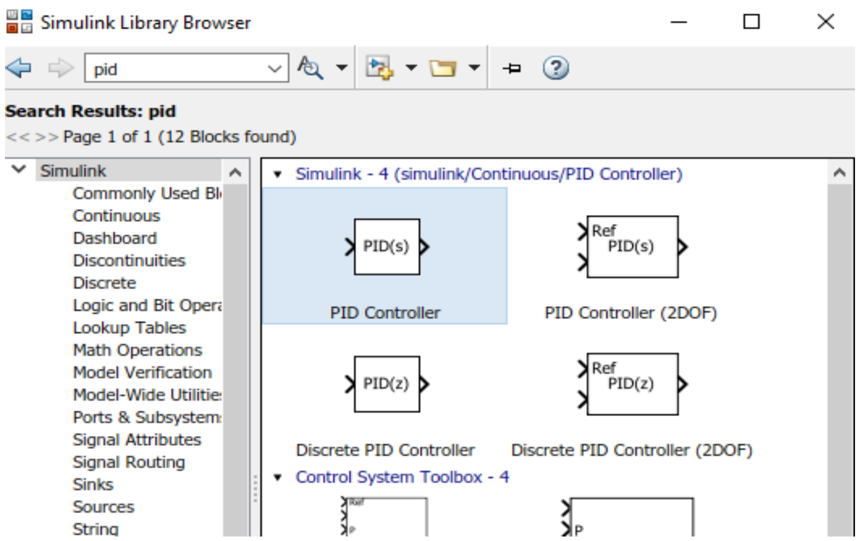

Figure 3 - 4 PID controller

The PID controller will be implemented in this block. We will need a supply, or a step response, to apply to the system. Select the sources as shown in Figure 3 - 5 from the Simulink library browser.

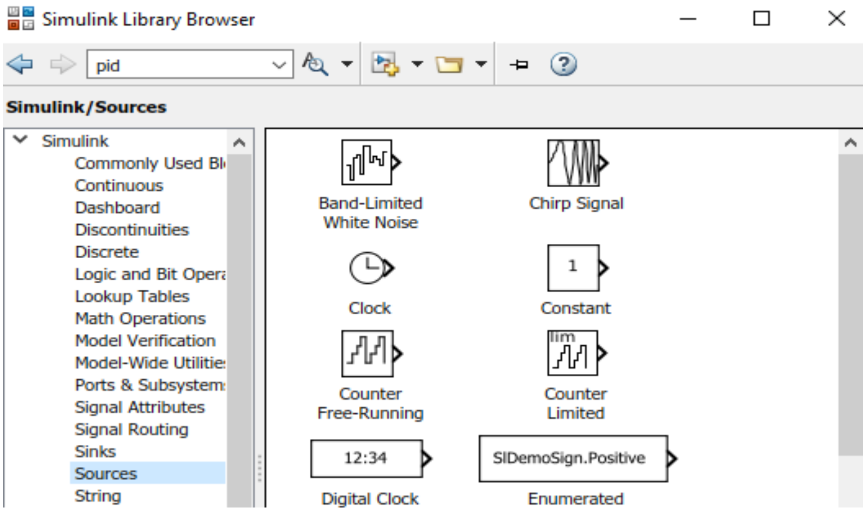

Figure 3 - 5 Sources

Select the Step block, which will be used as an input source to the PID block, from the Sources subsection, as shown in Figure 3 – 6.

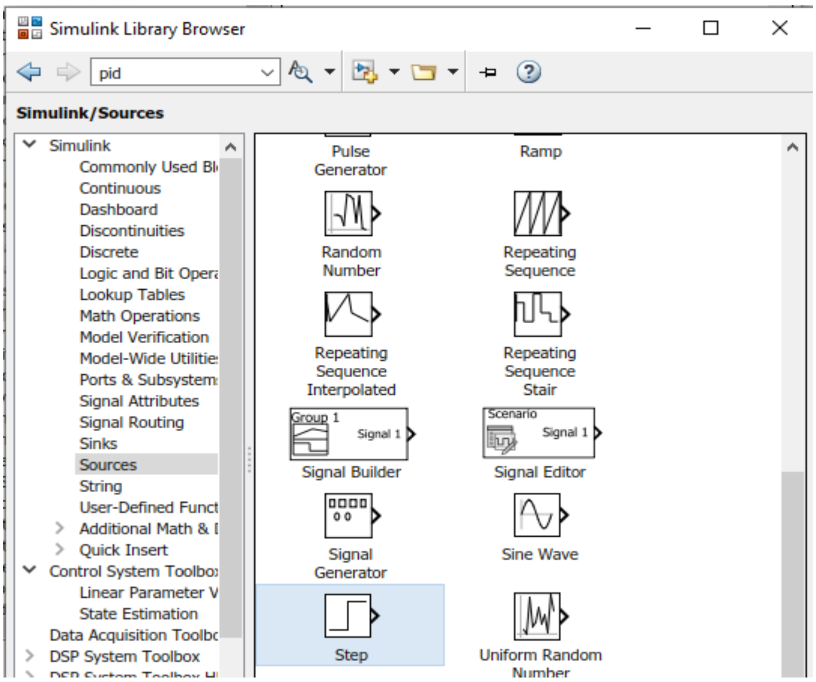

Figure 3 - 6 Step Response

As indicated in Figure 3 - 7, select the Sinks subsection from the list on the left side of the library browser.

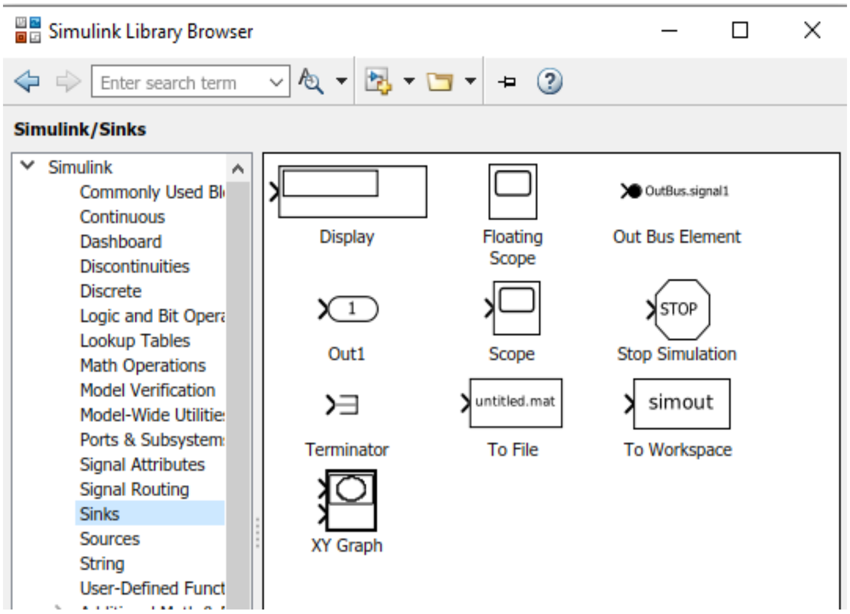

Figure 3 - 7 Sinks

Select the scope that will be used to display the output from the Sinks subsection. The scope block is also available in the library browser's commonly used blocks section. The alternative way to search for the block is in the library browser of Simulink is by typing its name into the library browser's search field.

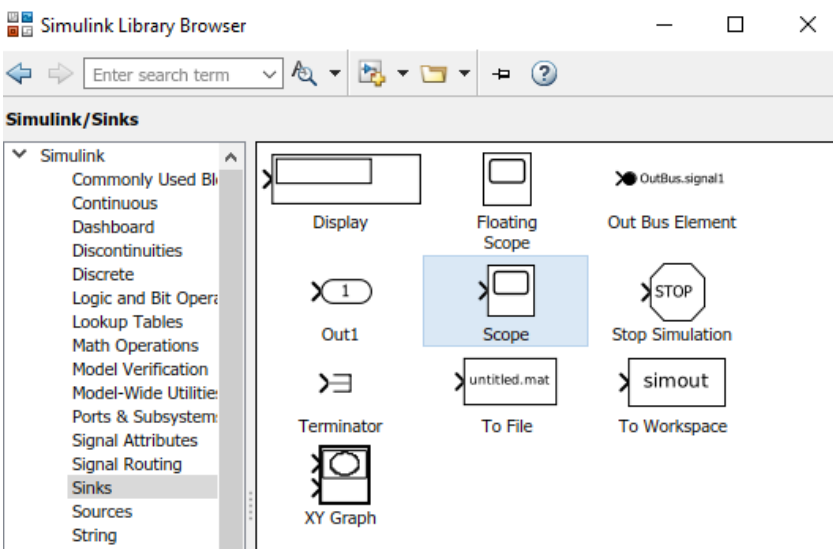

Figure 3 - 8 Scope